UMONS
Université de Mons

GMECA
Polytech Mons

Doctoral dissertation

Thesis submitted at the Faculty of Engineering by

Lorenzo Colantonio

In fullfillment of the requirements for the degree of

Doctor in Engineering Sciences and Technology

Cutting Tool Condition Monitoring and Forecasting with Artificial Intelligence Approaches

Composition of the Jury :

Prof.	Bernard Gosselin	Université de Mons	Chairman
Prof.	François Ducobu	Université de Mons	Supervisor
Dr.	Lucas Equeter	Université de Mons	Co-Supervisor
Prof.	Edouard Rivière-Lorphèvre	Université de Mons	Secretary
Prof.	Pedro-José Arrazola	Mondragon Unibertsitatea	Member
Prof.	Pierre Dehombreux	Université de Mons	Member
Prof.	Phuc Do	ENSM d'Alès	Member
Dr.	Bovic Kilundu	HE2B-ISIB	Member
Prof.	François Pérès	UTTOP	Member

November, 2025

Faculty of Engineering
Machine Design and Production Engineering Lab

Acknowledgements

J'aimerais remercier toutes les personnes qui ont rendu ces cinq dernières années inoubliables.

Mes premières pensées vont à François Ducobu et Lucas Equeter, qui m'ont accompagné tout au long de ces années et m'ont offert leurs conseils et leur temps malgré des agendas bien chargés. Chaque discussion avec vous a été un plaisir, grâce à votre bienveillance et à vos recommandations toujours pertinentes. Votre expertise scientifique et vos qualités humaines ont été une source d'inspiration et ont grandement enrichi mon parcours. Un grand merci à vous deux.

Je souhaite également remercier Pierre Dehombreux, qui a suivi ma thèse et, surtout, m'a fait confiance il y a maintenant plus de cinq ans. Je me rappelle encore notre appel téléphonique après mon TFE, lorsque tu m'as proposé de devenir assistant au sein du service GMECA. Ta sympathie a été une belle vitrine pour le service et m'a convaincu de me lancer dans cette aventure. Merci pour tous tes conseils et pour m'avoir offert cette opportunité de réaliser une thèse.

Je tiens également à exprimer ma gratitude à tous les membres du comité de thèse : Bernard Gosselin, Edouard Rivière-Lorphèvre et François Pérès, qui m'ont accompagné et guidé au cours de ces années, et ont contribué à la relecture de ce travail. Un grand merci également aux membres externes du jury : Pedro-José Arrazola, Phuc Do et Bovic Kilundu, pour votre relecture et votre participation à ma défense.

Je tiens également à remercier Olivier Verlinden, qui m'a permis de découvrir mon intérêt pour la recherche. Réaliser mon TFE avec toi m'a fait voir la recherche sous un angle que je ne connaissais pas. Merci de m'avoir aidé à me lancer dans cette aventure. J'en profite également pour remercier tout le service MRDV, et en particulier Kevin Nis, qui m'a aidé lors des acquisitions de signaux pendant ma campagne expérimentale ; Georges Kourousis pour ses recommandations sur le traitement des signaux vibratoires ; ainsi que Bryan Olivier (ou Olivier Bryan, personne ne sait...) qui m'a accueilli dans sa famille Spotify, sans laquelle cette thèse aurait été bien silencieuse.

J'ai eu la chance de réaliser ma thèse au sein du laboratoire GMECA. Bien que le bâtiment ne soit pas particulièrement bien isolé, les personnes qui y travaillent le réchauffent par leur convivialité (et parfois grâce à des chauffages d'appoint

quand il fait vraiment froid). Dans ce lieu, les collègues deviennent des amis, malgré des personnalités parfois très différentes. Et en cinq ans, je peux vous assurer que j'en ai croisé des phénomènes ! À mon arrivée, j'ai été accueilli par Clément Dutoit et Anthonin Demarbaix. Je tiens à vous remercier car, même si nos chemins ne se sont que brièvement croisés, vos conseils m'ont beaucoup appris et je les applique encore aujourd'hui. J'ai également rencontré le docteur (à l'époque encore monsieur) Spitaels, qui est très vite devenu un ami proche et une source d'inspiration tout au long de cette thèse. Bien que tu ne maîtrises pas encore totalement le concept de la chaîne du froid, tu excelles dans celui de l'écoute, et je voulais t'en remercier sincèrement. Je souhaite aussi remercier Valentin Dambly : lui aussi sait écouter... mais surtout parler ! Nos discussions et nos soirées en ligne ont rythmé ma thèse de moments mémorables, et je t'en suis reconnaissant. Merci également à Thomas Beuscart : tu parles moins, mais tes mots sont toujours justes. Merci pour nos échanges et bon courage pour ta propre thèse. Durant toutes ces années, j'ai partagé mon bureau avec Jonglôme Bouloncini (ce n'est pas son vrai prénom) et Molkyklom (ce n'est pas son vrai prénom non plus), avec qui nous avons beaucoup trop chanté des tubes des années 80 au lieu de travailler. Merci à vous pour l'ambiance incroyable que vous avez apportée au bureau : vous m'avez motivé chaque matin à venir travailler et, surtout, vous avez supporté toutes mes blagues nulles ! Merci pour tous ces fous rires. Je dois aussi présenter mes excuses à Édouard Rivière-Lorphèvre, l'homme au nom de famille beaucoup trop long pour tenir sur une page de garde sans réduire la taille de la police (vous irez vérifier). Bien que tu aies joué avec des raquettes de tennis en bois, tu n'es pas aussi vieux que je le laisse entendre. Mais je profite de ces excuses pour te remercier pour toutes nos discussions et tes précieux conseils pour ma thèse. Un grand (comme lui) merci à Nutte Matthias, qui m'a permis de réaliser mes premiers essais : sans toi, je n'aurais certainement pas su les mener à bien. Merci également à Margaux Lorenzoni qui, malgré une température corporelle plus basse que la moyenne (c'était elle le chauffage d'appoint), réussit à réchauffer l'ambiance partout où elle passe. Merci pour ta bonne humeur ! Je remercie aussi Laurent Vergari et Pascal Boeckx pour leur soutien technique tout au long de cette thèse. Enfin, ces remerciements commencent à être trop longs. Merci à tous ceux que j'ai pu croiser durant ces années : Hedi Hor (le plus italien des Tunisiens), Julien Bossu (qui n'a étonnamment pas de bosses), Khalil Homrani, Imi Ochana, Richard Foka Tembou, Florian Leroy, Colin Huart, Nathalie Staquet, Lahcen El Hiki, ...

Vous l'aurez compris: au service GMECA, on se fait beaucoup d'amis. Mais heureusement pour eux, tous mes amis ne travaillent pas avec moi. Parmi eux, je tiens particulièrement à remercier Margaux Saray, qui depuis plus de dix ans a été bien plus qu'un soutien : parfois une confidente, parfois... mon agenda (je fais des efforts, promis). Merci de m'avoir accompagné dans tous ces moments et pour ton écoute précieuse. Je ne serais pas là aujourd'hui sans ton aide. Merci également à Elisa Putzeys qui, grâce à son petit grain de folie, illumine chaque discussion. Merci aussi à Gilles Delansnay, Antoine Sion et Julien Mougenot pour ces bons moments et nos vacances, qui m'ont permis de me ressourcer entre deux quadrimestres. Antoine, Gilles, je suis venu à vos soutenances de thèse... vous devez venir à la mienne ! Un grand merci également au sang de la veine, le B, le Benjam. On a grandi ensemble : merci d'avoir toujours été là pour moi. Je

souhaite aussi remercier Maxime, Alice, Florent, Marika et Romain pour tous ces bons moments partagés.

Zoé, nous nous sommes rencontrés alors que j'étais en plein milieu de ma thèse, ce qui n'est pas toujours facile pour démarrer une relation. Merci d'avoir été là, de m'avoir aimé et soutenu si fort. Tu m'as fait voir la vie sous un angle que je ne connaissais pas, et elle est bien plus belle à tes côtés.

Je tiens également à remercier ma grande sœur Alicia, qui a toujours été un modèle pour moi en grandissant. Enfin, je souhaite adresser mes derniers remerciements, et certainement les plus importants, à mes parents : Maryse et Emilio. Merci pour votre éducation et les valeurs que vous m'avez transmises, merci d'avoir été des parents à l'écoute et disponibles depuis mon enfance, ce qui m'a permis de m'épanouir dans la vie. Vous m'avez toujours soutenu et conseillé, vous avez été des piliers sur lesquels j'ai pu m'appuyer en toutes circonstances. Vous m'avez guidé depuis toujours et, sans votre dévouement, cette thèse n'aurait jamais vu le jour. Merci du fond du cœur.

Lorenzo Colantonio

Contents

Contents v

List of Figures xii

List of Tables xx

Nomenclature xxv

1 Introduction to Tool Condition Monitoring 1
- 1.1 Motivation to the Thesis . . . 1
- 1.2 Integration of the Thesis into the Laboratory's Research Activities 4
- 1.3 Terminology Used in Tool Condition Monitoring . . . 5
- 1.4 Objective of the Thesis and Contributions . . . 7
- 1.5 Contents of the Document . . . 8

2 Theoretical Background and Developed Framework 13
- 2.1 Introduction . . . 13
- 2.2 Tool Wear in Machining . . . 13
- 2.3 Definition of Artificial Intelligence and Machine Learning . . . 16
- 2.4 Artificial Neural Network Learning Principle . . . 17
- 2.5 Convolutional Neural Network and their Specific Layers . . . 21
- 2.6 Vision Transformers . . . 23
- 2.7 Explainable Boosting Machines Working Principles . . . 25
- 2.8 Software and Hardware . . . 27
- 2.9 Training Networks Based Model and Overfitting Prevention . . . 30
- 2.10 Conclusions . . . 33

I Machining and Cutting Tool Replacement 35

3 Economic and Environmental Motivation of Tool Condition Monitoring 37
- 3.1 Introduction . . . 37

3.2 Methodology . . . 38
3.2.1 Production Model . . . 38
3.2.2 Simulated Scenarios . . . 43
3.3 Results . . . 44
3.3.1 Optimization of Scenarios . . . 44
3.3.2 Optimal Results Comparison . . . 46
3.4 Conclusions . . . 48

4 State of the Art of the Monitoring or Forecasting of the Cutting Tool Degradation 51
4.1 Introduction . . . 51
4.2 Review Protocol . . . 52
4.2.1 Research Questions . . . 53
4.2.2 Search Process . . . 54
4.2.2.1 Database . . . 54
4.2.2.2 Keywords . . . 55
4.2.3 Inclusion and Exclusion Criteria . . . 56
4.2.4 Article Quality Assessment . . . 57
4.2.5 Data Collection and Data Analysis . . . 58
4.3 Bibliometrics Analysis . . . 58
4.3.1 Distribution in Time . . . 58
4.3.2 Repartition of AI Type . . . 58
4.4 In Depth Analysis . . . 60
4.4.1 ANN . . . 63
4.4.1.1 General Architecture of the Neural Network . . . 63
4.4.1.2 Input . . . 63
4.4.1.3 Pre-Processing . . . 63
4.4.1.4 Output . . . 65
4.4.1.5 Comparison . . . 65
4.4.1.6 Conclusions . . . 66
4.4.2 CNN . . . 66
4.4.2.1 Presentation of the Articles . . . 66
4.4.2.2 Conclusions . . . 67
4.4.3 In Depth Analysis – Conclusions . . . 67
4.5 Discussion . . . 70
4.6 Answers to the Research Questions . . . 71
4.7 Conclusions of the Systematic Literature Review . . . 73
4.8 Literature Evolution since the SLR . . . 76
4.9 Integration of the Literature Findings into the Thesis . . . 77

II Development of Tool Condition Monitoring Methods on Turning Operation 79

5 Introduction to the Development of Tool Condition Monitoring Methods in Turning 81

6 Description of the Turning Database 83

6.1 Introduction . . . 83
6.2 Experimental Setup . . . 84
6.3 Features Extraction . . . 87
6.4 Correlation Analysis and Features Selection . . . 88
6.5 Training and Testing Dataset . . . 91
6.6 Conclusions . . . 91

7 Monitoring the Cutting Tool Degradation from Sensors Signals in Turning 93
7.1 Introduction . . . 93
7.2 Determination of the Best Network Architecture . . . 94
7.3 Results with the ANN Approach . . . 97
7.4 Results with the Confidence Interval Approach . . . 98
7.4.1 Motivation of the Confidence Interval Estimation . . . 98
7.4.2 Bootstrap Approach . . . 98
7.4.3 Results with the Bootstrap approach . . . 101
7.5 Results with Explainable Model . . . 104
7.5.1 Motivation to Explainability . . . 104
7.5.2 Comparison of ANN results with Explainable Model . . . 105
7.5.3 Global Explanation . . . 108
7.5.4 Local Explanation . . . 109
7.6 Conclusions . . . 110

8 Forecasting the Cutting Tool Degradation from Sensors Signals in Turning 113
8.1 Introduction . . . 113
8.2 Methodology . . . 114
8.3 Learning Process . . . 116
8.3.1 Training parameters . . . 116
8.3.2 Censored Learning . . . 116
8.3.3 Network Architecture . . . 117
8.4 Results . . . 119
8.4.1 Best model selection . . . 119
8.4.2 Overview of the Results . . . 120
8.4.3 Detailed Results – Example on one tool degradation trajectory 122
8.5 Conclusions . . . 127

9 Classifying the Cutting Tool State from Tool Images in Turning 129
9.1 Introduction . . . 129
9.2 Classification and Modification of Images . . . 130
9.3 Models for Transfer Learning . . . 134
9.4 Results . . . 136
9.5 Conclusions . . . 140

10 Monitoring the Cutting Tool Degradation from Tool Images in Turning 141
10.1 Introduction . . . 141
10.2 Data Preparation for Image Segmentation . . . 142

10.3 Model for Image Segmentation 143
10.4 Results 145
10.5 Conclusions 147

11 General Conclusions on the Development of Tool Condition Monitoring Strategies in Turning 149

IIIApplication of Tool Condition Monitoring Methods in Milling Operation 151

12 Introduction to the Development of Tool Condition Monitoring Methods in Milling 153

13 Framework for Creating a Milling Database for Tool Condition Monitoring 155
13.1 Introduction 155
13.2 Description of the Experimental Framework 156
13.2.1 Equipment Used During the Tests 157
13.2.2 On-Line and Off-Line Sensors 160
13.2.3 Machining and Data Acquisition procedure 164
13.3 Cutting Conditions 167
13.4 Signal Acquisition and Organisation of the results 170
13.5 Conclusions 174

14 Analysis of the Milling Database 175
14.1 Introduction 175
14.2 Cutting tool's Degradation 176
14.2.1 Flank Wear Degradation Pattern 177
14.2.2 Tip Wear Degradation Pattern 177
14.2.3 Wear Measurement 178
14.2.4 Degradation for Different Cutting Conditions 180
14.3 Roughness and Quality of Surfaces Produced 182
14.4 Evolution of the Cutting Forces Indicators with Tool Wear 184
14.4.1 Cutting Forces – Statistical Indicators from the Time Domain 184
14.4.2 Cutting Forces – Statistical Indicators from the Frequency Domain 185
14.4.3 Cutting Forces – Indicators Correlation with Tool Wear . . 189
14.5 Evolution of the Accelerations Indicators with Tool Wear 190
14.5.1 Accelerations – Statistical Indicators from the Time Domain 191
14.5.2 Accelerations – Statistical Indicators from the Frequency Domain 191
14.5.3 Accelerations – Indicators Correlation with Tool Wear . . . 192
14.6 Evolution of the Acoustic Pressure Indicators with Tool Wear . . . 192
14.6.1 Acoustic Pressure – Statistical Indicators from the Time Domain 193
14.6.2 Acoustic Pressure – Statistical Indicators from the Frequency Domain 193

14.6.3 Acoustic Pressure – Indicators Correlation with Tool Wear 194
14.7 Evolution of the Machine's Power Indicators with Tool Wear . . . 194
14.7.1 Machine's Power – Statistical Indicators from the Time Domain 195
14.7.2 Machine's Power – Statistical Indicators from the Frequency Domain . . . 195
14.7.3 Machine's Power – Indicators Correlation with Tool Wear . 196
14.8 Correlation of All Extracted Features with the Tool Wear . . . 197
14.9 Features Selection . . . 197
14.10 Training and Testing Dataset . . . 201
14.11 Conclusions . . . 203

15 Monitoring the Cutting Tool Degradation from Multiple Sensors Signals in Milling 205
15.1 Introduction . . . 205
15.2 Monitoring Cutting Tool Wear with ANN . . . 206
15.2.1 ANN Model Selection . . . 206
15.2.2 Monitoring Results with ANN . . . 208
15.3 Confidence Interval Estimation with Probabilistic Networks . . . 213
15.4 Explainability of the Monitoring Results with EBMs . . . 217
15.4.1 EBMs Hyperparameters Selection . . . 217
15.4.2 EBMs Results . . . 217
15.4.3 Global Explanation . . . 218
15.5 Conclusions . . . 219

16 Forecasting the Cutting Tool Degradation from Multiple Sensors Signals in Milling 223
16.1 Introduction . . . 223
16.2 Model Parameters and Selection . . . 224
16.3 Forecasting Results . . . 226
16.4 Conclusions . . . 233

17 Classifying the Cutting Tool State from Tool Images in Milling 235
17.1 Introduction . . . 235
17.2 Classification and Modification of Images . . . 236
17.3 Pre-Trained Models for Image Classification . . . 237
17.4 Classification Results . . . 239
17.4.1 Results . . . 239
17.4.2 Explainability of the models . . . 241
17.5 Conclusions . . . 242

18 Monitoring the Cutting Tool Degradation from Tool Images in Milling 245
18.1 Introduction . . . 245
18.2 Image Preparation for Segmentation . . . 246
18.3 Segmentation Models . . . 247
18.4 Results . . . 248
18.4.1 Segmentation Results . . . 248
18.4.2 Evaluating the Wear from Segmentation . . . 249

18.5 Conclusions 251

19 General Conclusions on the Application of Tool Condition Monitoring Strategies in Milling 253

IV Conclusions and Perspectives 255

20 Conclusions of the Thesis and Outlooks 257
20.1 Summary of the Thesis Contribution for Tool Condition Monitoring and Conclusions 257
20.2 Outlooks 261

Bibliography 263

Appendix 284

A Scientific contributions 285
A.1 Journal Papers 285
A.2 Conference Papers 286
A.3 Oral Presentations with Abstract Published or Not 286
A.4 Posters 287

B Surface Roughness of the Cutting Tools 289
B.1 Introduction 289
B.2 Roughness in Side Cutting Edge 289
B.3 Roughness in Face Cutting Edge 290
B.4 Conclusions 291

C Continued State of the Art 293
C.1 Continued In-depth analysis 293
C.1.1 Fuzzy Inference System and ANFIS 293
C.1.1.1 Presentation of the Articles 293
C.1.1.2 Input 293
C.1.1.3 Pre-Processing 294
C.1.1.4 Output 294
C.1.1.5 Comparison 294
C.1.1.6 Conclusions 294
C.1.2 SVM 294
C.1.2.1 Input 294
C.1.2.2 Pre-Processing 296
C.1.2.3 Output 296
C.1.2.4 Comparison and Particular Approaches 296
C.1.2.5 Conclusions 297
C.1.3 SOM 297
C.1.3.1 Input 297
C.1.3.2 Pre-Processing 297
C.1.3.3 Output 297

C.1.3.4 Comparison 299
C.1.3.5 Conclusions 299
C.1.4 HMM 299
C.1.4.1 Presentation of the Articles 300
C.1.4.2 Conclusions 300
C.1.5 CNN 302
C.1.5.1 Presentation of the Articles 302
C.1.5.2 Conclusions 302
C.1.6 Other 303
C.2 Systematic Literature Review - Tables 303

List of Figures

1.1 Source of disturbances of the machining process. Inspired from [3]. . . 2
1.2 Schematic representation of the steps covered in the thesis. 4
1.3 Tool Condition Monitoring strategies. To define a complete TCM strategy, one element must be selected from each of the following categories: data acquisition type, temporal perspective, and the objective of the strategy. 6
1.4 Organisation of the document. The colours of the boxes correspond to the colours of the index tabs placed on the pages throughout this document, with similar colours indicating related topics. 11

2.1 Tool wear standardized measurement for a turning insert. Inspired from [27]. 14
2.2 Milling tool wear measurement. Inspired from [28] 15
2.3 Tool life for a given cutting speed. 16
2.4 General Architecture of an ANN 18
2.5 A neuron connected to the previous layer 18
2.6 Sigmoid, Tanh and ReLU activation functions 19
2.7 Neural network learning principle. The network parameters are initially selected at random. Then, gradient descent computes how to adjust them in order to minimize the error defined by the cost function. . . . 20
2.8 CNN principles. Convolutional and pooling layer are added to fully connected layers. These additional layers aim to extract relevance features from raw data or images. 22
2.9 Convolution applied to an image matrix. The convolution window slides over the image, applying the kernel to compute the resulting values. The window slide is controlled by the stride value. Here the stride is 1. In this example, zero-padding of size 1 is added around the image. . . 23
2.10 Max and average pooling example. A 4×4 convolved feature map from the convolutional layer is divided into 2×2 blocks, based on the kernel size and stride of 2×2. Each colour represents a distinct window. The resulting outputs from both max pooling and average pooling operations are shown. 24

2.11 Working principle of a vision transformer for classification. Inspired from [44] . . . 25
2.12 Pipeline of steps developed in the thesis and the remaining steps to deploy the models in production. . . . 29
2.13 Evolution of learning and validation losses in the case of overfitting or not. When the model is overfitted, the validation loss diverges from the learning loss. . . . 32

3.1 Framework of the simulated model developed in this chapter. The model consists of two main parts: the simulation of workpiece production and the implementation of tool replacement strategies. . . . 39
3.2 Inventory of the inputs, outputs and processes considered in the environmental impact evaluation of the milling operation. Values for CO_2-eq emissions: tool manufacturing [55]; tool coating [56]; electricity from Belgian grid from Electricity maps [57]; workpiece production [58–61]. 40
3.3 Solid carbide tool used for face milling and coated with TiAlN/TiN. . 40
3.4 Cutting tool degradation simulated in the model. The end-of-life is modelled using a Weibull distribution (blue). The orange curve is simulated using a hyperbolic sine (sinh) function. Imperfect monitoring is represented in green and modelled using a normal distribution, illustrating either an overestimation or underestimation of the actual cutting tool wear. . . . 41
3.5 Timeline of all scenarios. T_{Mach}: tool lifetime at a given inspection time. T_R: predefined tool replacement time. $Wear_{Mon}$: monitored tool wear at a given inspection time. $Wear_{Lim}$: predefined tool wear limit. . . . 45
3.6 Results for the different scenarios presented showing the cost and CO_2-eq of the different scenarios. The confidence interval estimation is computed from 10 simulations and the optimal points are highlighted with dots. The optimal scenarios are pointed by a dot. . . . 47
3.7 Cost repartition to produce a good part in the different tool replacement scenarios. The text in the bar correspond to the mean and standard deviation. The deviation is computed on 10 simulations. This deviation is not represented as it is too small to be seen. . . . 48
3.8 CO_2 repartition to produce a good part in the different tool replacement scenarios.The text in the bar correspond to the mean and standard deviation. The Deviation is computed on 10 simulations. This deviation is not represented as it is too small to visualise. . . . 49

4.1 Methodology pursued to identify the 102 articles. . . . 54
4.2 Number of publications per year in turning. The average values are computed over 7 years. . . . 59
4.3 Sorted global repartition by type of AI. . . . 60
4.4 Chronological repartition of the type of AI—Note that some articles present several AI techniques. . . . 61
4.5 General form of an ANN used for tool condition monitoring. . . . 64
4.6 Repartition of inputs for ANN techniques. . . . 64
4.7 Repartition of pre-processing for ANN techniques. . . . 65

4.8 Repartition of outputs for ANN techniques. 66
4.9 Output variables of all the approaches. 72
4.10 Input variables of all the approaches. 73
4.11 Feature extraction techniques of all the approaches. 74
4.12 Evolution of articles about TCM published by year on the *Scopus* database. Data retrieved in September 2025. 76

6.1 Flowchart illustrating the approach developed in this chapter, structured according to the data preparation logic outlined in Chapter 2. 84
6.2 Technical drawing of the C45 steel bar before and after machining during the turning test. 85
6.3 Experimental straight turning setup. The reference frame of the machine is displayed in red. 85
6.4 Example of VB measurement according to the ISO 3685 standard [27] for tool number 14 bar number 5. VB_{max}: 135 µm and VB_{mean}: 78 µm. 87
6.5 Example of raw temporal signal acquired during the turning for the tool number 16 and bar 5. 88
6.6 Example of features extracted with the statistical and frequency analysis. 89
6.7 Correlation analysis on the whole dataset with respect to VB. 90
6.8 Degradation trajectory for tools: 23, 24, 26, 27, 28 and 29. 92

7.1 Summary of the approaches developed in this chapter. 94
7.2 Learning comparison between the best identified model and an uniform network – Note that the "EarlyStopping" is disabled to allow to have 1000 epochs and not be stopped after convergence. 97
7.3 Monitoring: Estimated tool wear on the testing dataset. 99
7.4 Bootstrap approach with 100 randomly initialised identical neural networks. 100
7.5 Quantile-quantile plot and histogram of the estimation of the neural networks. 101
7.6 Estimated tool wear on the testing dataset with a confidence interval estimated with the Bootstrap approach. 102
7.7 Comparison of performances of the EBM model and the ANN model trained and tested on the same dataset. 107
7.8 Global explanation showing the 5 top and 5 last average contribution of all the input of the model. 109
7.9 Global explanation of the selected features F_x RMS and its contribution to estimation of the tool wear with respect to the input value. The graph below shows density of each interval inside the database. 110
7.10 Local explanation of one prediction of the EBM model showing the contribution of each input features to estimate the tool wear for the penultimate wear measurement of trajectory 29 (Figure 7.7). 111

8.1 Summary of the approach developed in this chapter. 114
8.2 Forecasting Principle. Uses the previous points measured in the past to predict the future evolution of the degradation. 115

8.3 Schematic representation of the model used. In entry a matrix containing the timed features is given. In output the model forecast the H previous evolution of the wear of the tool. 115
8.4 Concept of censored learning. It is not possible to evaluate the error outside the range of measurement. Nonetheless, some prediction overlap evaluable and inevaluable part of the dataset. 117
8.5 Schematic representation of the architecture of the RNN, LSTM and GRU neurons (or units). 118
8.6 Global network architecture layout. The recurrent component consists of multiple layers of RNN, LSTM, or GRU units, optionally followed by stacked fully connected layers. 119
8.7 Comparison of performance for different architectures. The lower MAPE values indicates more accurate results. 120
8.8 Violin plot showing the distribution of error between the real RUL and the predicted one. A mean value (represented by the white dot) closer to 0 indicates better results. The shape of the distribution also indicates how it is spread around the mean value The upper line indicates the third quartile, the middle line with the white dot shows the mean value, and the lower line represents the first quartile. 121
8.9 Violin plot of error for the different time horizon of the best model. . . 122
8.10 Violin plot of the error of the estimated RUL with the best model. . . 123
8.11 Degradation trajectory of Tool 28 with produced bars. 124
8.12 Different stages of wear forecast during the life of the tool 28. The blue line represents the actual degradation, the light-green line shows the previously estimated wear by the model, and the dashed line indicates the future evolution of the tool. The orange line illustrates the network's prediction for the next 3 minutes of machining. Finally, the orange area represents the error between the prediction and the forecasted degradation. 125
8.13 Summary of the prediction of the wear divided by horizon of prediction. 126
8.14 Zoom on the Remaining Useful Life predicted by the model when its forecasting crosses the limit of 200 µm for VB. 126

9.1 Summary of the approach developed in this chapter. 130
9.2 Image of an insert (VB: 280 µm). 131
9.3 Classification of the tool degradation into 3 classes. 132
9.4 Image transformations applied to the image from Figure 9.2. 133
9.5 Transfer learning principle: A given architecture is reused for a different task. The model's architecture and weights are transferred to the new task, where the output layer is adapted to suit the new objective. . . . 136
9.6 Confusion matrices for the three tested models. The values are computed based on the results presented in Table 9.2 137
9.7 Classification of the tool degradation into 3 classes with transition zone between the classes. 138
9.8 Grad-CAM attention maps from VGG19 classification. The left image shows correct focus on the wear zone, while the right image illustrates misclassification due to altered contrast. 139

10.1 Segmentation : Tool wear and its degradation detected manually. . . . 142
10.2 Summary of the approach developed in this chapter. 142
10.3 Image segmentation with one hot encoding principle. The segmentation map is divided into multiple channels, each corresponding to a given class. In the channel decomposition of the segmentation map, colours correspond to value of 1 while black correspond to value of 0. 144
10.4 Schematic representation of an encoder-decoder architecture. The encoder part use convolution and pooling layer to reduce the dimension into a latent space. The decoder uses this latent space with deconvolution and unpooling layer to reconstruct the output segmentation map. 145
10.5 Example of prediction obtained by the U-Net++ model. From left to right: Image given to the model, expected segmentation map and map produced by the model. 147

13.1 Overview of the experimental setup. The three acquisition means are shown along with their respective measured phenomena and the position of the components within the machine. 157
13.2 Machine used during for the experimental milling campaign: GF MIKRON VCE 600 Pro. 158
13.3 Sacrificial and control workpiece mounting assembly. 158
13.4 Selected cutting tool to perform the experimental milling campaign [62]. 159
13.5 Alicona Infinite Focus G6 and the 3D reconstruction of a new cutting tool. 160
13.6 Geometry of the cutting edge radii for the cutting tool. 161
13.7 Kistler 9257B and its assembly with the sacrificial piece during machining. The sensor frame differs from the one of the machine. 162
13.8 Dytran 3263A2 fixed on the closest flat surface on the machine. The sensor frame differs from the one of the machine. 163
13.9 Microphone oriented toward the centre of the machine. 163
13.10Microscope used to take picture of the cutting tool. In this configuration, an image of the tip of the tool is acquired. 164
13.11Example of image acquired with the microscope. For the tool 0020 tooth number 2. 165
13.12Keyence VR-6000 profilometer inspecting the surface roughness of a control piece. 165
13.13Overview of the machining procedure divided into three steps: first, machining the sacrificial workpiece while recording sensor data; second, machining the control workpiece without any data acquisition; and finally, inspecting the cutting tool under a microscope. This procedure is repeated until the tool reaches its end of life. 166
13.14First step of the experimental campaign: ZIG machining. In blue, the cutting tool is in the material at the speed of the determined feed rate. In green, the cutting tool move at its maximum velocity as it is outside material. Example with the machining of the layer #5 with only the 2 first pass out of 15 represented. 168

13.15Second phase of the experimental campaign. The tool machines the control workpiece at the position associated from the layer it machined in the first phase. Example from the layer number 5 associated with the Figure 13.14. The removed material shows how the material is removed from the 4 differents passes. 169
13.16Architecture of the directory tree for organizing data. 171
13.17Architecture of the directory tree for organizing data of tool 0010. . . 172

14.1 Flowchart illustrating the approach developed in this chapter, structured according to the data preparation logic outlined in Chapter 2. 176
14.2 Image of the side cutting edges of a new and a worn cutting tool acquired using an Alicona Infinite Focus G6 (see Figure 13.5a). 177
14.3 Flank wear of the cutting tool 178
14.4 Crater wear of the worn cutting tool 0020 178
14.5 Example of tip images acquired during the experimental campaign. . . 179
14.6 Wear measurement for tool 0020 layer 010. Evaluation zone length is 3 mm. 180
14.7 Superimposed degradation of the cutting tools for different cutting conditions. 181
14.8 Degradation of cutting tools for fixed feed per tooth. 181
14.9 Example of surfaces produced by tool 0020 throughout its lifetime. The tool's lifespan is represented from left to right, with the left side corresponding to a new tool and the right side to a worn tool. 182
14.10Arithmetic roughness in function of machining time for all cutting conditions. 183
14.11Mean cutting force evolution in function of the flank wear for different feed conditions. 184
14.12Cutting force evolution for a new and worn tool (Tool: 0020). The highlighted area show the selected signal on which the frequency analysis will be presented. 185
14.13Discrete Fourier Transform of the cutting force of tool 0020. Comparison at the beginning of its life and at the end. 186
14.14Power Spectral Density comparison between tool 0020. Comparison at the beginning of its life and at the end. 187
14.15Spectral centroid of the cutting force evolution in function of the flank wear for different feed conditions. 190
14.16Acceleration in the cutting direction evolution in function of the flank wear for different feed conditions. 191
14.17HNR indicator of the acceleration in the cutting direction evolution in function of the flank wear for different feed conditions. 192
14.18Sound level in the machine in function of the flank wear for different feed conditions. The oscillating pattern, illustrated for one cutting condition (Figure 14.18a), is representative and occurs across all tested conditions. 193
14.19HNR of sound in the machine in function of the flank wear for different feed conditions. 194
14.20Spindle power in function of the flank wear for different feed conditions. 196

14.21Spectral entropy for the spindle power in function of the flank wear for different feed conditions. 196
14.22Correlation analysis of the different measured signals with respect to tool wear. 198
14.22...*Continued* 199
14.23Feature selection process. First, the RFECV method is applied to determine the optimal number of features. Once this number is known, multiple selection methods are used to identify the most relevant feature combinations. 200
14.24Division of the dataset into training and testing subsets based on the 9 distinct cutting conditions. The learning process is repeated 9 times, with each iteration using a different cutting condition as the test set. This ensures that the model is evaluated across all conditions, providing a complete assessment of its performance. 203

15.1 Summary of the approaches developed in this chapter. 206
15.2 Architecture of the optimization scheme used to select the most suitable family of neural network models. 207
15.3 Selected architecture of the monitoring neural network for the milling database. 208
15.4 Comparison of different inputs on the monitoring capabilities for each sensor. The monitored tool is the tool 0020. 212
15.4 ...*Continued* 213
15.5 Monitoring capabilities with sensor fusion. The monitored tool is the tool 0020. 214
15.6 Selected architecture of the monitoring probabilistic neural network for the milling database. Compared to the Figure 15.3, this network have 2 outputs: one for μ and one for σ. 215
15.7 Comparison of different monitored trajectory with a probabilistic network. 216
15.8 Global explanation showing the sorted average contribution of the 40 most important inputs ranked by the EBM model. 219
15.9 Global explanation of the selected features “cumulative duration” and its contribution to estimation of the tool wear with respect to the input value. The graph below shows density of each interval inside the database. 220
15.10Global explanation of the selected features F_y Mean and its contribution to estimation of the tool wear with respect to the input value. The graph below shows density of each interval inside the database. 220

16.1 Summary of the approach developed in this chapter. 224
16.2 Layout of the recurrent architectures tested during the optimization process. 226
16.3 Comparison of different inputs on the forecasting capabilities for each sensor. The monitored tool is the tool 0020. The colour of each prediction curve reflects the time at which the forecast was made: the earlier the prediction, the lighter the curve. 230
16.3 ...*Continued* 231

16.4 Comparison of different sensor fusion inputs on the forecasting capabilities. The monitored tool is the tool 0020. 232

17.1 Summary of the approach developed in this chapter. 235
17.2 Cropped image from the database to focus on the wear zone. The resolution is reduced to the models input size (224 pixels by 224 pixels). 237
17.3 Maximum image modifications applied to image from Figure 17.2. . . 238
17.4 Confusion matrices for the three tested models. Each value represents the cumulative results over 10 independent runs of the same model. . 240
17.5 XGrad-CAM attention map for VGG19 classification with the original image and the same image with a decrease in contrast. Warmer colours indicate stronger attention. 242

18.1 Summary of the approach developed in this chapter. 246
18.2 Example of flank and tip images from the database with their associated mask. The images are in the model's input resolution (256 by 256 pixels). 247
18.3 Comparison of the predictions from three models on the same flank wear image. Green areas indicate correctly predicted regions (intersection between ground truth and prediction). Red areas represent false positives (predicted regions not present in the ground truth), while blue areas highlight false negatives (ground truth regions missed by the prediction). 249
18.4 Comparison of the predictions from three models on the same tip wear image. Green areas indicate correctly predicted regions (intersection between ground truth and prediction). Red areas represent false positives (predicted regions not present in the ground truth), while blue areas highlight false negatives (ground truth regions missed by the prediction). 250
18.5 Example of flank wear measurement based on the ground truth mask and the predicted masks. 251

20.1 Schematic overview of all topics addressed in the thesis and their key conclusions. 260

B.1 Definition of the face and side cutting edges. 290
B.2 Arithmetic roughness (R_a) for the side cutting edges of the cutting tools. 290
B.3 Arithmetic roughness (R_a) for the face cutting edges of the cutting tools. 291

C.1 Repartition of inputs for fuzzy techniques. 295
C.2 Repartition of pre-processing for fuzzy techniques. 295
C.3 Repartition of outputs for fuzzy techniques. 296
C.4 Repartition of SVM inputs. 298
C.5 Repartition of SVM pre-processing. 298
C.6 Repartition of SVM outputs. 299
C.7 Repartition of SOM inputs. 300
C.8 Repartition of SOM pre-processing. 301
C.9 Repartition of SOM outputs. 301

List of Tables

1.1 Relation between the maintenance concepts and the terminology used in TCM. 7
1.2 Monitoring and forecasting approaches developed in the thesis. D: Developed during the thesis. NC: Not considered during the thesis. Approaches marked with * were developed but are not described in this document. Approaches marked with P have been published, and those marked with FP are currently being prepared for a future publication in academic papers. 8

2.1 Overview of the chapter structure, with each section linked to the chapter(s) where its concept is applied. 13
2.2 Activation functions and associated equations 19
2.3 Main softwares used to obtain the results in the thesis 28
2.4 Hardware used to obtain the results in the thesis with their computing capabilities 30

3.1 Model parameters of the cutting tool. 41
3.2 Model parameters of the workpiece. 42
3.3 Model parameters of the machine. 43
3.4 Optimal results for each scenario over a one-month production simulation. Standard deviations are calculated based on 10 repeated simulations under identical conditions. Percentages are calculated in relation to the average value obtained in scenario 1. 47

4.1 List of keywords. Wildcard character (*) includes these different forms of the “tool” word: “tool”, “tools”, and “tooling”. 56
4.2 Comparison of the three most used: input, pre-processing and output for each approaches. 69

6.1 Chemical Composition of C45 Steel (EN 1.0503) [125]. 86
6.2 Cutting conditions for each turning test. All tests were performed on C45 steel, with a constant feed of $0.2\,\mathrm{mm\,tooth^{-1}}$ and a depth of cut of 1 mm. 86

6.3 Measured parameters and their sensor during the turning tests. 87
6.4 The 5 statistical features processing methods and their mathematical equations. 88
6.5 Most correlated features with respect to the tool wear. 91

7.1 Comparison of the relative performance of several neural network architectures after a maximum of 1,000 training epochs, sorted by performance. The best architecture is highlighted in bold. 96
7.2 Hyperparameter of the EBM model with the recommended value by the InterpretML library [148] and the value selected through a grid search algorithm. 106
7.3 Comparison of performance metrics from the trajectories presented in Figure 7.7. 108

8.1 Model and learning parameters of the forecasting models. 116

9.1 Number of parameters in the pre-trained components of the selected models. The GFLOPS values indicate their required computational resources. 135
9.2 Comparison of accuracy for the different methods segmented by class and the type of image modification applied. C1 corresponds to Class 1, C2 corresponds to Class 2 and C3 corresponds to Class 3. 137
9.3 Comparison of the position of errors for the different architectures. A transition between classes is added. 138

10.1 Number of parameters of the different models. 145
10.2 Comparison of IoU accuracy across different architectures under various image modifications. Each value represents the mean IoU over 10 independent runs of the same model. 146

13.1 Specifications of the cutting tool. 159
13.2 Nominal and determined cutting condition to perform the test. 169
13.3 Cutting conditions modified during the tests, with three associated levels. 170
13.4 All combination of cutting conditions tested during the experimental test. 170
13.5 Summary of the three acquisition chains, including the number of measured channels and their respective sampling frequencies. 171
13.6 Comparison of the turning and milling database. 174

14.1 Surface roughness classes according to ISO 21920 [176]. 183
14.2 Cutting force temporal indicators correlation with the wear. 190
14.3 Cutting force frequency indicators correlation with the wear. 190
14.4 Acceleration in the cutting direction temporal indicators correlation with the wear. 192
14.5 Acceleration in the cutting direction frequency indicators correlation with the wear. 192
14.6 Sound temporal indicators correlation with the wear. 195
14.7 Sound frequency indicators correlation with the wear. 195
14.8 Spinde power temporal indicators correlation with the wear. 197

14.9 Spinde power frequency indicators correlation with the wear. 197
14.10Feature selection sets by category. Note that the notation are expressed in the sensor frame. 202

15.1 List of hyperparameters optimized using the Optuna optimizer. For each hyperparameter, the tested value range is presented along with the optimal result obtained. The selected value within this optimal range is also indicated in brackets. 207
15.2 R^2 score and mean absolute error for each combination of input features (as described in Table 14.10). Each value represents the mean R^2 score or the mean absolute error and its standard deviation over 10 independent runs of the same model. The performance indicators are computed on the whole dataset. 210
15.3 Hyperparameter of the EBM model with the recommended value by the InterpretML library [148] and the value selected through a grid search algorithm. 217
15.4 R^2 score and mean absolute error for each combination of input features (as described in Table 14.10). The performance indicators are computed on the whole dataset. 218

16.1 Model parameters of the forecasting models. 224
16.2 List of hyperparameters optimized using the Optuna optimizer. For each hyperparameter, the tested value range is presented along with the optimal result obtained. The selected value within this optimal range is also indicated in brackets. 225
16.3 R^2 score for each combination of input features (as described in Table 14.10). Each value represents the mean R^2 score and its standard deviation over 10 independent runs of the same model. The R^2 score is computed on the whole dataset. 228
16.4 Mean absolute error for each combination of input features (as described in Table 14.10). Each value represents the mean absolute error and its standard deviation over 10 independent runs of the same model. The mean absolute error is computed on the whole dataset. 229
16.5 Remaining Useful Life (RUL) mean absolute error for each combination of input features (as described in Table 14.10). The RUL is evaluated for wear reaching 200 µm. Each value represents the mean absolute error and its standard deviation over 10 independent runs of the same model. The mean absolute error is computed on the whole dataset. . . 233

17.1 Comparison of classification accuracy across different architectures under various image modifications. Each value represents the mean accuracy over 10 independent runs of the same model. 240
17.2 Position of errors in the classification. More errors in the transition have less impact in real milling application. 241

18.1 Comparison of IoU accuracy across different architectures under various image modifications. Each value represents the mean IoU over 10 independent runs of the same model. 248

18.2 Mean error to estimate the cutting flank wear. Each value represents the mean error over 10 independent runs of the same model. The mean target value of VB_{max} is 338.9 µm and VB_{mean} is 166.9 µm. 251

C.1 General summary of ANN-related publications. 304
C.2 General summary of fuzzy logic related publications. 311
C.3 General summary of neuro-fuzzy related publications. 314
C.4 General summary of SVM-related publications. 319
C.5 General summary of SOM-related publications. 325
C.6 General summary of HMM-related publications. 327
C.7 General summary of CNN-related publications. 328
C.8 General summary of publications related to various other methods. . . 329

Nomenclature

Acronyms

AE Acoustic Emission

AI Artificial Intelligence

ANFIS Adaptative Neuro Fuzzy Inference System

ANN Artificial Neural Network

ARIMA Auto-Regressive Integrated Moving Average

CI Confidence Interval

CNC Computer Numerical Control

CNN Convolutional Neural Network

DB Database

DOI Digital Object Identifier

DU-Net Double U-Net

EBM Explainable Boosting Machine

EC Exclusion Criteria

EOL End Of Life

FDA Functionnal Data Analysis

FFT Fast Fourier Transform

FIS Fuzzy Inference System

FMECA Failure Mode Effects and Criticality Analysis

GAM General Additive Model

GDP Gross Domestic Product

GFLOPS Giga Floating Point Operations Per Second

Grad-CAM Gradient-weighted Class Activation Mapping

GRU Gated Recurrent Unit

HMM Hidden Markov Model

HSV Hue Saturation Value

IC Inclusion Criteria

IoU Intersection over Union

LSTM Long-Short Term Memory

MAPE Mean Absolute Percentage Error

ML Machine Learning

MLP Multi Layer Perceptron

MSE Mean Squared Error

MTTF Mean Time To Failure

NLP Natural Language Processing

PRISMA Preferred Reporting Items for Systematic Reviews and Meta-Analyses

PSD Power Spectral Density

PVD Physical Vapour Deposition

QA Quality Assessment

ReLU Rectified Linear Unit

RMS Root Mean Square

RNN Recurrent Neural Network

RPM Revolutions Per Minute

RQ Research Question

RUL Remaining Useful Life

SLR Systematic Literature Review

SOM Self Organizing Map

SVM Support Vector Machine

Tanh Hyperbolic Tangent

TCM Tool Condition Monitoring

TFLOPS Tera Floating Point Operations Per Second

ViT Vision Transformer

W.r.t. With respect to

Symbols

α	Learning rate	
$\bar{t}$	Average of the temporal signal	
$\bar{t}$	Mean value of all t_i	
$\bar{y_p}$	The average estimation of the tool wear	[µm]
$\bar{y}$	mean value of the y_i	
β	Weibull's shape parameter	
β_0	Baseline or intercept value in EBM models	
η	Weibull's scale parameter	
Γ	Gamma function	
κ	Cohen's Kappa statistic	
μ	Mean value	
σ	Standard deviation	
$\widehat{y_i}$	Output of the i^{th} neuron	
$\widehat{y_{pi}}$	The i^{th} estimation of the tool wear	[µm]
$\widehat{y}$	Output of a model	
a_e	Radial depth of cut	[mm]
a_p	Axial depth of cut	[mm]
a_w	Intensity of the derivative	
b	Bias of a neuron	
b_w	Position of the inflection point	
C	Cost function	
c_w	Proportional parameter	
F	Cutting force components, with possible subscripts	

- F_x Cutting force along axis x

- F_y Cutting force along axis y
- F_z Cutting force along axis z

[N]

$f_i(x_i)$	Contribution function of the variable xi in EBM models	
f_z	Feed per tooth	$[\mathrm{mm\,tooth^{-1}}]$
f_t	Tooth Passing Frequency	[Hz]
M	Cutting torque components, with possible subscripts • M_x Cutting torque along axis x • M_y Cutting torque along axis y • M_z Cutting torque along axis z	[N m]
m_i	Central moment	
N	Spindle Rotation	$[\mathrm{rev\,min^{-1}}]$
N_t	Number of teeth of a milling cutting tool	
P_e	Probability of random agreements	
P_o	Observed probability	
R_a	Arithmetic average roughness	[µm]
SS_{res}	Sum of squares of the residual errors	
SS_{tot}	Total sum of squares	
t_i	Temporal signal i^{th} sample	
T_R	Cutting tool replacement time	[min]
T_{mach}	Cutting tool lifetime or machining duration	[min]
v_c	Cutting speed	$[\mathrm{m\,min^{-1}}]$
v_f	Feed rate	$[\mathrm{mm\,min^{-1}}]$
w_i	Weight of i^{th} neuron	
y	Real output of a model	
Z	Sum of inputs of a neuron	
FN	False negative value	
FP	False positive value	

H	Horizon of prediction of forecasting models	
P	Past value used by forecasting models	
TN	True negative value	
TP	True positive value	
VB	Mean Cutting Tool Flank Wear • Also denoted as VB_{mean} to avoid confusion with VB_{max}	[µm]
VB_{max}	Maximum Cutting Tool Flank Wear	[µm]

Chapter 1

Introduction to Tool Condition Monitoring

1.1 Motivation to the Thesis

The manufacturing sector is a vital part of global industry, representing around 15 % of worldwide Gross Domestic Product (GDP) [1, 2]. The fundamental objective of manufacturing is to produce the required quantity of workpieces with specified characteristics in the most cost-efficient way [3]. Within manufacturing, machining is a critical subset that consists of removing material thanks to a cutting tool. This process is essential across many industries because it enables the production of high-value, high-precision components that cannot be achieved through other manufacturing methods [4]. Therefore, machined parts are often complex and have specifications that are challenging to meet.

Machining is most commonly performed using Computer Numerical Control (CNC) machines, which guide a cutting tool through the material to shape the desired part. These machine tools are complex systems, and maintaining them is a challenging task. Poor management of maintenance operations can lead to machine failures and the breakdown of critical subsystems [5]. Moreover, maintenance represents a significant operational cost, adding complexity to the overall exploitation of these machines. To better understand the origin of these costs, a previous thesis conducted a Failure Mode, Effects, and Criticality Analysis (FMECA) on a CNC machine used in an oil pipe threading factory [6]. The analysis identified two categories of maintenance-related costs: those associated with the machine itself and those related to the cutting tools. The latter accounting for up to 50 % of the total maintenance cost. Maintenance of machine-related equipment is well-documented and widely understood in the literature, often addressed through probabilistic approaches [6]. Therefore, among the two categories of maintenance costs identified, this thesis will focus on those related to cutting tools. As such, the machine will be considered free from defects, allowing the study to focus exclusively

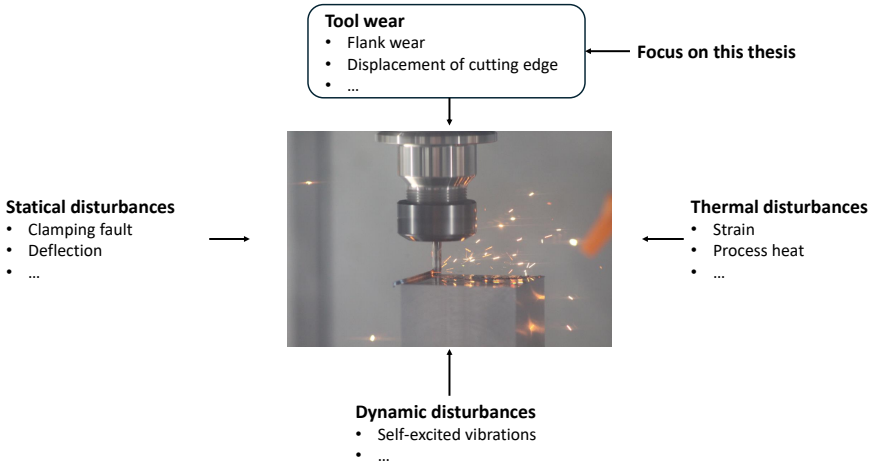


Figure 1.1: Source of disturbances of the machining process. Inspired from [3].

on the cutting tool. The tool that perform the machining process has its own sources of disturbances that can affect part production, as illustrated in Figure 1.1. These disturbances may be external or internal, and can originate from static, dynamic, or thermal sources, as well as directly from cutting tool wear. In this thesis, the primary focus will be on cutting tool wear, a major internal disturbance that significantly impacts machining performance and product quality.

Tool wear results from the interaction between the cutting tool and the workpiece material. Several wear mechanisms can occur during machining, often acting simultaneously and influencing the process in complex ways [7, 8]. This degradation alters the interface between the tool and the workpiece such that it becomes unable to produce parts that meet the required specifications. Multiple strategies exist to extend tool life, but they often involve compromises between ecological, economical, and practical considerations [9–11]. Despite these efforts, the tool will eventually become worn and need to be replaced.

Effective management of tool replacement has a significant impact on productivity, machining costs, and environmental impact (see Chapter 3). But tool replacement is a complex issue. While replacing a worn tool helps prevent the production of defective parts, excessive replacement increases tooling expenses, raises environmental emissions, and reduces overall profitability. Striking a balance between maintaining product quality, minimizing tool replacement costs, and avoiding unexpected tool failures is a challenging task [12]. Moreover, tool wear is neither stationary nor fully predictable. This was first proven experimentally in 1971, that the tool life is not deterministic and must be represented by statistical variable [13]. This means that two identical tools may exhibit significantly different lifespans, further complicating replacement strategies. As a result, manufacturers often adopt conservative safety margins, which lead to increased tool consumption and reduced productivity [14]. In practice, tools are typically used only until 50 % to 80 % of

their maximum degradation [14, 15]. Given that tooling can account for 2 % to 30 % of production costs, there remains considerable potential for optimization [14, 16].

There are numerous strategies for tool replacement. Some rely solely on the operator's experience to decide when a tool should be replaced [17, 18]. While this approach is inherently imprecise, it is simple to implement. Other methods treat tool degradation as a deterministic process, linking tool life directly to cutting conditions. These models estimate tool lifespan using empirical laws, such as Taylor's law, which relates tool wear to cutting speed [14]. However, these approaches are limited by the assumption that tool life is deterministic, which does not reflect the actual statistical nature of tool degradation [13]. To address this, probabilistic models have been developed. Notably, the preceding thesis to this work employed Cox proportional hazards models to better capture the variability in tool wear [6]. Although these methods outperform earlier approaches, the rise of Industry 4.0 and the increasing availability of real-time machining data have shifted the focus of recent research toward artificial intelligence-based methods [19]. These techniques enable real-time, case-specific monitoring of tool degradation, allowing decisions to be made based on observed wear rather than relying solely on probabilistic models. This leads to more accurate and adaptive tool replacement strategies tailored to each tool's actual condition and usage.

This thesis therefore takes places within the context of tool condition monitoring (TCM) in machining. It primarily focuses on experimental data collected from two manufacturing processes: turning and milling. The aim is to investigate various TCM approaches for tracking the degradation of cutting tools. Specifically, the objective is to develop Artificial Intelligence (AI) models capable of accurately estimating and forecasting tool condition based on signals acquired during machining. These models seek to address existing gaps in the literature (see Chapter 4) and contribute to more effective monitoring strategies. The industrial relevance of this thesis lies in its evaluation of TCM solutions designed to optimize tool monitoring strategies. By improving the accuracy of tool condition assessment, these approaches help reduce both the economic and environmental costs associated with premature or inappropriate tool replacement resulting from unreliable monitoring methods (see Chapter 3).

Figure 1.2 shows the different steps from the machining process to the decision to replace the tool. This figure highlights the focus of the thesis. The diagram first shows the machining process, which can be either turning or milling. This machining process uses a tool that will wear over time. During machining, signals from various sensors, such as cutting forces and accelerations, are measured. These signals are then used by TCM techniques to determine the tool's condition. The thesis focuses on these acquisition and TCM technique steps to obtain the results. The final step, which involves making the decision to replace the tool, is not discussed in depth in the thesis. Indeed, this decision depends on the machining objective and therefore on the parts being produced. In this thesis, it is assumed that the produced part has no specific requirements, and the standards associated with tool wear will be followed to define the tool's end of life. This part will therefore not be addressed in detail in this thesis. Moreover, the relationship between tool condition and machining quality is not linear, and is therefore not

considered within the scope of this work.

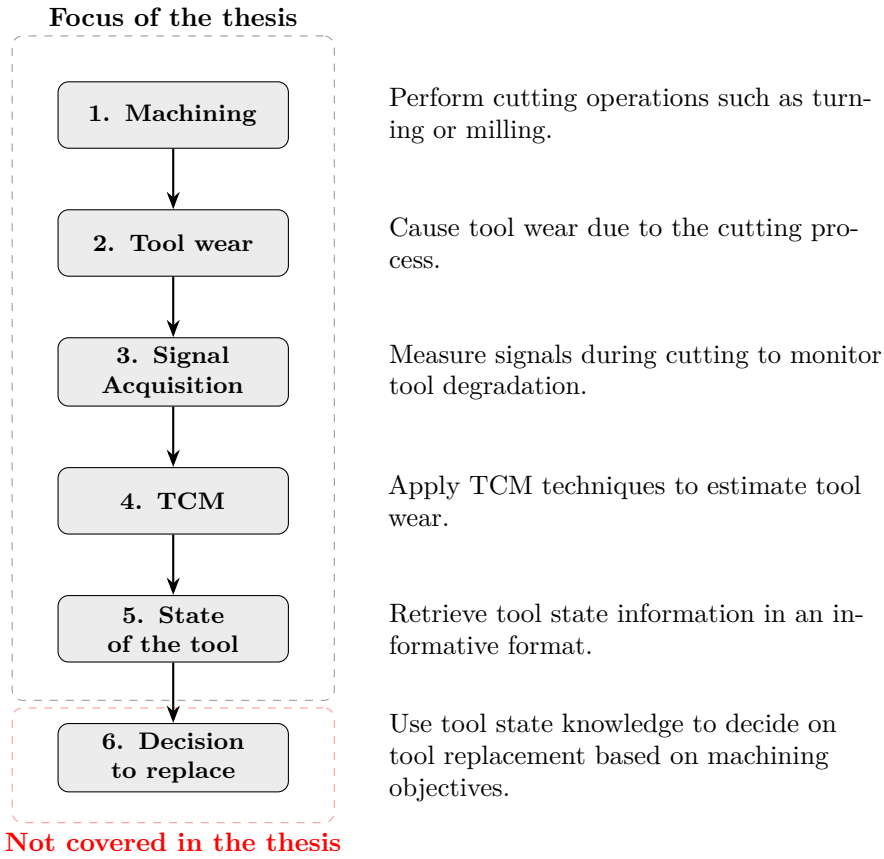


Figure 1.2: Schematic representation of the steps covered in the thesis.

1.2 Integration of the Thesis into the Laboratory's Research Activities

This thesis is part of a research effort spanning over 20 years within our laboratory, focused on the maintenance of industrial equipment. It is important to highlight some of the key contributions that have guided the lab toward the topic of this thesis.

In 2008, Bovic Kilundu's work enabled the diagnosis of rotating machinery using neural networks [20]. This was followed by Olivier Basile's doctoral research, which focused on the reliability of equipment subjected to variable loads [21]. A few years later, Christophe Letot contributed with research on predictive maintenance of industrial equipment, based on modeling, estimation, and simulation of degradation laws. His thesis also introduced several stochastic models [22]. Building on these foundations, the previous thesis by Lucas Equeter applied stochastic approaches,

specifically Cox models, to monitor and predict the degradation of cutting tools [17].

This thesis continues in the same line of research, with a dual focus on both diagnostics and prognostics in maintenance. The application case involving cutting tools, initiated by Lucas Equeter, is further developed here using artificial intelligence methods, similar to those applied to rotating machinery several years ago.

1.3 Terminology Used in Tool Condition Monitoring

Throughout this document, a specific terminology related to tool condition monitoring (TCM) will be used to categorize the results and chapters. This terminology consists of three key words that, when combined, define a TCM strategy. Each word represents a distinct aspect of the strategy. To build a complete strategy, one word must be chosen from each of the following categories (Figure 1.3):

1. **Data Acquisition Type**. This first word refers to how data is collected in relation to the tool. It describes the type of signal being used. There are two main approaches:
   - **Direct**: Data is collected directly from the tool, for example through an image of the tool itself.
   - **Indirect**: Data is collected through a sensor or information source that is not directly from the tool. For instance, using a force sensor to gather data.
2. **Temporal Perspective**. The second word refers to the time-based aspect of the strategy used to assess the tool's condition. Again, two approaches are distinguished:
   - **Monitoring**: This involves using data at a specific moment (time T) to assess the tool's condition at that same moment. The focus is on the current state of the tool.
   - **Forecasting**: This approach estimates the tool's condition both at the current time (T) and at a future point (T+H) over a prediction horizon H. It goes beyond assessing the present state to also predict how the tool degradation will evolve over time.
3. **Objective of the Strategy**. The third word defines the goal of the strategy. Two main objectives are identified:
   - **Classification**: The aim is to categorize the tool's condition into predefined classes, such as "new", "moderately worn", or "worn". The focus is on determining which class the tool belongs to.
   - **Regression**: The goal is to predict a continuous numerical value representing the tool's wear level (e.g., VB indicator see Chapter 2, Section 2.2). This provides a precise, quantitative assessment of the tool's condition.

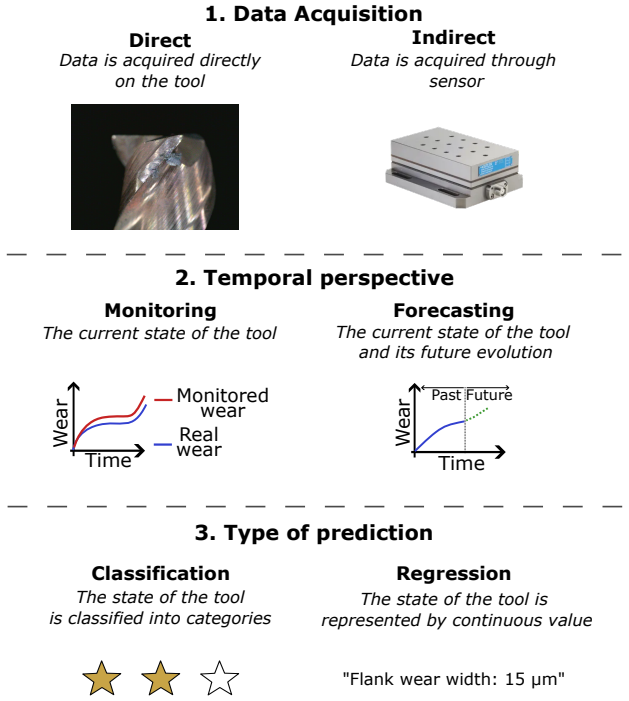


Figure 1.3: Tool Condition Monitoring strategies. To define a complete TCM strategy, one element must be selected from each of the following categories: data acquisition type, temporal perspective, and the objective of the strategy.

As example:

$$\underbrace{\text{Indirect}}_{\text{Acquisition Type}} \quad \underbrace{\text{Monitoring}}_{\text{Temporal Perspective}} \quad \underbrace{\text{Regression}}_{\text{Objective}}$$

Here, "indirect" refers to the use of sensors to assess tool degradation, "monitoring" indicates that the tool's condition will be estimated at a specific moment in time (T), and "regression" means that the model will provide an estimate of the tool's wear level.

This topic and the associated strategies are not exclusive to machining. In fact, these concepts originate from industrial maintenance, which defines similar principles, forming the foundation from which TCM strategies are derived. In the context of maintenance, four key concepts are commonly used to describe the management of equipment [23]:

- **Information**: the collection of data regarding the condition of equipment.
- **Detection**: the identification of an anomaly or an ongoing failure.
- **Diagnosis**: the determination of the root cause of the anomaly or failure.

- **Prognosis**: the estimation of the probable time of failure or the remaining useful life.

The link between these maintenance concepts and TCM is illustrated in Table 1.1. All previously defined strategies are represented there, but framed within the formalism of maintenance. It should be noted that, in the context of maintenance, prognosis generally refers to the estimation of the Remaining Useful Life (RUL) of equipment. In this thesis, however, prognosis is used to forecast the condition of the cutting tool. As a result, the RUL is not directly predicted by the models. Instead, it is derived from the forecasted degradation of the cutting tool (see Chapters 8 and 16. Throughout this thesis, the TCM formalism will be used as the primary framework for analysis and discussion.

Table 1.1: Relation between the maintenance concepts and the terminology used in TCM.

Maintenance Concept	**Associated TCM Approach**
Information	Collect images (**Direct**) or sensor signals (**Indirect**) that reflects the condition of the tool.
Detection	Detect wear from images (**Direct**) or from sensor signals (**Indirect**).
Diagnosis (**Monitoring**)	Identify the amount of wear and explain the cause. It is either represented as a class (**Classification**) or a value (**Regression**).
Prognosis (**Forecasting**)	Estimate the remaining tool life.

1.4 Objective of the Thesis and Contributions

TCM is a broad field encompassing a variety of strategies. The objective of this thesis is to explore these strategies in order to address gaps identified in the existing literature (see Chapter 4). Table 1.2 presents all the approaches developed throughout this work. The table is divided into two main sections: one focused on monitoring (or diagnosis) and the other on forecasting (or prognosis).

For the monitoring aspect (see Table 1.2a), both classification and regression methods were developed under direct and indirect approaches. Each of these methods has resulted in either published papers or manuscripts currently in preparation. The complete list of publications is presented in the Appendix of this document (see Appendix A).

Regarding forecasting (see Table 1.2b), the thesis primarily focused on the indirect regression approach. In fact, the forecasting methodology itself represents a novel contribution to the existing literature. This novelty enabled a strategic selection among the various available approaches, with priority given to those offering the greatest industrial relevance. In particular, direct forecasting approaches were not pursued in this work. This decision stems from the fact that forecasting

tool wear evolution directly from images requires very large datasets to train reliable models. This kind of dataset was not available during the majority of the thesis. Therefore, indirect approaches were favoured. Among the indirect methods, regression was prioritized. A regression model capable of estimating the tool's condition can also be used to classify its state, making it a versatile and industrially valuable solution. Consequently, the research effort was dedicated exclusively to developing the indirect forecasting regression approach, which is both the most general and the most promising from an industrial perspective.

The list of all publications made during the thesis is available in the Appendix A.

Table 1.2: Monitoring and forecasting approaches developed in the thesis. D: Developed during the thesis. NC: Not considered during the thesis. Approaches marked with * were developed but are not described in this document. Approaches marked with ^{P} have been published, and those marked with ^{FP} are currently being prepared for a future publication in academic papers.

(a) Monitoring approaches developed in the thesis.

Monitoring		
	Direct	Indirect
Classification	D^{P}	D^{*P}
Regression	D^{FP}	D^{P}

(b) Forecasting approaches developed in the thesis.

Forecasting		
	Direct	Indirect
Classification	NC	NC
Regression	NC	D^{FP}

1.5 Contents of the Document

The document is divided into different parts, each further divided into chapters. Figure 1.4 shows how the document is organised. The colours assigned to each chapter are used consistently throughout the document and also appear in the index tabs, ensuring visual continuity and facilitating navigation through the document. Below, each parts and chapters are described.

Part I – Machining and Cutting Tool Replacement
This part aims to highlight the relevance of TCM methods and provide a state-of-the-art overview. It is organized as follows:

- **Chapter 3 – Economic and Environmental Motivation of Tool Condition Monitoring**
  This chapter describes a simulation comparing three different TCM scenarios, allowing for a comparison of production performance from both economic and environmental perspectives. This chapter further motivates the thesis topic and highlights the advantages of TCM compared to a strategy based solely on time-based tool replacement.

- **Chapter 4 – State of the Art of the Monitoring or Forecasting of the Cutting Tool Degradation**
  This chapter presents the state of the art in tool condition monitoring using artificial intelligence methods. It highlights trends in the literature and, more

importantly, identifies research gaps that justify the different approaches explored in the remainder of the document.

Part II – Development of Tool Condition Monitoring Methods on Turning Operation & Part III – Application of Tool Condition Monitoring Methods in Milling Operation

Part I highlights the key literature concepts developed throughout this thesis. These concepts are addressed in the context of both turning and milling. The methods are first developed using a turning dataset in Part II, and then applied to milling in Part III. Both parts follow a similar structure:

- **Chapter 5** for turning and **Chapter 12** for milling.
  These chapters introduce the motivations and context for applying TCM methods to turning and milling.
- **Chapter 6** for turning and **Chapters 13 and 14** for milling.
  These chapters describe the datasets used. The turning dataset originates from a previous thesis [17], while the milling dataset was developed specifically for this work. The latter is presented in more detail across two chapters: the first (13) defines the experimental setup, and the second (14) analyzes the collected data.
- **Chapter 7** for turning and **Chapter 15** for milling.
  These chapters develop an indirect monitoring regression approach to track tool condition using sensor data. A key contribution is the estimation of confidence intervals and the integration of explainability techniques.
- **Chapter 8** for turning and **Chapter 16** for milling.
  These chapters propose an indirect forecasting regression approach, aiming to predict future tool condition using recurrent neural networks.
- **Chapter 9** for turning and **Chapter 10** for milling.
  These chapters present a direct monitoring classification approach using tool images to classify wear into three categories: new, moderately worn, and worn. Image analysis and transfer learning techniques are used to train the models.
- **Chapter 10** for turning and **Chapter 18** for milling.
  These chapters introduce a direct monitoring regression approach using deep encoder-decoder networks to detect wear zones and directly estimate tool condition from images.
- **Chapter 11** for turning and **Chapter 19** for milling.
  These chapters conclude each respective part, summarizing the findings and contributions related to turning and milling.

Part IV – Conclusions and Perspectives
After presenting the results for both turning and milling, Part IV concludes the thesis by summarizing the key outcomes and offering perspectives for future developments.

This page was intentionally left blank

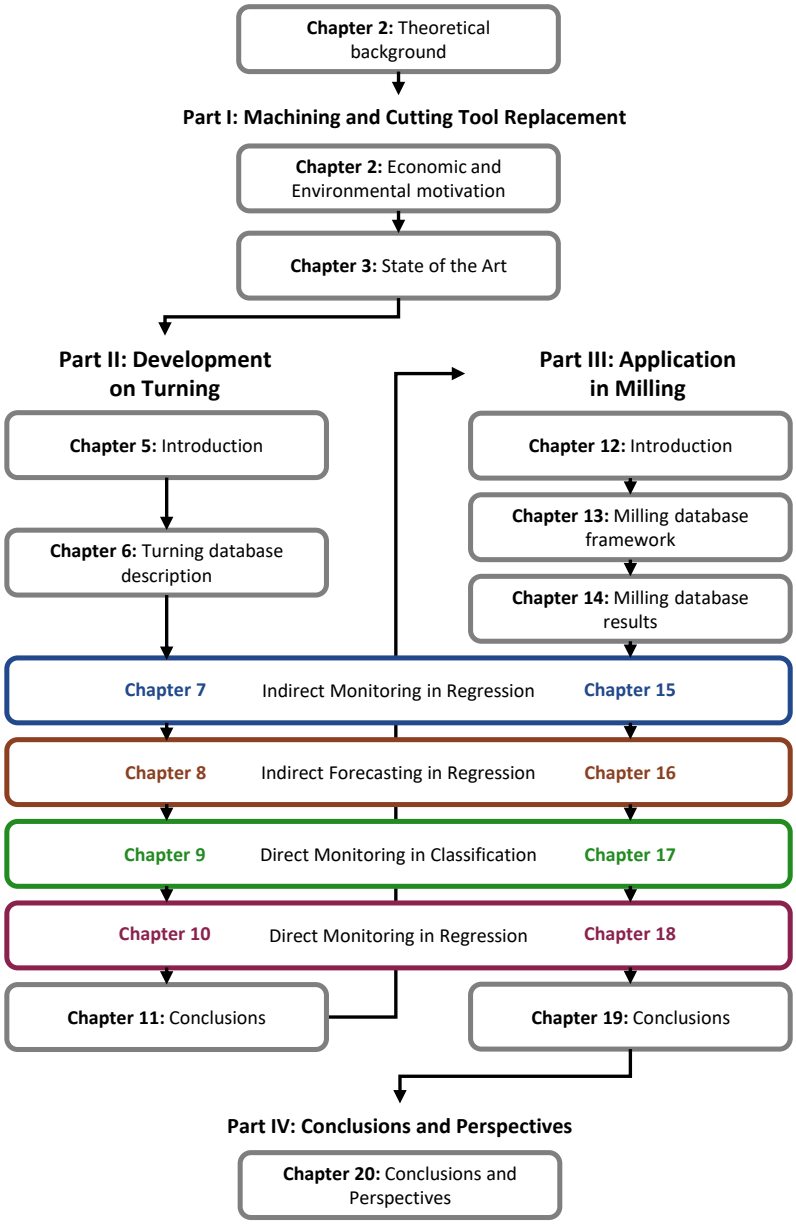


Figure 1.4: Organisation of the document. The colours of the boxes correspond to the colours of the index tabs placed on the pages throughout this document, with similar colours indicating related topics.

Chapter 2

Theoretical Background and Developed Framework

2.1 Introduction

This chapter focuses on presenting the theoretical foundations that will be used throughout the document. Table 2.1 presents the chapter structure. Each section introduces a concept that is either used throughout the entire document or more specifically within certain chapters.

Table 2.1: Overview of the chapter structure, with each section linked to the chapter(s) where its concept is applied.

Section	Described concept	Associated Chapter(s)
2.2	Tool wear in machining	All
2.3	Artificial Intelligence and Machine Learning Definitions	All
2.4	Artificial Neural Network	All
2.5	Convolutional Neural Network	9, 10, 17 and 18
2.6	Vision Transformers	10 and 18
2.7	Explainable Boosting Machines	7 and 15.
2.8	Software and Hardware	All
2.9	Training and Overfitting Prevention	All

2.2 Tool Wear in Machining

This concept is used throughout the whole document.

Any machining operation induces tool wear. The causes of degradation of the tool can be numerous and of very varied origins. Among them, the main causes of degradation are adhesion, abrasion, tribochemical reactions and surface disruption [24]. The word "wear" groups together the surface interactions leading to all of these degradation mechanisms. For a given cutting tool and machine, the dominance of one mechanism over another depends on the cutting conditions and the materials involved [7]. In nominal cutting conditions, the main mechanism of wear is abrasive wear resulting in the degradation of the flank face. This type of wear is favoured by the tool manufacturers, as it is considered as the steadiest and most predictable [25, 26]. Prominence of other kinds of wear generally comes from poor cutting parameters choice [17]. The workpiece material and the choice of cutting parameters each have an effect on the tool life.

Defining an end-of-life criterion for a tool can be complex. Wear is generally accompanied by a degradation in the quality of the machined surface and compliance with required tolerances. The end-of-life criterion is thus variable, depending on the objective of the machining process.

Nonetheless, the ISO 3685 standard defines the value of VB (expressed in mm or µm) as the end-of-life criterion for flank wear in the context of single-point turning tool life testing [27]. The value of VB is obtained by directly measuring the wear land width on the flank face (Figure 2.1). Wear measurement is performed in section B of the flank, which is why VB is sometimes referred to as VB_B in the literature. However, this notation will not be used in this document. For simplicity, VB_{mean}, representing the average wear within the evaluation area, is referred to as VB, while VB_{max} denotes the maximum wear observed in that same area. According to ISO 3685, a tool is considered worn if the value of VB_{mean} reaches 300 µm or VB_{max} is above 600 µm. In industrial applications, this criterion may vary depending on the machining purpose. In particular, the standard allows this criterion to be adapted based on the type of operation being performed, such as roughing or finishing.

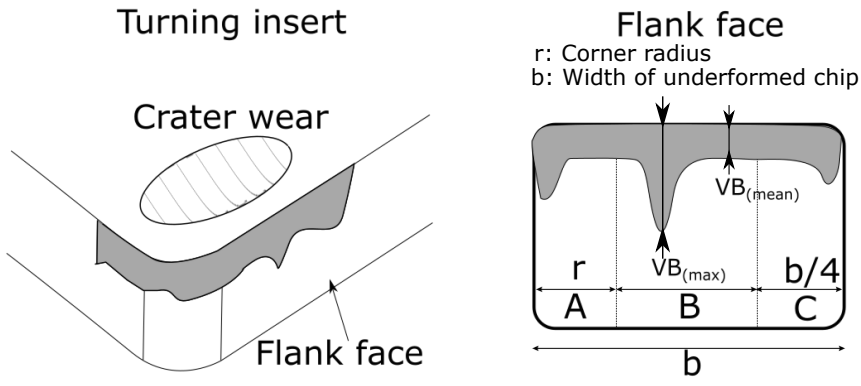


Figure 2.1: Tool wear standardized measurement for a turning insert. Inspired from [27].

In milling, the ISO 8688 standard also defines the VB indicator to measure the flank wear on the teeth of the tool [28]. Figure 2.2 shows how the wear is measured. Similarly to turning, the indicator VB_{mean} and VB_{max} are defined. But, unlike turning, wear is assessed over the entire worn length. Since a milling tool has multiple teeth, VB_{mean} represents the average wear across all teeth. The tool is considered worn out when this average exceeds 300 µm. In contrast, VB_{max} is used to detect excessive wear on individual teeth: if any single tooth surpasses 600 µm, the tool is considered worn.

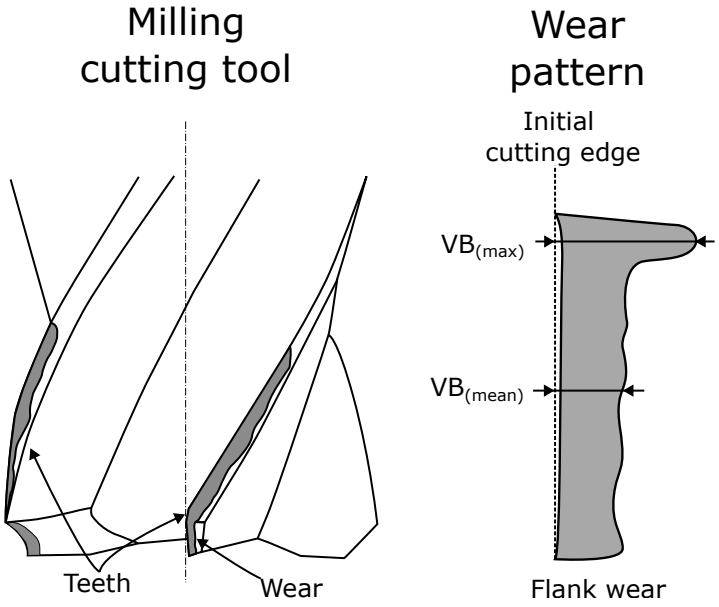


Figure 2.2: Milling tool wear measurement. Inspired from [28]

For both, turning and milling, the evolution of flank wear depends on the cutting condition and is constituted of three phrases (Figure 2.3):

1. **An initial wear zone**. In this phase, the new tool starts to wear quickly. This period is generally short in regard of the cutting life;

2. **A steady-state region**. In this region, the wear slowly gradually increases. The tool spends most of its life in this phase;

3. **Accelerated wear**. This is the end-of-life of the cutting tool, the wear rate starts to increase significantly until the tool is worn. When the tool end-of-life criterion is reached, the tool is replaced by a new one.

Note that, Figure 2.3 shows a specific case where the end-of-life of 300 µm is reached in the third phase of the life of the tool. Other degradation trajectories can be observed depending on the cutting condition.

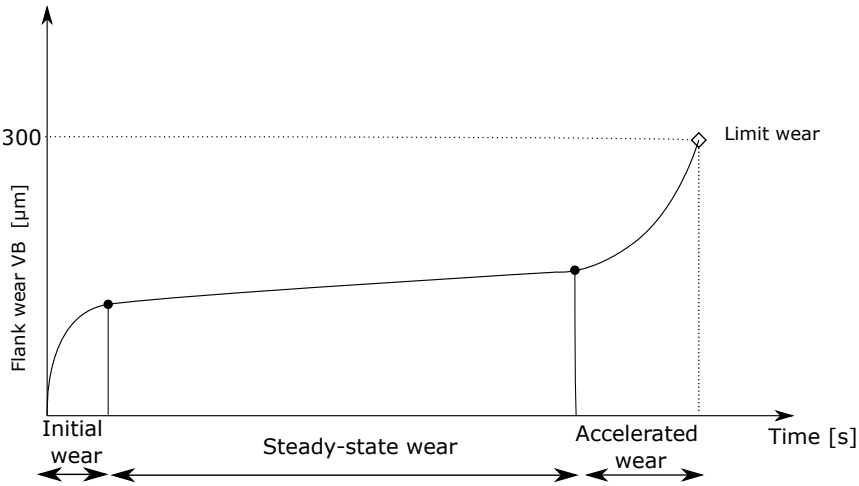


Figure 2.3: Tool life for a given cutting speed.

2.3 Definition of Artificial Intelligence and Machine Learning

This concept is used throughout the whole document.

The Artificial Intelligence (AI) paradigm finds its origin during the 1950s [29] and can be defined as: *"an area of study concerned with making computers copy intelligent human behaviour"* [30]. The word "intelligence" should be taken in the broad sense. Indeed, while the task of recognizing a cat from a dog is not considered as a marker of high intelligence in humans, achieving this task is considered as AI for machines. A definition for intelligence in this framework can be proposed as: *"the computational part of the ability to achieve goals in the world"* [31].

Machine Learning (ML) algorithms can achieve different tasks without the intervention of human beings in the process; they therefore belong to a part of the AI paradigm. They are soft computing techniques in the way that they can adapt their architecture in order to achieve the desired task without having explicitly been programmed to do so. The process of adapting the architecture is called "learning". A model learns through experience, in such a way that the model is influenced by the input data and the desired output. The interest of these techniques is that they are not only able to learn from the data but also to generalize the results for previously unseen data [32]. ML can therefore be defined as: *"The use and development of computer systems that are able to learn and adapt without following explicit instructions, by using algorithms and statistical models to analyse and draw inferences from patterns in data"* [30].

There exist plenty of different approaches that satisfies these definitions. In the following, some approaches will be described but these methods are mainly used for 4 purposes:

1. Classification. The method classify the system in a known class. For example

a data driven image recognition train to classify if the picture is a dog or a cat. The output of the network is the state of the input image (cat or dog)

2. Regression/Monitoring. The approach predicts a value that is not limited to classes. For example, in this thesis case, the method try to evaluate the width of the wear on the cutting tool.

3. Forecasting. The method uses previous knowledge to forecast a tendency in the future. For example : data driven application in the stock market often used forecasting techniques to predict the evolution of the market.

4. Image processing. Images can also be processed by artificial intelligence techniques to generate classification or segmentation on images. Segmentation refers to the process of dividing an image into meaningful regions or parts, such as separating objects from the background or identifying different components within the image. This approach mainly use Convolutional Neural Network to achieve this task.

In this document, the term artificial intelligence will be used as it encompasses related concepts and is the most commonly used term in the literature [33].

2.4 Artificial Neural Network Learning Principle

This concept is used throughout the whole document.

The Artificial Neural Network (ANN) is one of the most popular AI techniques. Inspired by the human brain, the architecture of an ANN is composed of at least three layers: an input layer, some hidden layers and one output layer (Figure 2.4) [34]. Each layer is composed of neurons which are interconnected with the neurons in the following layer. Each neuron has an associated activation function and the weights of the interconnections are updated in the training phase to obtain the desired relationship between the input and the output. The advantages of ANN are their ability to reproduce highly nonlinear input-output characteristics, their ease of software implementation and their robustness to noisy data. Thanks to these strengths, artificial neural networks are sometimes considered capable of performing as effectively as any other method for a given task, assuming adequate computational resources [35]. The main downside of ANN is their need for a large amount of data and processing power to learn from the dataset, the trial and error process to choose their parameters and the fact that they are to be considered as black boxes [36]. ANN can be used to achieve regression or classification.

Figure 2.5 shows a neuron connected to all the N neurons from the previous layer with their respective weight w_i. A neuron performs 2 operations: the sum of the inputs and the computation of the output via an activation function. The first operation sums all the entries from the previous layers and add a bias (b) (Eq.(2.1)). The second operation is to apply the result of this sum (Z) and pass it into an activation function $F(Z)$ (Eq. (2.2)) [37]. This activation function is a hyperparameter that can differ from neuron to neuron. Figure 2.6 and Table 2.2 present some of the most used activation functions, namely: Sigmoid, Tanh and

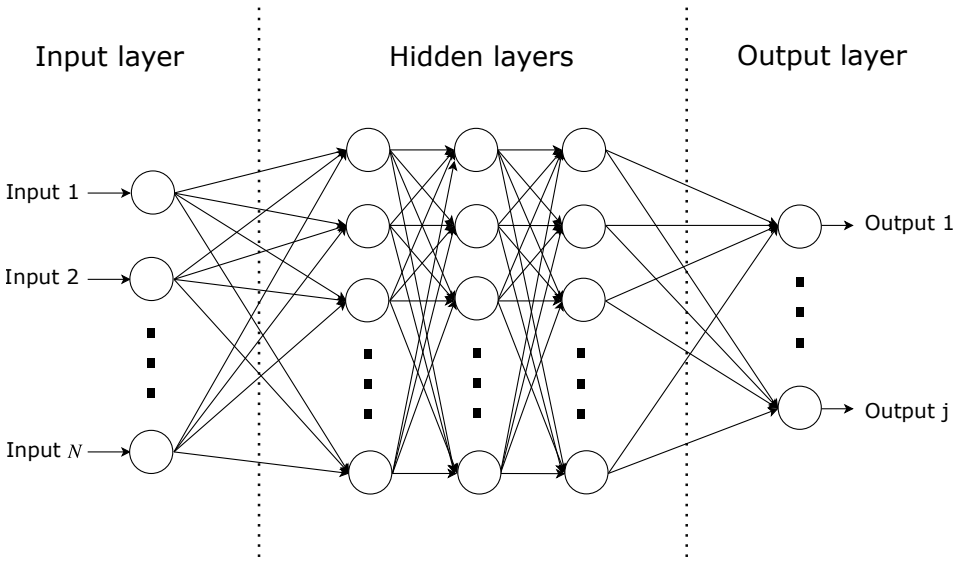


Figure 2.4: General Architecture of an ANN

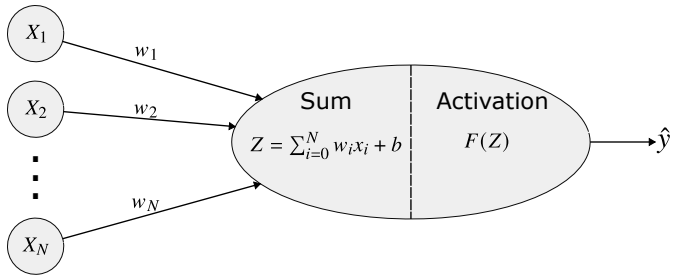


Figure 2.5: A neuron connected to the previous layer

ReLU [38].

$$Z = \sum_{i=0}^{N} w_i x_i + b \tag{2.1}$$

$$\widehat{y} = F(Z) = F(\sum_{i=0}^{N} w_i x_i + b) \tag{2.2}$$

These 2 simple operations (i.e. sum and activation function) define the principle of a neuron. To enable learning from data, the weights and biases of each neuron must be adjusted to establish the correct relationship between input and output. This is achieved through a two-part algorithm: backpropagation and optimisation. Backpropagation computes the gradient of a loss function (C) as a function of weights ($\frac{\partial C}{\partial w_i}$) and bias ($\frac{\partial C}{\partial b}$), while optimisation updates the weights according to these gradients. The goal of this approach is to find the minimum in the solution

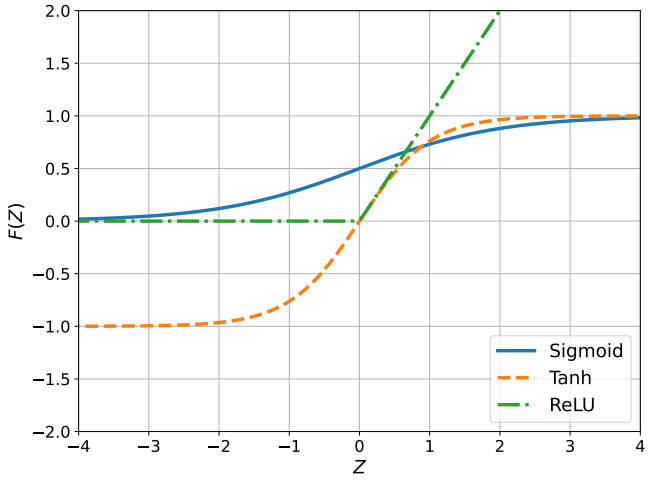


Figure 2.6: Sigmoid, Tanh and ReLU activation functions

Table 2.2: Activation functions and associated equations

Activation function	Equation
Sigmoid	$F(Z) = \frac{1}{1+e^{-Z}}$
Tanh	$F(Z) = \frac{e^{Z}-e^{-Z}}{e^{Z}+e^{-Z}}$
ReLU	$F(Z) = \max(0, Z)$

space of the loss function C. The gradient is calculated to determine in which direction the weights should be adjusted in order to minimize the loss, and thus reduce the error between the input and the output (Figure 2.7).

In the following example of this learning algorithm, a mean squared error (MSE) loss function between the correct output (y) and the actual neuron output ($\widehat{y_i}$) (Eq.(2.3)) and a sigmoid activation function (Eq.(2.4)) are used as they are the most common in practice. The gradient of the loss function (C) with respect to the weight (w_i) can be computed as in Eq.(2.5). In this equation, the chain rule is applied as the loss function C and the weight w_i are not directly related. A similar approach is computed for the gradient of the loss with respect to the bias (Eq. (2.6)) [39].

$$MSE = \frac{1}{N}\sum_{i=0}^{N}(y - \widehat{y_i})^2 = C \tag{2.3}$$

$$\widehat{y} = F(Z) = \frac{1}{1+e^{-z}} = \phi(Z) \tag{2.4}$$

$$\frac{\partial C}{\partial w_i} = \frac{\partial C}{\partial \widehat{y}}\frac{\partial \widehat{y}}{\partial Z}\frac{\partial Z}{\partial w_i} \tag{2.5}$$

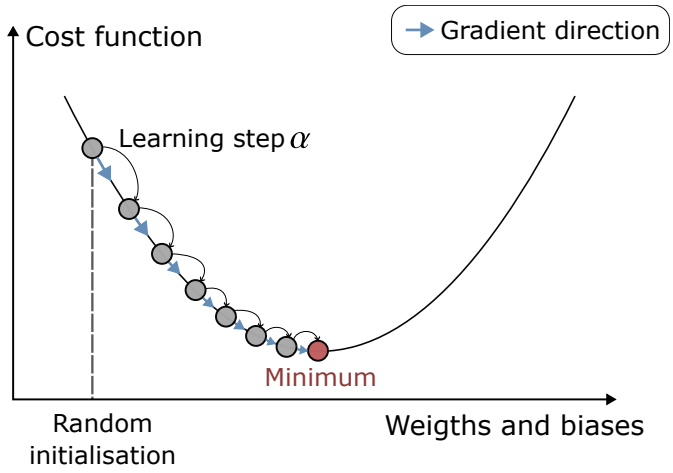


Figure 2.7: Neural network learning principle. The network parameters are initially selected at random. Then, gradient descent computes how to adjust them in order to minimize the error defined by the cost function.

$$\frac{\partial C}{\partial b} = \frac{\partial C}{\partial \widehat{y}} \frac{\partial \widehat{y}}{\partial Z} \frac{\partial Z}{\partial b} \tag{2.6}$$

Each partial derivative is computed as follow (Eq.(2.7) to (2.10))[39]:

$$\frac{\partial C}{\partial \widehat{y}} = \frac{\partial \frac{1}{N} \sum_{i=0}^{N} (y - \widehat{y_i})^2}{\partial \widehat{y}} = 2 \frac{1}{N} \sum_{i=0}^{N} (\widehat{y} - \widehat{y_i}) \tag{2.7}$$

$$\frac{\partial \widehat{y}}{\partial Z} = \frac{\partial F(Z)}{\partial Z} = \frac{\partial \phi(Z)}{\partial Z} = \phi(Z)(1 - \phi(Z)) \tag{2.8}$$

$$\frac{\partial Z}{\partial w_i} = x_i \tag{2.9}$$

$$\frac{\partial Z}{\partial b} = 1 \tag{2.10}$$

Such that the complete gradient equations are (Eq.(2.11) and (2.12)) [39]:

$$\frac{\partial C}{\partial w_i} = \frac{2}{n} \phi(Z)(1 - \phi(Z)) x_i (\sum_{i=0}^{n} (y - \widehat{y_i})) \tag{2.11}$$

$$\frac{\partial C}{\partial b} = \frac{2}{n} \phi(Z)(1 - \phi(Z)) (\sum_{i=0}^{n} (y - \widehat{y_i})) \tag{2.12}$$

The computation of these gradient concludes the first part of the learning algorithm. In order to update the weight, the optimization approach is used. A common optimization approach is the gradient descent for which the learning rate (α) parameter must be defined [40]. The new weight (w_i^{new}) and bias (b^{new}) are updated as:

$$w_i^{new} = w_i - (\alpha \frac{\partial C}{\partial w_i}) \tag{2.13}$$

$$b^{new} = b - (\alpha \frac{\partial C}{\partial b}) \tag{2.14}$$

This algorithm is repeated until the weights and bias converge to establish the correct relationship between the input and output of the network. During training, it is possible to update the weights in a timely manner, this value is called the "batch size" and defines the amount of data used before the model is updated. It is often necessary to repeat this learning process several times over the whole dataset, the number of runs over the whole dataset is called the "epoch".

2.5 Convolutional Neural Network and their Specific Layers

This concept is used in Chapters 9, 10, 17 and 18.

Working Principle

A Convolutional Neural Network (CNN) is an extension of an Artificial Neural Network (ANN), designed to handle more complex data such as images or raw signals (Figure 2.8) [41]. Unlike classical ANNs, CNNs include additional layers: convolutional and pooling layers that allow the model to automatically extract relevant features from the input data. These layers are stacked to progressively capture higher-level patterns in the data and are fully decribed below [42]. The convolutional layers apply trainable filters (kernels) to the input, highlighting specific features such as edges or textures [42]. The pooling layers reduce the spatial dimensions of the data, helping to lower computational cost. Together, these layers form the feature extraction part of the network. After this stage, the extracted features are passed to one or more fully connected layers, similar to those in a traditional ANN. These layers perform the final classification or regression task. Non-linear activation functions are applied throughout the network to enable the learning of complex patterns [42].

CNNs are particularly well-suited for tasks such as image classification, object detection, and segmentation [41]. Because they process high-dimensional data and involve many parameters, CNNs are generally deeper and more computationally intensive than classical ANNs. However, their ability to autonomously learn relevant features makes them a powerful tool in modern deep learning.

Their learning mechanism is still based on gradient descent, as described for ANNs, allowing the network to iteratively adjust its parameters to minimize a loss function.

Convolutional Layers

Convolutional layers are defined by three key parameters: kernel size, stride, and padding. These parameters are illustrated in the following example.

Figure 2.9 illustrates how an image (represented as a matrix of numbers) is transformed through a convolution operation. In a convolutional layer, the kernel size is first defined, which determines the dimensions of the filter (shown in yellow

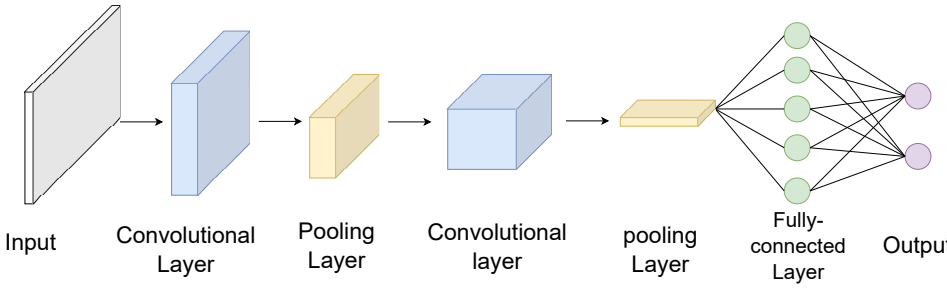


Figure 2.8: CNN principles. Convolutional and pooling layer are added to fully connected layers. These additional layers aim to extract relevance features from raw data or images.

in the figure). In this example, the kernel is 3×3. This kernel is applied to a window (shown in dark blue) that slides across the image. At each position, the convolution operation is performed by computing the element-wise product between the kernel and the window, followed by summing the results to produce a single value in the output image [38].

To preserve the spatial dimensions of the input image, zero-padding is often added around the borders. In this example, a padding of size 1 is used, meaning one border of zeros is added around the image. Padding ensures that the output image retains the same size as the input, which is especially useful when stacking multiple convolutional layers.

Another important parameter is the stride, which defines how many pixels the window moves at each step. A stride of 1 means the window moves one pixel at a time, while a larger stride results in a smaller output and more aggressive downsampling.

In the figure, the image is represented as a one channel matrix. Nonetheless, CNNs also support multiple channels, such as the red, green, and blue channels in a colour image. In this case, the kernel extends across all channels, and the convolution is computed across the full depth of the input [38].

Each convolutional layer typically uses multiple kernels, each producing a different feature map. These feature maps capture different aspects of the input, such as edges, textures, or patterns. The kernel values are trainable parameters, meaning they are learned during training to detect features relevant to the task.

Pooling Layers

A pooling layer (or subsampling layer) is often used after a convolutional layer in a Convolutional Neural Network (CNN). Its main purpose is to reduce the spatial dimensions of the intermediate representations (convolved feature maps) while preserving the most important information. This helps to reduce the number of parameters, limit overfitting, and decrease computational cost [38].

Figure 2.10 shows the working principle of the pooling layer. In this figure the 4×4 convolved feature map is divided on 2×2 blocks due to a kernel size and stride of 2×2. The most common type of pooling is max pooling, which divides the input into small regions (e.g., 2×2 blocks) and retains only the maximum value from each region. Another variant is average pooling, which computes the average of

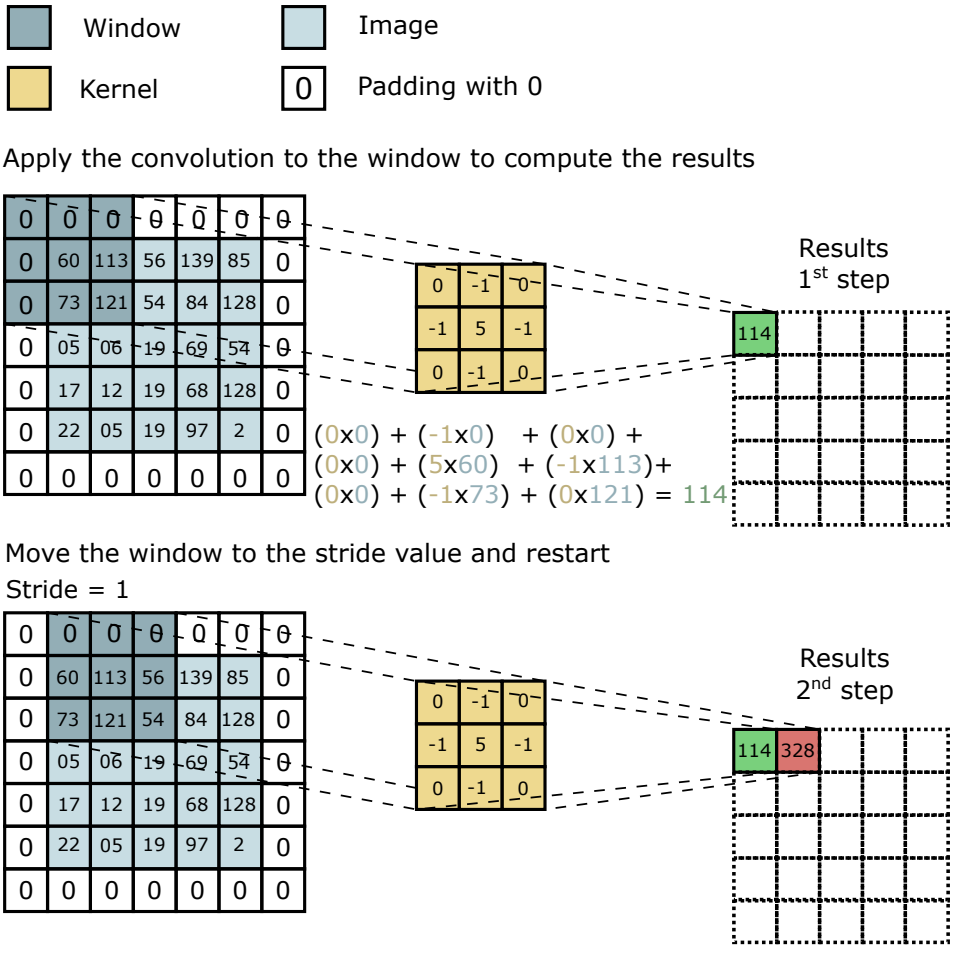


Figure 2.9: Convolution applied to an image matrix. The convolution window slides over the image, applying the kernel to compute the resulting values. The window slide is controlled by the stride value. Here the stride is 1. In this example, zero-padding of size 1 is added around the image.

the values in each region. These operations help preserve dominant features while making the model more robust to small translations or distortions in the input.

Pooling layers do not involve any learnable parameters; they are fixed operations that act as a form of simplification. By combining convolution and pooling, a CNN can extract increasingly abstract and hierarchical representations as the data progresses through the network layers.

2.6 Vision Transformers

This concept is used in Chapters 10 and 18.

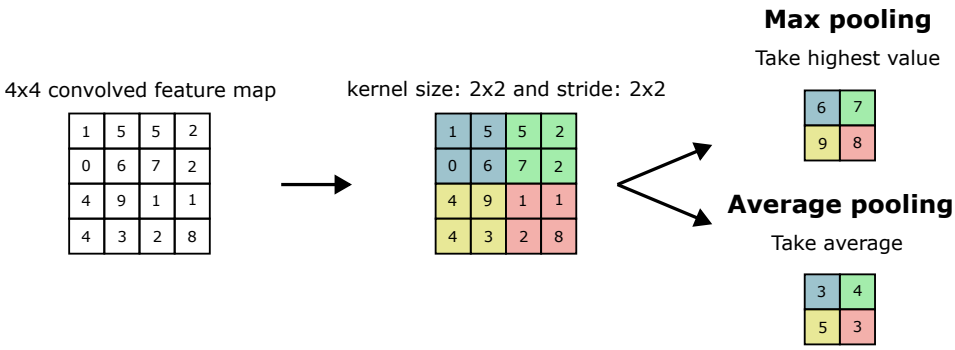


Figure 2.10: Max and average pooling example. A 4×4 convolved feature map from the convolutional layer is divided into 2×2 blocks, based on the kernel size and stride of 2×2. Each colour represents a distinct window. The resulting outputs from both max pooling and average pooling operations are shown.

Deep learning applications based on Convolutional Neural Networks (CNNs) have dominated most field of AI application. More recently, transformer-based approaches have emerged, achieving comparable or even superior performance on large datasets [43, 44].

Transformers originated in the field of Natural Language Processing (NLP), where they are designed to understand and interpret human language [43]. The primary innovation of these models was the replacement of traditional recurrent layers with attention mechanisms. This shift was motivated by the fact that attention layers are computationally less complex, easier to parallelise, and particularly effective at capturing long-range dependencies in data [43]. These models have since been adapted to other domains, including tasks traditionally handled by CNNs.

In the following example, the focus is on Vision Transformers (ViTs) [44]. However, it is important to note that transformers are not limited to images. They can also be applied to other types of data, such as time series. Their core operating principle remains the same; only the nature of the input changes.

Figure 2.11 shows an example of a Vision Transformer that performs classification. The input image is divided into patches. These patches are then flattened and embedded before being passed into the transformer encoder. The encoder processes the sequence of patch embeddings using self-attention mechanisms to capture relationships between different parts of the image. At the output of the model, fully connected layers are used. In this example, the task is image classification, so the Multi Layer Perceptron (MLP) head is connected to an output layer that predicts the class to which the image belongs. A special component of the Vision Transformer is the classification token (often denoted as [CLS]). This token is prepended to the sequence of patch embeddings and does not correspond to any specific image region. Instead, it serves as a global representation of the image. Through the self-attention mechanism, the [CLS] token interacts with all other tokens and learns to aggregate the most relevant information from the entire image. At the end of the transformer encoder, the final state of this token is used as the input to the classification head.

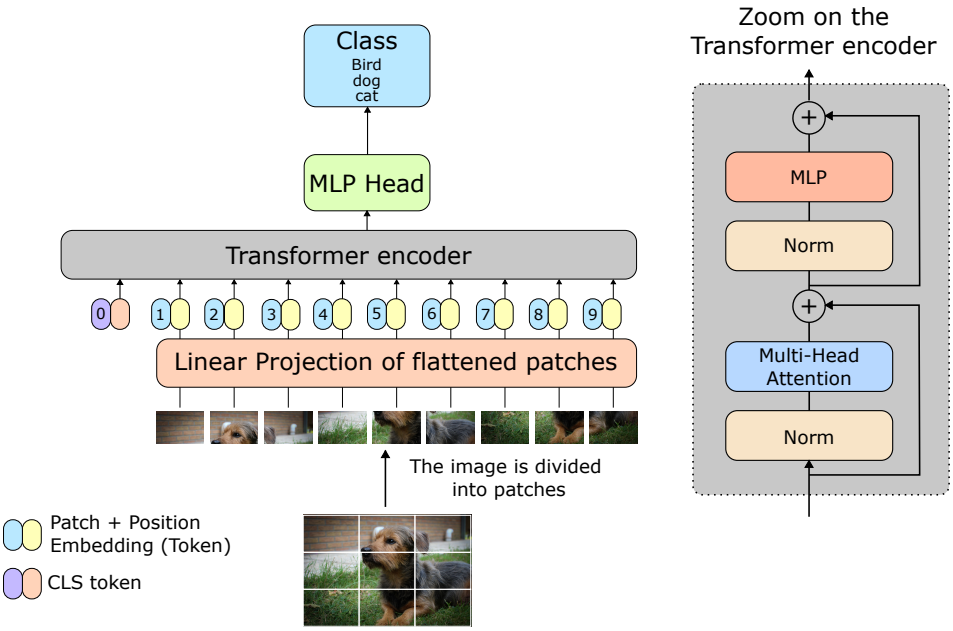


Figure 2.11: Working principle of a vision transformer for classification. Inspired from [44]

Vision Transformers contain a large number of trainable parameters, distributed across several components:

- Linear projection layers for embedding image patches.
- Self-attention layers, including the query, key, and value matrices.
- Feed-forward networks within each transformer block.
- Positional embeddings, which may be learned.
- The classification token vector itself, which is also learned during training.
- The MLP head and final fully connected output layers.

This high number of parameters gives Vision Transformers strong modelling capabilities, but also makes them more data-intensive than CNNs. However, their ability to capture long-range dependencies between input data allows them to recognise complex and distant patterns that might escape traditional convolutional architectures. This makes them particularly well suited to tasks where contextual understanding of the entire image is crucial.

2.7 Explainable Boosting Machines Working Principles

This concept is used in Chapters 7 and 15.

Models such as neural networks are often referred to as "black-box" models because their internal workings are not easily interpretable. In contrast, "glass-box" models, as the name suggests, are transparent and allow for greater explainability. These models are designed to provide explanations for their predictions, making them more suitable in contexts where understanding the decision process is critical.

However, a common challenge with glass-box models is that they tend to be less accurate than black-box models. There is often a trade-off between model explainability and performance, and achieving both simultaneously remains a challenge. Recently, new approaches have emerged aiming to bridge this gap by offering models that are both interpretable and highly effective [45].

One such method is the Explainable Boosting Machine (EBM). EBM is an interpretable machine learning technique based on the principle of boosting applied to generalized additive models (GAMs) [45]. Unlike traditional gradient boosting models, EBM builds its predictions by combining the individual effects of each feature, while maintaining a transparent and easily visualizable structure.

The prediction made by an EBM can be expressed using the following formula (Eq. (2.15)):

$$\hat{y} = \beta_0 + \sum_{i=1}^{N} f_i(x_i) \tag{2.15}$$

Where:

- $\hat{y}$ is the model prediction,
- β_0 is the baseline value (or "intercept"), corresponding to the mean of the targets (in regression) or the average log-odds (in classification),
- $f_i(x_i)$ is the contribution function of the variable x_i,
- N is the total number of variables.

Each function f_i is learned independently using small decision trees, typically constrained in the number of leaves (This hyperparameter is named: max leaves in Chapters 7 and 15), in order to capture the specific effects of each variable. These functions are then smoothed to produce continuous and interpretable curves, especially for numerical variables (This hyperparameter is named: Smoothing round in Chapters 7 and 15). The model can also incorporate interactions between variables, represented as functions $f_{ij}(x_i, x_j)$. This interaction can be controlled to limit the model to a pure additive model (This hyperparameter is named: interaction in Chapters 7 and 15).

The learning process relies on bagging and boosting techniques to enhance both the robustness and explainability of the model:

- Boosting is used to iteratively refine the model by focusing on the errors made in previous iterations. In EBM, this is applied in a controlled way to maintain explainability while improving accuracy. Specifically, EBM uses multiple sub-samples (This hyperparameter is named: inner bags in Chapters 7 and 15) for each feature to train small decision trees that capture

individual variable effects. These trees are then aggregated to form smooth, interpretable functions for each feature.

- Bagging (Bootstrap Aggregating) helps reduce variance and prevent overfitting by training multiple models on different random samples of the data and averaging their predictions. Specifically, EBM employs outer bags (This hyperparameter is named: outerbag in Chapters 7 and 15), which are independent models trained on different subsets of the data. These outer bags serve to estimate the uncertainty of the predictions, providing confidence intervals and helping to assess the stability of the learned patterns.

This combination of techniques allows EBM to produce robust predictions while maintaining a transparent structure, accompanied by explanatory curves that clearly show how each variable contributes to the final output.

2.8 Software and Hardware

This concept is used throughout the whole document.

Software Tools

To develop and train AI models, it is essential to use appropriate software tools that support model creation and training. In this thesis, all models, figures, and related outputs were created using the Python programming language. Python is particularly well-suited for data analysis and artificial intelligence due to its powerful capabilities and its rich ecosystem of libraries for data manipulation, visualization, and machine learning.

Table 2.3 lists the software libraries used throughout this work. Among them, PyTorch [46] and TensorFlow [47] are the primary frameworks employed for building neural networks. Initially, TensorFlow was used, but it was later replaced by PyTorch, which proved to be better aligned with the software tools developed for data manipulation in this thesis. Other essential libraries include Pandas [48] and NumPy [49], which are widely used for efficient data manipulation, especially when working with large numerical datasets. SciPy [50] and Matplotlib [51] were also utilized, SciPy for advanced mathematical computations and Matplotlib for generating visual representations of the results.

Developed Software

To obtain the results presented throughout this thesis, dedicated software code was developed. Figure 2.12 illustrates the sequence of operations that were necessary to implement in order to produce the different results. This sequence represents a data processing pipeline designed to prepare and structure the data for training the various models discussed in this document.

The first step is data collection, which involves gathering raw data from diverse sources, regardless of format. This is followed by data cleaning, a crucial phase to ensure the data is both accessible and reliable. Particular attention was paid to

Table 2.3: Main softwares used to obtain the results in the thesis

Software	Task
Python	Programming language
PyTorch [46]	Open source platform for machine learning
Tensorflow [47]	Open source platform for machine learning developed by Google
Pandas [48]	Data manipulation
Scipy [50]	Mathematical computation
Numpy [49]	Array manipulation
Matplotlib [51]	Figure generation

these initial steps, especially to extract the most relevant information from sensors and to handle issues such as missing values. These steps were especially critical given the volume of data involved (several hundreds of gigabytes) which required the development of robust algorithms for efficient processing.

Once the data has been collected and cleaned, it is essential to understand its structure and the trends it may reveal. This is typically achieved through data exploration, which includes visualization and the computation of indicators to assess how the data evolves and how it can be leveraged in subsequent stages. The next phase is feature engineering, which consists of selecting and/or creating features that best capture the patterns observed in the dataset. This step also defines which data is accessible to the models and which is excluded.

After identifying the relevant features, the data is organized into a structured dataset, often segmented into subgroups. The model is then trained according to its specific objectives and subsequently evaluated. The evaluation phase is particularly important to ensure that the results are not biased due to poor data selection in earlier stages.

Finally, this entire process is inherently iterative. Model improvements are often based on the evaluation results and may involve adjustments to the model itself or refinements in the feature engineering process. Within the scope of this thesis, it was not necessary to develop the subsequent stages required for industrial deployment, such as model deployment, monitoring, and maintenance.

It is important to emphasize that all the code developed was designed to be as generalizable as possible. In other words, every component, whether for data processing, model building, or training, was created to be modular and adaptable, allowing methods to be applied without having to redevelop everything for each specific case. This modularity is precisely what will enable, in this thesis, the application of methods developed for turning to milling, with all code being reused and only minimal adjustments required for each specific context.

Hardware

In addition to the software tools, the hardware used in this thesis played a significant role in supporting the development and training of AI models (Table 2.4). Two main systems were utilized. For smaller models and data preprocessing tasks,

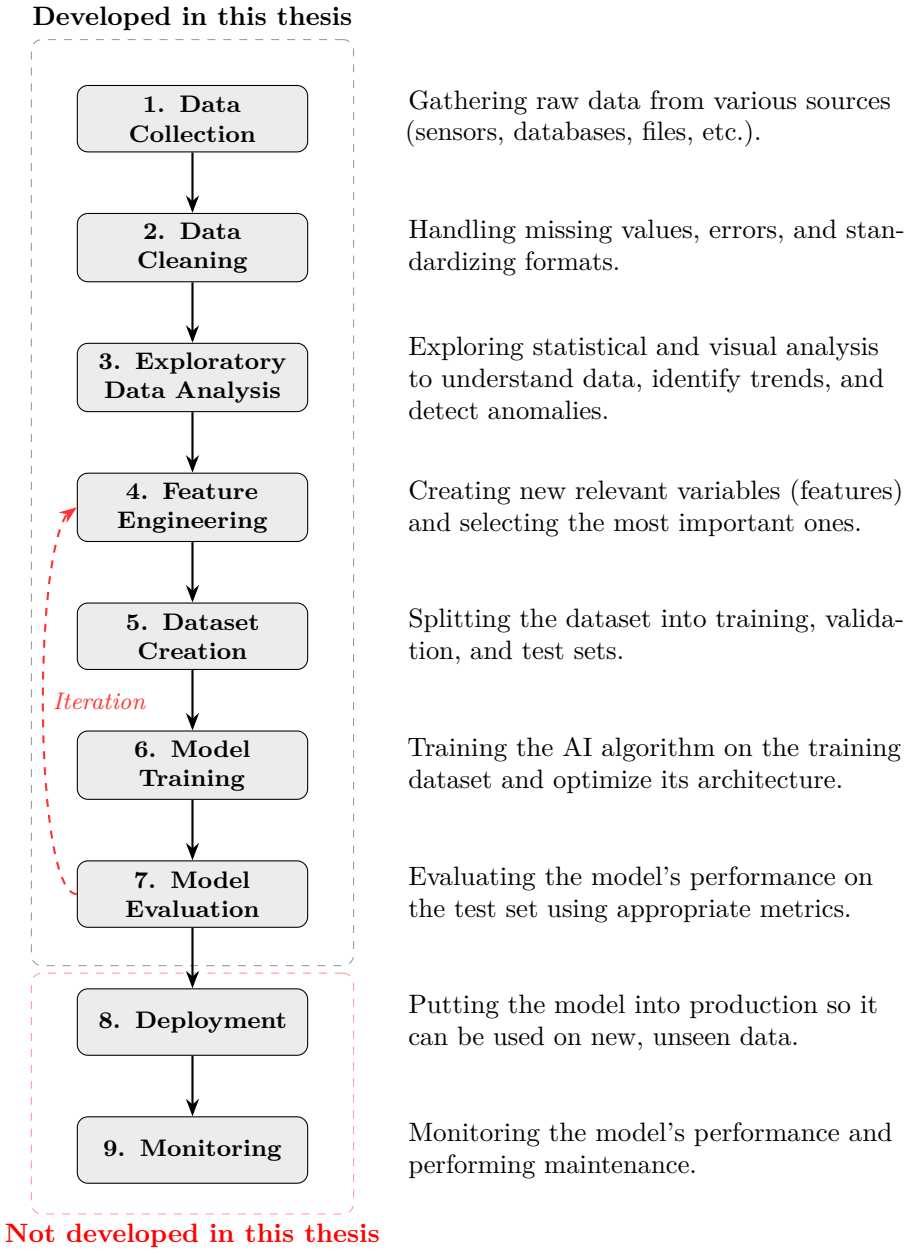


Figure 2.12: Pipeline of steps developed in the thesis and the remaining steps to deploy the models in production.

a laptop equipped with an Intel Core i7-9750H CPU and an NVIDIA GeForce GTX 1650 GPU was used. The i7-9750H delivers approximately 0.337 TFLOPS of single-precision performance, while the GTX 1650 offers around 3.04 TFLOPS, making this setup suitable for lightweight computations and initial experimentation. TFLOPS stands for Tera Floating Point Operations Per Second. It is a measure of a computer's or GPU's performance, specifically how many trillions of floating-point calculations it can perform in one second. In the context of AI and machine learning, higher TFLOPS generally means the hardware can process more data faster, which is especially useful for training large neural networks. For more demanding tasks, a desktop PC featuring an AMD Ryzen 9 7950X3D CPU and an NVIDIA GeForce RTX 4080 GPU was employed. The Ryzen 9 7950X3D provides about 1.17 TFLOPS, and the RTX 4080 delivers a substantial 48.7 TFLOPS, enabling efficient training of large-scale neural networks and handling complex data processing workflows.

Table 2.4: Hardware used to obtain the results in the thesis with their computing capabilities

Hardware	**TFLOPS**
Lightweight tasks	
Intel Core i7-9750H	0.337 TFLOPS
NVIDIA GeForce GTX 1650	3.04 TFLOPS
Intensive tasks	
AMD Ryzen 9 7950X3D	1.17 TFLOPS
NVIDIA GeForce RTX 4080	48.7 TFLOPS

2.9 Training Networks Based Model and Overfitting Prevention

This concept is used throughout the whole document.

As mentioned in Section 2.4, the goal of training any neural network is to find a combination of parameters that minimizes the error between the model's input and output. These parameters include the weights and biases in a standard neural network, the kernel values in convolutional layers, or any trainable parameters. To train a neural network, one must first define a loss function and an optimizer. The loss functions, discussed in Section 2.4, quantifies the error between the predictions and the true labels. The optimizer applies gradient descent to determine how the weights should be adjusted to reduce this error. To effectively train the network, the training process is repeated over multiple iterations. Each complete pass through the training dataset is referred to as an "epoch", while the "batch size" determines how the dataset is split into smaller subsets that are processed sequentially. During each batch iteration, the model performs a forward pass to generate predictions, followed by a loss computation to evaluate the difference between the predicted and

true outputs. Then, a backward pass is executed to calculate gradients, which are used by the optimizer to update the model parameters. In this thesis the "Adam" optimizer will always be used as it has a good compromise between speed and stability [52]. This cycle is repeated across all batches and epochs to progressively improve the model's performance. The pseudocode of this simple training loop follows the logic outlined at Algorithm 1.

Algorithm 1 Simple Neural Network Training Loop

```
Define training data
Initialize model parameters (architecture, weights, biases,...)
Define loss function and optimizer
for each epoch do
    for each batch in training data do
        Forward pass: compute model predictions
        Compute loss between predictions and true labels
        Backward pass: compute gradients of loss w.r.t. parameters
        Update parameters using optimizer
    end for
end for
```

This pseudocode outlines the core steps of training: forward pass, loss computation, backward pass, and parameter updates. While this represents the basic training loop, in practice, the process is adapted to reduce the risk of overfitting.

Overfitting occurs when a model performs well on the training data but fails to generalize to unseen data. This happens when the network memorizes specific patterns or noise in the training set rather than learning the underlying relationships between inputs and outputs [38]. An overfitted model is therefore overly constrained and may produce misleading results when applied in real-world scenarios. To detect and prevent overfitting, the dataset is typically divided into three subsets: a training set, a validation set, and a test set [53]. The training and validation sets are used during the training process, while the test set is reserved for final evaluation after training is complete. This division ensures that the model is assessed on data it has not seen during training, providing a more reliable measure of its generalization ability. A common practice is to allocate approximately 75 % of the dataset to the training and validation sets, and the remaining 25 % to the test set. However, these proportions may vary depending on the specific characteristics of the dataset.

During training, the validation set is used to monitor the model's performance and detect signs of overfitting. If the model performs significantly better on the training set than on the validation set, it may be overfitting. Figure 2.13 shows how overfitting is detected in the evolution of the training and validation loss. The training loop is therefore extended to include validation at each epoch, as shown in the pseudocode Algorithm 2.

Algorithm 2 Training Loop with Validation

```
Define training data and validation data
Initialize model parameters (architecture, weights, biases,...)
Define loss function and optimizer
for each epoch do
    for each batch in training set do
        Forward pass: compute model predictions
        Compute loss between predictions and true labels
        Backward pass: compute gradients of loss w.r.t. parameters
        Update parameters using optimizer
    end for
    Compute loss on validation set
end for
```

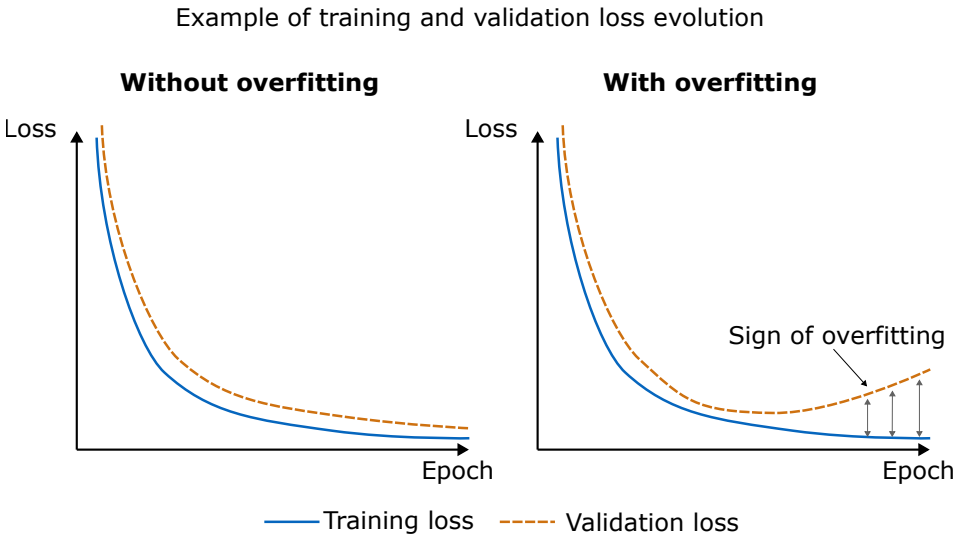


Figure 2.13: Evolution of learning and validation losses in the case of overfitting or not. When the model is overfitted, the validation loss diverges from the learning loss.

In addition to monitoring the validation loss to prevent overfitting, other techniques are commonly used to guide and optimize the training process. One such technique is "EarlyStopping", which helps avoid unnecessary computations by stopping the training process once the model has converged. It works by monitoring the validation loss, and if no improvement is observed over a specified number of epochs, defined by the patience parameter, the training is stopped. This prevents the model from overfitting and conserves computational resources.

Another widely used technique is "ReduceLROnPlateau" (Reduce Learning Rate on Plateau). Like EarlyStopping, it monitors the validation loss during training. When it detects that the loss has stopped improving, it reduces the learning rate (denoted by α in Section 2.4) by a specified factor. This allows the model to fine-tune its learning and avoid diverging from an optimal solution. In this

thesis, the factor is typically set to 0.1. Both techniques use a patience parameter, which is adjusted based on the number of epochs required to train the model. Importantly, the patience for ReduceLROnPlateau should be set lower than that of EarlyStopping to allow the learning rate to adjust before training is stopped.

Together, these methods ensure that the model reaches optimal performance while avoiding overfitting and reducing unnecessary computation. The updated pseudocode that includes both techniques is presented in Algorithm 3.

Algorithm 3 Training Loop with EarlyStopping and ReduceLROnPlateau

```
Define training data and validation data
Initialize model parameters (architecture, weights, biases,...)
Define loss function, optimizer, and learning rate scheduler
Set patience and factor for ReduceLROnPlateau
Set patience for EarlyStopping
Initialize counters for validation loss tracking
for each epoch do
    for each batch in training set do
        Forward pass: compute model predictions
        Compute loss between predictions and true labels
        Backward pass: compute gradients of loss w.r.t. parameters
        Update parameters using optimizer
    end for
    Compute loss on validation set
    if validation loss improved then
        Save model state
        Reset patience counters
    else
        Update patience counters
        if ReduceLROnPlateau patience exceeded then
            Reduce learning rate by factor
        end if
        if EarlyStopping patience exceeded then
            Stop training
        end if
    end if
end for
```

This training loop, along with the monitoring techniques described, is consistently applied throughout this thesis, even when not explicitly mentioned, to ensure that the results are not affected by overfitting and to optimize the learning process across the various models studied.

2.10 Conclusions

This chapter brings together several concepts that are used throughout this document. These concepts will therefore be referenced without further explanation in the following sections.

Some notions are consistently used across the entire document, such as tool wear degradation, the concept of artificial intelligence, learning methods, and the developed Python framework. Other concepts appear more periodically, depending on the chapters in which they are applied as shown in Table 2.1.

Part I

Machining and Cutting Tool Replacement

Chapter 3

Economic and Environmental Motivation of Tool Condition Monitoring

The methodology presented in this chapter has been published in a scientific article [54], which focused on a different material than the one discussed here. Nevertheless, the majority of the chapter's content has been adapted from that publication.

3.1 Introduction

This thesis aims to develop methods for monitoring the degradation of cutting tools in order to enable their replacement at the optimal moment. However, before developing such approaches, it is important to assess their actual industrial benefits compared to currently used tools replacement strategies. Understanding the potential advantages of these monitoring systems is therefore essential prior to their implementation.

Therefore, this chapter aims to quantify the economic and environmental impacts of various tool replacement strategies within an idealized case study. The case study compares three distinct tool replacement strategies, which are described in detail in Section 3.2.2. These scenarios replicate the impact of implementing, or not implementing, tool wear monitoring techniques. The simulated model used to assess the economic and environmental impacts is limited to three key components (Section 3.2.1):

- **The cutting tools**, modelled with a lifespan derived from a Weibull distribution, with degradation trajectories simulated using a hyperbolic sine function.

- **The CNC machine**, represented as a medium-sized unit that consumes a constant amount of power during machining.
- **The workpiece**, produced from raw C45 material.

Each of these components is characterized by cost and CO_2-eq emission values sourced from the literature. Using these three components, it is possible to simulate a simplified production process of workpieces. The simulated process represents a machine that operates using a cutting tool, which is employed to machine a workpiece. During machining, the degradation of the cutting tool is simulated, as it directly affects the quality of the produced workpiece. To evaluate the differences between the various strategies (whether or not TCM is used), tool replacement is simulated differently depending on the scenario. The overall production performance is then assessed from both economic and environmental perspectives. To incorporate variability in the degradation process, the model employs a Monte Carlo simulation. The complete description of the model and its results are structured as follows:

- Section 3.2 presents the methodology used to generate the results, including a description of the simulation model (Section 3.2.1) and the scenarios considered (Section 3.2.2).
- Section 3.3 compares the outcomes of the different scenarios by identifying their optimal operating points (Section 3.3.1) and compare their optimal results (Section 3.3.2).

3.2 Methodology

Figure 3.1 illustrates the framework used to evaluate the economic and environmental impact of different tool replacement strategies. The framework is divided into two parts: the production model, which simulates workpiece manufacturing, and the set of scenarios considered. In the following sections, Section 3.2.1 presents the parameters and characteristics of the three machine components, while Section 3.2.2 describes the three corresponding scenarios.

3.2.1 Production Model

The model presented in this chapter aims to simulate various cutting tool replacement strategies and to evaluate their economic and environmental impacts, while also considering the surface quality produced as a function of the tool's condition. These impacts are measured in euros (EUR) and kilograms of CO_2-equivalent (kg CO_2-eq).

The case study focuses on the machining of C45 parts, representative of the experimental scenario described in Part II and III. It is assumed that the parts are machined using a CNC machine operating in a two-shift system (16 hours per day), 7 days a week, over the course of one month, with the facility located in Belgium.

Figure 3.2 illustrates the system boundaries, which encompass three main components: the cutting tool, the workpiece, and the electrical consumption of the CNC machine. The system accounts for the sourcing and processing of raw

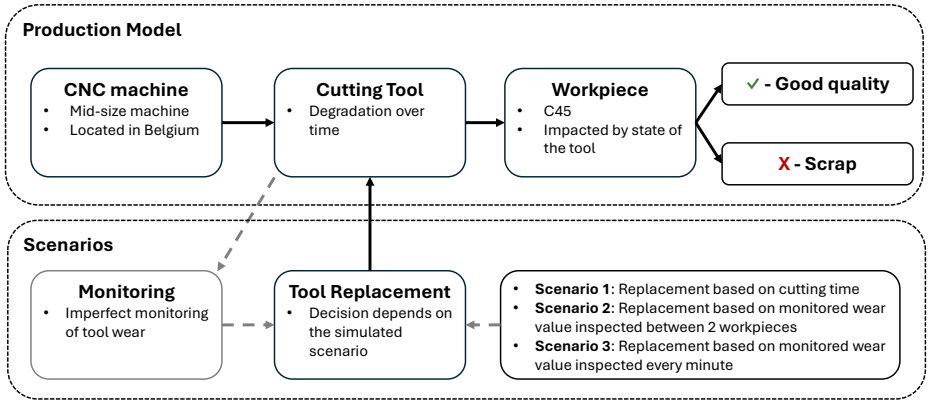


Figure 3.1: Framework of the simulated model developed in this chapter. The model consists of two main parts: the simulation of workpiece production and the implementation of tool replacement strategies.

materials needed to produce coated cutting tools and C45 parts. The electricity used in the process is considered supplied by the Belgian electric grid, whose quantity of CO_2 per consumed kilowatt-hour depends on Belgian energy mix and imported electricity. The outputs of the system include the cutting tool and the machined parts. These parts may either meet quality standards or be classified as scrap if they were produced with a tool that was worn at the time of machining. This study does not consider the use phase of the machined parts, nor the potential end-of-life stages of the tools or workpieces, such as recycling or refurbishing. Additionally, the life cycle of the CNC machine itself is excluded from the evaluation of the system's economic and environmental impacts.

The data for the three components: the tools, the C45 workpieces, and the energy mix are all sourced from the literature and summarized in Tables 3.1, 3.2 and 3.3. A detailed description of each component is provided below.

Cutting Tools

The selected cutting tool is a solid carbide tool used for face milling and coated with TiAlN/TiN using a Physical Vapour Deposition (PVD) process (Table 3.1). The cutter has a diameter of 5 mm, 4 teeth, and is suitable for machining C45 alloy (Figure 3.3). The cost of the tools is estimated at 40 EUR per tool. This estimate is based on prices from various tool manufacturers, such as Holex [62]. In practice, the actual cost may vary depending on the negotiations between the company and the tool supplier. The impact on Climate Change of the tool is calculated in two parts:

1. The CO_2-equivalent emissions induced by the production of the tungsten carbide tool, going from the extraction of raw materials to the sintering of the tool [55].

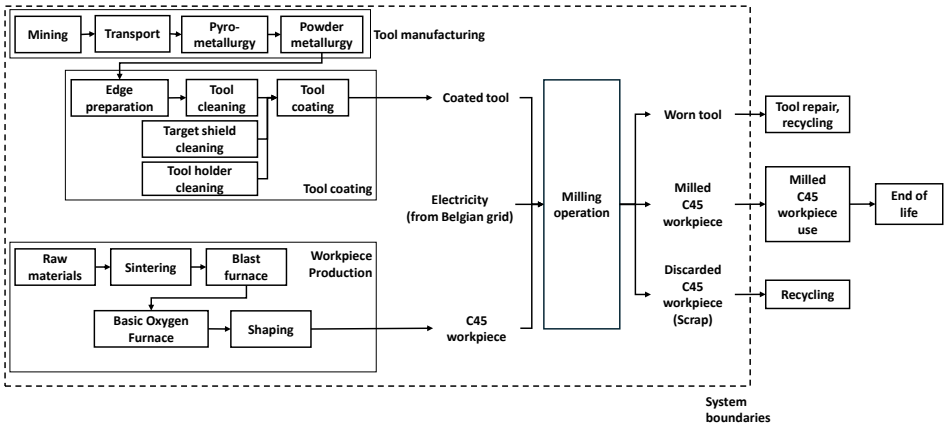


Figure 3.2: Inventory of the inputs, outputs and processes considered in the environmental impact evaluation of the milling operation. Values for CO_2-eq emissions: tool manufacturing [55]; tool coating [56]; electricity from Belgian grid from Electricity maps [57]; workpiece production [58–61].

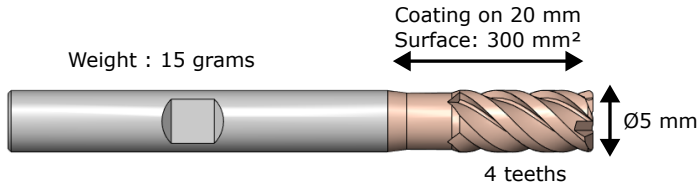


Figure 3.3: Solid carbide tool used for face milling and coated with TiAlN/TiN.

2. The CO_2-equivalent emissions induced by the deposition of the TiAlN/TiN coating on the tool, considering tool edge preparation, tool cleaning, tool holder cleaning, target shield cleaning and tool coating process itself [56]

It is estimated that producing 1 kg of sintered tungsten carbide emits 17 kg CO_2-eq [55]. This type of cutter generally weighs around 15 g, thus the carbon footprint associated to the uncoated tool is 0.255 kg CO_2-eq. On the other hand, CO_2-eq emissions related to the coating deposition process is 0.78 kg CO_2-eq per square millimeter [56]. Based on the general geometry of the cutter, the coated surface area is approximately 300 mm^2. The total CO_2-eq emissions induced by the production of a tool is 234 kg CO_2-eq (as mentioned in Table 3.1).

A tool end of life is assumed to be reached when the flank wear width of the tool's teeth reaches 300 µm (shown in blue in Figure 3.4) as recommended by ISO 8688-2:1989 [28]. Assuming constant cutting parameters, the Mean Time To Failure (MTTF) of cutting tools is estimated to be 20 min. To create random variations around this MTTF, individual tool lifespans are generated randomly from a Weibull distribution with a fixed shape parameter (β) of 7 as suggested by Aramesh et al. [63] (Figure 3.4). The Weibull's scale parameter (η) is calculated from the MTTF through Eq. (3.1).

Table 3.1: Model parameters of the cutting tool.

Cutting tool	
Diameter	5 mm
Number of teeth	4
Coating	TiAlN/TiN (PVD)
Coated surface	300 mm^2
Mass	15 g
Cost	40 EUR
CO_2-eq	234 kg CO_2-eq

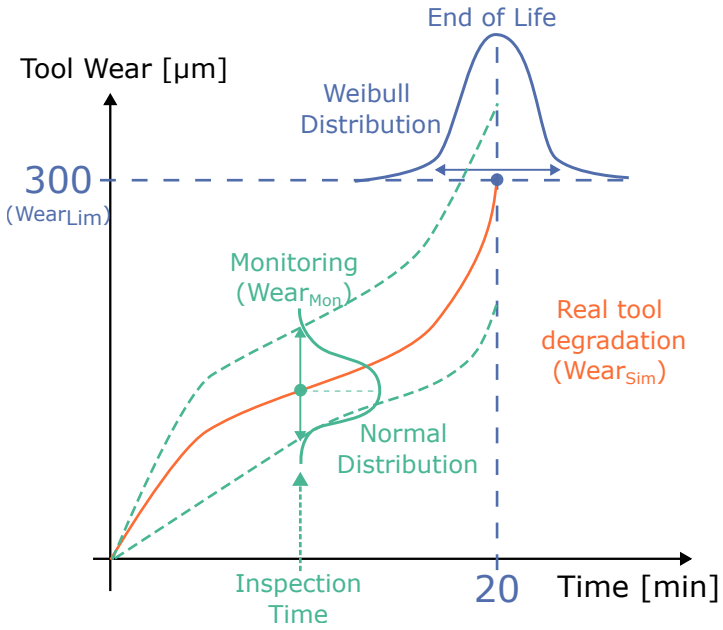


Figure 3.4: Cutting tool degradation simulated in the model. The end-of-life is modelled using a Weibull distribution (blue). The orange curve is simulated using a hyperbolic sine (sinh) function. Imperfect monitoring is represented in green and modelled using a normal distribution, illustrating either an overestimation or underestimation of the actual cutting tool wear.

$$\eta = \frac{\text{MTTF}}{\Gamma\left(1 + \frac{1}{\beta}\right)} = 21.38\,\text{min} \tag{3.1}$$

The degradation trajectory (Wear_{Sim}) of the tool from its initial point to its end of life is approximated by the following equation (in orange in Figure 3.4) [64]:

$$\text{Wear}_{\text{Sim}} = c_w \times (\sinh(a_w \times b_w) + \sinh(a_w \times (t - b_w))) \tag{3.2}$$

In equation 3.2, the parameter a_w defines the intensity of the derivative at the inflection point, b_w the position of the inflection point, and c_w is a proportional

parameter. To generate a degradation trajectory, the tool's lifetime is first randomly sampled using a Weibull distribution (an example with a duration of 20 min is shown in Figure 3.4). This sampled lifetime corresponds to the point at which the tool reaches its maximum allowable wear, denoted as Wear_{Lim}. To model the degradation from the initial moment (when the tool has no wear) to the end of its life, Equation (3.2) is used. The parameters a_w and b_w are randomly drawn from uniform distributions ($a \in [0.2, 0.5]$, $b \in [5, 10]$) to reflect the natural variability of the wear process. The parameter c_w is then fitted based on the values of a_w, b_w, and the final point of the trajectory (Wear_{Lim}) in order to complete the degradation curve.

Workpiece

The workpiece is composed of C45 (Table 3.2). It is assumed that its mass before machining is 2.75 kg. It is considered that the machining operation will remove around 42 % of the workpiece mass, resulting in a finished product weighing 1.58 kg after 2 min and 45 s of machining. The cutting duration is inspired from the turning test as described in Chapter 6.

Table 3.2: Model parameters of the workpiece.

Workpiece	
Alloy	C45
Raw mass	2.75 kg
Useful mass	1.58 kg
Production time	2 min 45 s
Cost	10.58 EUR
CO_2-eq	5.5 kg CO_2-eq

Based on the current price of C45 alloy, the cost of the raw material is estimated at 3.85 EUR/kg [65]. The production of 1 kg of this type of steel emits between 1.5 to 2.5 kg CO_2-eq [58–61]. Therefore, it is estimated that the production of 1 kg of C45 produces 2 kg CO_2-eq. In the model, the workpiece weighs 2.75 kg; therefore, the production of the workpiece emits 5.5 kg kg CO_2-eq, with a cost of 10.58 EUR. The manufactured workpiece can either be good or scrap depending on the condition of the cutting that was used to produce it. For a workpiece to be considered scrap, it must be machined by a tool whose actual wear (Wear_{Sim}) exceeds the limit of 300 µm at any time during machining (i.e., if $\text{Wear}_{\text{Sim}} > 300$ µm at the end of machining).

Machine

The machine considered is a CNC well suited for machining C45. The model considers only the machine in its active machining state. The energy consumption of such a machine includes not only the machining process itself but also all auxiliary systems required for lubrication, cooling, and other functions. Although energy consumption can vary depending on the specific machining operations,

for the purpose of simulation and simplicity, the machine's power consumption is assumed to be constant and approximated at 15.5 kW (Table 3.3). This value represents a compromise between the machining-phase consumption of both small (7 kW) and large machines (24 kW), as described by [66]. It is considered that the carbon footprint of the Belgian electricity production mix is equal to 0.2 kg CO_2-eq/kWh [57]. In this context, it is also assumed that the machine is capable of automated tool replacement; therefore, the tool replacement duration is assumed to be negligible over the simulated period.

Table 3.3: Model parameters of the machine.

Machine	
Mean electrical consumption (1 hour of machining)	15.5 kWh
Cost	0.15 EUR/kWh
CO_2-eq	0.2 kg CO_2-eq/kWh

In all scenarios, the machine is assumed to be a modern CNC system equipped with various sensors suitable for tool condition monitoring. These may include, for example, built-in power sensors that monitor the machine's energy consumption during operation to estimate tool wear [33]. The machine and the monitoring system economic and environmental costs are considered to be out of the scope of the model. This assumption is justified by the fact that, compared to the cost of cutting tools and raw materials, such machine and monitoring systems represent an upfront investment that can be amortized over time [67, 68]. Therefore, their impact on the overall economic and environmental assessment of one workpiece is not considered.

3.2.2 Simulated Scenarios

The model is evaluated under three scenarios, each varying the tool replacement criterion (Figure 3.5):

- Scenario 1: In this scenario, the tool can be replaced between the production of two parts. The replacement criterion is based on the tool's machining time (T_{Mach}) and is compared to a predefined replacement time (T_R). If the T_{Mach} is greater than T_R then the tool is replaced (Figure 3.5). This is the industry standard for tool replacement.

- Scenario 2: This scenario involves monitoring tool degradation. The literature on the subject shows that, on average, tool wear monitoring techniques can achieve a Mean Absolute Percentage Error (MAPE) of approximately 15 % [69] (See Chapter 7). To simulate the monitoring, a Normal distribution with a standard deviation equivalent to a MAPE error of 15 % is used to create an estimate of the wear ($Wear_{Mon}$) (Figure 3.4). It is also observed that monitoring methods either overestimate or underestimate the wear error.

For each tool, the monitoring randomly selects between overestimating and underestimating the state. The replacement criterion is thus based on the monitored value of wear (Wear_{Mon}). As in scenario 1, the tool can only be replaced between the production of two parts. If Wear_{Mon} is greater than the determined wear limits (Wear_{Lim}) the tool is changed (Figure 3.5).

- Scenario 3: This scenario builds on the assumptions of Scenario 2, but with a key difference: the tool can be replaced at any given time. A monitoring system is simulated and inspects tool degradation once per minute (Figure 3.5). This means that the manufacturing process can be briefly interrupted for an automated tool replacement and then resumed to continue the machining operation.

For each scenario, the economic and environmental impacts of the tools, materials, and electricity are compared. These impacts are then normalized to the production of a good part through Eq. (3.3):

$$\text{Impact} = \frac{\text{Impact}_{\text{Good}} + \text{Impact}_{\text{Scrap}}}{\text{Number of Good parts}} \tag{3.3}$$

Where Impact is the average equivalent economic (EUR) or environmental impact (in kg CO_2-eq) of a good part produced in the simulation run, taking into account the scrap produced in order to achieve the production of good parts. $\text{Impact}_{\text{Good}}$ is the total economical (EUR) or environmental (in kg CO_2-eq) impact of all good parts produced in a simulation run, and $\text{Impact}_{\text{Scrap}}$ is the total economical (EUR) or environmental (in kg CO_2-eq) impact of all bad parts produced in a simulation run.

3.3 Results

This section compares the model with the scenarios presented previously. First, to ensure a proper comparison, the process of selecting the optimal scenario is described. Once each scenario has been optimized, the optima results are then analyzed from an economic and environmental point of view.

3.3.1 Optimization of Scenarios

Each scenario is optimized to identify the best-performing parameter values and to enable comparison across strategies. In Scenario 1, the optimization parameter is the tool replacement time. For instance, if this time is set to 6 minutes, the tool is replaced whenever a workpiece change occurs after 6 minutes of tool usage. In Scenarios 2 and 3, the optimization parameter is the monitored wear threshold that triggers tool replacement. In Scenario 2, this wear value is checked at each workpiece change. In Scenario 3, tool wear is estimated once per minute and as soon as the estimated wear exceeds the defined threshold, production is stopped and the tool is replaced. To evaluate each parameter setting, the simulation models one month of production under the corresponding conditions, generating outputs such as the number of parts produced, the number of tools used, and other relevant

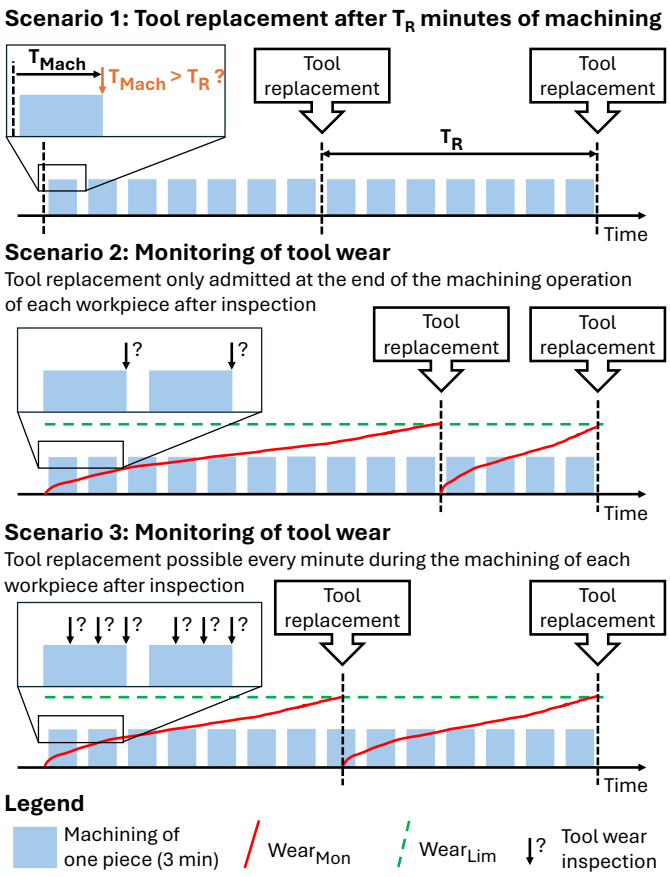


Figure 3.5: Timeline of all scenarios. T_{Mach}: tool lifetime at a given inspection time. T_R: predefined tool replacement time. $Wear_{Mon}$: monitored tool wear at a given inspection time. $Wear_{Lim}$: predefined tool wear limit.

metrics. To account for variability, each simulation is repeated ten times. The results are then averaged, and the standard deviation is calculated for each output value.

Figure 3.6 illustrates the optimization results. It combines the optimization of Scenario 1 in terms of replacement time and Scenarios 2 and 3 in terms of monitored wear values. The solid line represents the cost, while the dashed line indicates the CO_2 emissions associated with producing one good part. The shaded area around these lines shows the 95 % confidence interval (CI) obtained from 10 simulations. This number of scenarios was chosen because increasing the number of simulations does not affect the confidence interval. This figure highlights the economic and environmental impacts of each scenario.

From an economic perspective, Scenario 1 performs worse than Scenario 2, which in turn is outperformed by Scenario 3. A similar trend is observed for environmental impact: Scenario 3 is the most favorable, followed by Scenario 2,

then Scenario 1. The figure also highlights a key insight: the economically optimal point does not align with the environmentally optimal one. This discrepancy arises because the economic impact of a tool relative to the material is less significant than its environmental impact. Consequently, scrapping a functional tool may be economically advantageous compared to the potential production of scrap parts. However, from an environmental perspective, given the tool's greater environmental footprint, the optimal approach prioritizes maximizing its use, even if it leads to some scrap, as the material's environmental impact remains lower. This leads to an environmental optimum that differs from the economic optimum. In Scenario 3, this effect is so pronounced that the environmental optimum occurs even after the tool reaches its maximum allowable wear. It should also be noted that the curve shown in Figure 3.6 is a zoomed-in view. At first glance, one might think that the environmental impact remains stable above a certain value, but this is not the case. When zooming out, a bell-shaped curve clearly appears, emphasizing the negative effect of replacing the tool too late on CO_2 emissions.

In this theoretical case study, the tool's lifespan and the monitoring model's error are both known. In practice, however, while the tool's lifespan can be estimated through trial and error, assessing the accuracy of the monitoring system is much more challenging. Therefore, in a real-world scenario, one would likely choose a replacement time of around 18 minutes for Scenario 1, knowing that the tool's actual lifespan is approximately 20 minutes. This approach favours caution, which is a reasonable strategy, as shown in figure 3.6. In contrast, for the monitoring system, since the exact error is unknown and only the upper limit is known to be 300 µm, a similar safety margin would be applied. One would select a threshold below this 300 µm limit to ensure reliability. For the remainder of this document, and to enable a meaningful comparison between scenarios, we will use the economic optima from Scenarios 2 and 3 as reference points. It is important to note, however, that these values are more difficult to determine in practice than a simple estimate of tool lifespan.

3.3.2 Optimal Results Comparison

To facilitate comparison, the economic optimum of each scenario is selected as the baseline for evaluating differences across scenarios.

Table 3.4 presents a comparison of production outcomes across the different optimized scenarios over a one-month production period. It includes the number of good-quality workpieces produced, the amount of scrap generated, the number of tools used, as well as the total and per-piece economic and environmental costs for each scenario. The percentage is used to compare the average economic and environmental costs relative to Scenario 1, which serves as the baseline (100 %). Scenario 1 results in the lowest number of good-quality parts produced and the highest tool consumption. This is reflected in its economic and environmental cost per part, which is the highest among the three scenarios. In contrast, Scenarios 2 and 3 achieve higher production volumes compared to Scenario 1, although they require more tools for scenario 2. While, for this scenario, this leads to higher overall economic and environmental costs, the efficiency gains translate into lower costs per good-quality part. Specifically, Scenario 2 reduces the cost per part

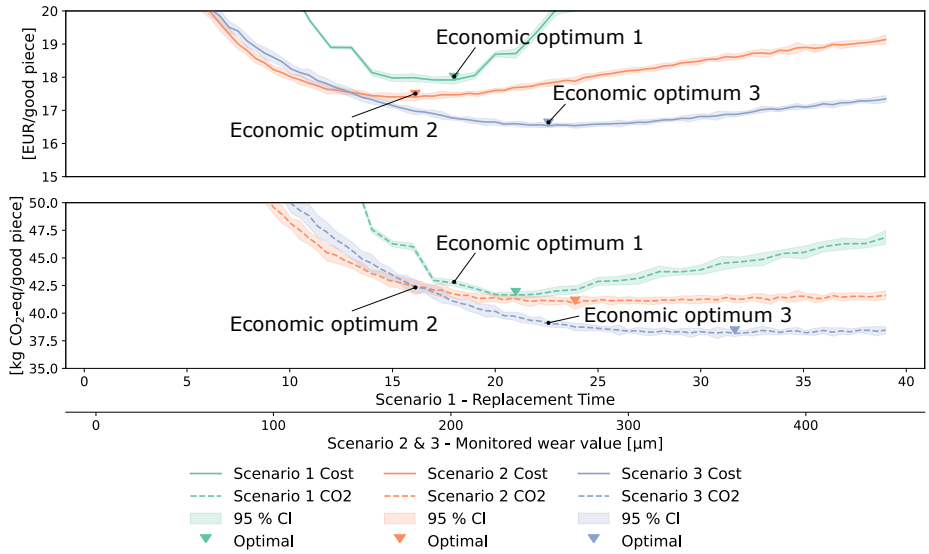


Figure 3.6: Results for the different scenarios presented showing the cost and CO_2-eq of the different scenarios. The confidence interval estimation is computed from 10 simulations and the optimal points are highlighted with dots. The optimal scenarios are pointed by a dot.

by 2.9 % and CO_2-eq emissions by 1.0 %, while Scenario 3 achieves even greater improvements, with a 7.8 % reduction in cost and an 8.7 % decrease in emissions per part.

Table 3.4: Optimal results for each scenario over a one-month production simulation. Standard deviations are calculated based on 10 repeated simulations under identical conditions. Percentages are calculated in relation to the average value obtained in scenario 1.

	Scenario 1	**Scenario 2**	**Scenario 3**
Global Results for the Whole Simulated Period			
Number of good workpieces	9584.2 ± 33.0	$10\,010.3 \pm 16.6$	$10\,310.0 \pm 12.4$
Number of scrap	887.4 ± 34.5	461.9 ± 15.2	158.5 ± 11.5
Number of tools	1496.5 ± 1.1	1558.4 ± 14.9	1466.2 ± 5.3
Total Cost [1000 EUR]	171.7 ± 0.1	174.2 ± 0.5	170.4 ± 0.2
Total CO_2 [1000 kg CO_2-eq]	409.0 ± 0.3	423.9 ± 3.4	402.2 ± 1.2
Results for the Production of One Good Part			
Cost/good part [EUR]	17.91 ± 0.06 (100 %)	17.40 ± 0.06 (97.1 %)	16.53 ± 0.02 (92.2 %)
kg CO_2-eq/good part [kg CO_2-eq]	42.7 ± 0.1 (100 %)	42.3 ± 0.3 (99.0 %)	39.0 ± 0.1 (91.3 %)

Figure 3.7 illustrates the average production cost per good-quality part across the different scenarios. The total cost includes contributions from tools, materials,

and electricity. While tool and material costs vary between scenarios, electricity costs remain low and constant throughout. Compared to Scenario 1, Scenario 2 benefits from lower material costs, as it generates less scrap. Scenario 3 further reduces both tool and material costs by using fewer tools and producing even less scrap, resulting in the most cost-efficient outcome among the three scenarios.

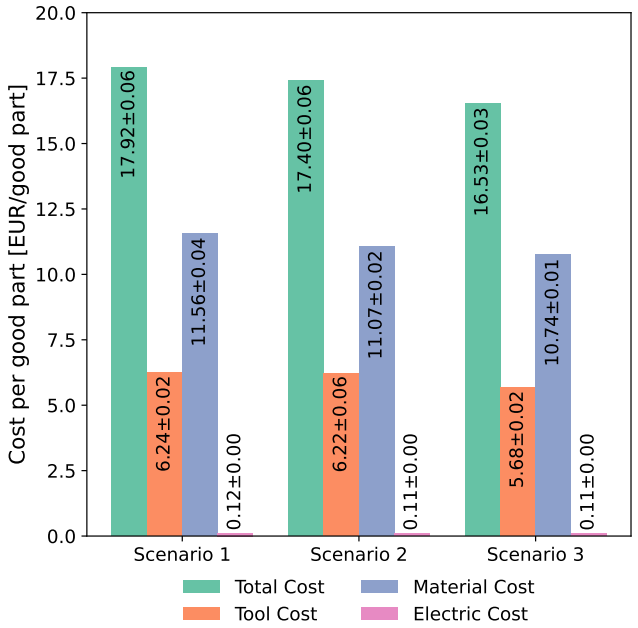


Figure 3.7: Cost repartition to produce a good part in the different tool replacement scenarios. The text in the bar correspond to the mean and standard deviation. The deviation is computed on 10 simulations. This deviation is not represented as it is too small to be seen.

Figure 3.8 shows the average environmental impact of producing a good part across different scenarios. As before, this impact is calculated based on the cutting tool, the material, and the electrical energy consumed. Similar conclusions to those observed in Figure 3.7 can be drawn. The introduction of cutting tool monitoring has a beneficial effect on the environmental impact. In this case study with coated tools, the environmental impact of the tool is very significant. Therefore, optimizing the number of tools is crucial to reducing the environmental impact of production, as the tool is responsible for around 85 % of the emissions in producing a part.

3.4 Conclusions

The scenarios modelled in this study demonstrate that cutting tool management has a significant impact on both the production cost and the environmental footprint. The results indicate that replacing cutting tools based on their machining time does not optimize the economic cost and environmental impact of manufacturing

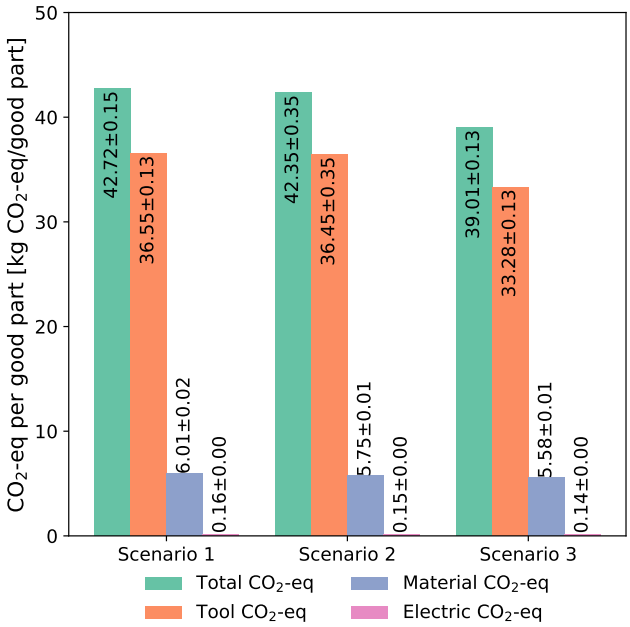


Figure 3.8: CO_2 repartition to produce a good part in the different tool replacement scenarios.The text in the bar correspond to the mean and standard deviation. The Deviation is computed on 10 simulations. This deviation is not represented as it is too small to visualise.

parts as effectively as monitoring the tool's degradation state. Although the quality of the wear monitoring in the simulations is intentionally made imperfect to match current monitoring accuracies, the impact on production is still positive. While directly monitoring tool wear remains a technical challenge, the results presented in this study highlight the substantial benefits that monitoring could offer. In such a highly competitive sector, implementing tool condition monitoring could lead to production cost savings of up to 7.8 %

The simulation results also show the benefit of using cutting tool degradation monitoring from an environmental perspective with a reduction in CO_2 emissions by nearly 8.7 %. In the current environmental and political context, this further justifies adopting tool condition monitoring methods in industrial practice. This environmental impact is mainly due to the manufacturing of cutting tools, especially due to the application of their coating, which accounts for more than 90 % of a tool's environmental impact [56].

Although the case study is a simplification, the relative improvements observed between the scenarios remain significant. In real applications, these differences could be amplified or reduced by additional factors beyond the machining itself, such as part cost, supply chain considerations, other operations on the workpiece or the criticality of the machining operation.

The results clearly show that adopting a smarter replacement strategy, beyond

a simple time-based approach, offers real benefits. Given the high variability of tool wear, relying purely on machining time does not provide an optimal result. This highlights the importance of implementing tool degradation monitoring methods and serves as a key motivation for the subject of this thesis.

Chapter 4

State of the Art of the Monitoring or Forecasting of the Cutting Tool Degradation

The state of the art is based on a Systematic Literature Review (SLR) published at the end of 2021, which covers research from 2000 to 2020, with a particular focus on turning processes [33]. This chapter is derived from that SLR, with an additional section included to assess developments in the literature since early 2022. Although the SLR primarily targets turning, similar trends have been observed in milling.

4.1 Introduction

Monitoring the degradation of cutting tools is a long-standing and broad research topic. Over the years, numerous methods have been developed to monitor tool wear, with varying degrees of success. Initially based on empirical observations [70], these approaches evolved into statistical methods [6], and for the past two decades, AI-based techniques have emerged. These AI-driven methods have largely supplanted earlier approaches due to their adaptability and effectiveness in tool condition monitoring.

However, the literature on this subject remains highly fragmented, making it challenging to gain a clear overview of the methods used and their relative effectiveness. The diversity of approaches further complicates the task of identifying consistent trends or best practices.

To address this, a Systematic Literature Review (SLR) was conducted to establish the state of the art in AI-based cutting tool degradation monitoring. The review focuses primarily on turning operations, in order to limit the scope and

avoid the complexity of comparing across different machining paradigms, while still capturing meaningful trends in the literature.

Given the broad nature of the topic, a SLR methodology was followed. This approach aims to define a clear and rigorous review protocol, ensuring a comprehensive and efficient exploration of the research landscape. Unlike traditional literature reviews, a SLR includes all relevant studies that meet predefined criteria, providing a more detailed and objective view of the field and its evolving trends.

In the context of this thesis, the SLR was conducted at the beginning of the research to gain a solid understanding of the topic and to identify gaps in the existing literature. This chapter presents the main findings of that review. Since the original SLR covered publications from 2000 to 2020, an additional section has been included to highlight developments in the literature since then. This chapter is therefore structured as followed:

- Section 4.2 presents the review protocol followed in the SLR. This protocol adheres to the guidelines for conducting systematic literature reviews as recommended by Kitchenham et al. [71, 72], similar to the "Preferred Reporting Items for Systematic Reviews and Meta-Analyses" (PRISMA) framework [73].
- Section 4.3 provides a bibliometric analysis of the reviewed literature, highlighting key trends observed in the field.
- Section 4.4 offers a more in-depth examination of the approaches identified in the SLR. To maintain conciseness, only the Artificial Neural Network (ANN) and Convolutionnal Neural Network (CNN) approaches are discussed in detail, while other methods are described in the appendices of this thesis.
- Sections 4.5 to 4.7 discuss the main findings of the SLR and their implications.
- Section 4.8 outlines how the literature has evolved since the completion of the SLR.
- Finally, Section 4.9 explains how the results of the SLR are leveraged throughout the rest of this document, with a focus on identifying and addressing gaps in the literature.

4.2 Review Protocol

This SLR was conducted following the review protocol suggested by Kitchenham et al. [71, 72]. The definition of a review protocol is a critical step because it ensures the repeatability of this review and reduces the researcher bias in the presented results. This protocol is composed of six stages with the objective of defining the framework of this SLR:

1. **Research questions** (Section 4.2.1). This step consists of defining the research questions that are addressed in this SLR.
2. **Search process** (Section 4.2.2). In this step, the methodology to obtain relevant literature entries is described.

3. **Inclusion and Exclusion criteria** (Section 4.2.3). The literature obtained from the previous step needs to be sorted in order to obtain only the literature entries that answer the research question. Therefore, in this step, paper selection criteria are defined in order to best answer the research question based on the title and abstract of the articles.

4. **Quality assessment** (Section 4.2.4). In this step, the content of the selected papers is analysed, and quality criteria are applied to reject papers that do not answer the research question.

5. **Data collection** (Section 4.2.5). In this step, the researchers extract the useful information from the selected studies.

6. **Data analysis** (Section 4.2.5). Finally, this step allows the presentation of the results in a relevant and comparable way.

All six stages of this protocol are exposed in the remainder of Section 4.2. Figure 4.1 presents a summary of the research protocol with the corresponding results.

4.2.1 Research Questions

As presented in the introduction, the interest of AI techniques applied in manufacturing is a constantly growing subject. This subject is in fact too broad to be treated in, a single review. In consequence, the main research question of this review is intentionally limited to the use of AI techniques to monitor the tool wear in stable turning operations. The research question is expressed as:

RQ1 How is the AI used in the framework of monitoring/predicting the condition of tools in a stable turning condition?

The formulation of this research question aims to precisely define the framework of this chapter. The concepts of AI and ML have been described in Chapter 2. The words "monitoring/predicting" are used to encompass all monitoring cases, i.e., in real time or with a prediction of the future state of the tool. As many indicators can be used to monitor the tool, the word "condition" is used to encompass them all. The formulation "stable turning condition" aims to reject all papers that discuss chatter or tool breakage detection. As this research question mainly focuses on AI methods applied to the health of the tool, quality-oriented approaches which focus on workpieces are not considered in this review.

The research question is voluntarily broad, such that some sub-questions can be addressed:

RQ2 What are the most used AI techniques used in this context ?

RQ3 What is the forecasting horizon of each of the identified AI techniques in this context ?

RQ4 What are the most common inputs used with the different identified AI techniques in this context ?

RQ5 What are the most used feature extraction techniques used in this context ?

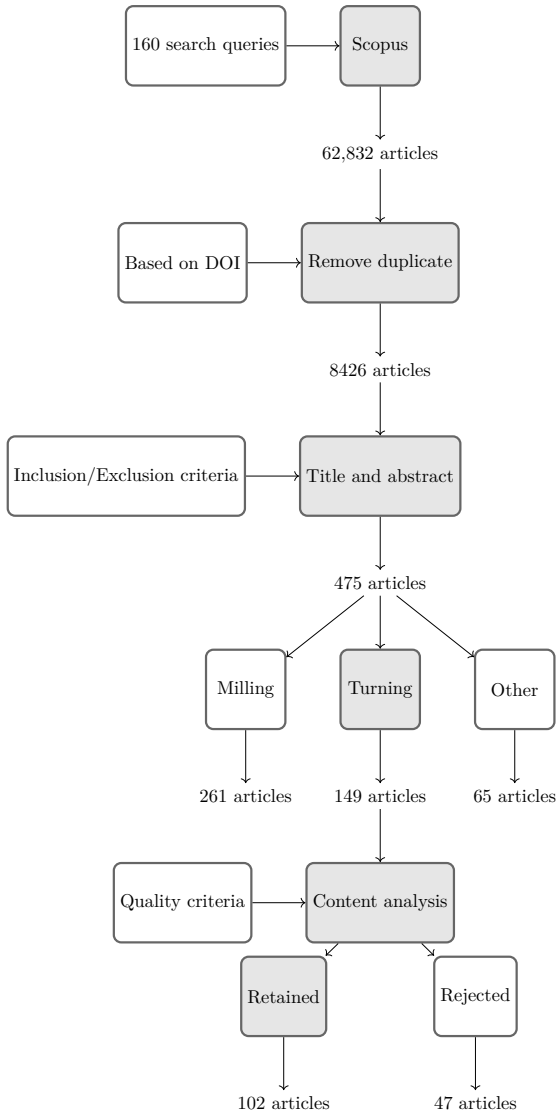


Figure 4.1: Methodology pursued to identify the 102 articles.

These research questions are answered throughout the SLR analysis process, and they are explicitly answered in Section 4.6.

4.2.2 Search Process

4.2.2.1 Database

To answer the research question, adequate databases should be consulted. The database must be as large as possible and related to the subject to have a representative panel of articles constituting the current state of the art. In this review, it was chosen to

use the *Scopus* database as it regroups a large number of relevant journals from various publishers [74]. Due to the evolution of AI techniques and computing power, it was considered that articles dating from before 2000 were not representative of the current state of the art in the frame of this SLR. Thus, it was chosen to limit the search process over a 20-year period from January 2000 to December 2020. Furthermore, other studies also stated that before 2000 the vast majority of AI techniques used was ANN [36]. A literature review made in 2002 [75] already addresses this part of the literature [75].

4.2.2.2 Keywords

The search terms must be chosen to be the most general in the framework of the research question. It ensures that the research addresses the topic without introducing bias. The keywords used to perform the search in the *Scopus* database are listed in Table 4.1. A semantic approach was chosen such that each search query is constructed with one element of each semantic part. It was chosen to compose the search query with four parts:

1. the tool part;
2. the aims part;
3. the method part;
4. the indicator part.

For the "tool part" in Table 4.1, the word cutting tool uses the wildcard character (*), which is used to include these different forms of the "tool" word: "tool", "tools", and "tooling". A total of 160 research queries is constructed with this method, creating all possible combinations with the desired composition. The keywords used to build the search query are voluntarily broad and include all machining processes. The main advantage of this approach is that it allows for going through all the possible combinations, but it creates a significant amount of "duplicate" results which must be removed. This last step can easily be realized automatically.

As mentioned previously, the research was done in the *Scopus* database, each combination of keywords is entered individually, and the results are obtained with the "search within all fields" option. For the results of each piece of research, *Scopus* provided the following informations: Title, authors, year, Digital Object Identifier (DOI), abstract, document type, publisher, number of citation, ...

From the 160 search queries, 62,832 articles were obtained. The duplicates were removed based on their DOI. This lead to 8426 unique articles. On average, every search made with this approach yielded 393 results. The most successful search query yielded 3180 results and was

- "machining" AND "prediction" AND "artificial intelligence" AND "data"

With only two results, the least successful search was

- "cutting tool*" AND "prognosis" AND "intelligent data analysis" AND "data"

Table 4.1: List of keywords. Wildcard character (*) includes these different forms of the "tool" word: "tool", "tools", and "tooling".

Tool Part	Aims	Method	Indicator
Cutting tool*	Prediction	Artificial intelligence	Wear
Machining	Monitoring	Machine learning	Remaining useful life
	Reliability	Intelligent data analysis	Condition
	Prognosis	4.0	Data
	Assessment		

From the 160 combinations of keywords, 19 combinations gave more than 1000 results. From these 19 combinations, it is possible to generalize a search query that would yield most results into this form:

- "machining" AND ("prediction" OR "monitoring") AND ("artificial intelligence" OR "machine learning") AND ("condition" OR "data" OR "wear")

4.2.3 Inclusion and Exclusion Criteria

From the previous step in the procedure, 8426 articles are identified. Not all these articles are relevant to answer the research question. To assess the eligibility of each article, it has to be compared with some inclusion and exclusion criteria. These criteria are chosen in accordance with the perspectives of this review and try to be inclusive enough to find all the articles of interest while clearly delineating the research question. These criteria are listed below.
Inclusion Criteria (IC):

IC1 The paper describes the application of an AI method applied to cutting tools.

IC2 The paper is about turning operation. This information must be clearly stated in the title or the abstract.

IC3 The paper is written in English.

IC4 The AI must provide information on the tool condition.

IC5 The paper has been published between January 2000 and December 2020.

Exclusion Criteria (EC):

EC1 The paper must not include instabilities/chatter in the machining process.

EC2 The paper must not be a review.

Based on titles and abstract of the identified articles, the inclusion and exclusion criteria were applied to obtain 475 results. At this stage, the inclusion criteria IC2 was not applied to obtain an overview of the application of AI techniques in different machining processes. From the 475 results, 149 articles are in turning, 261 are in milling, and 65 in other machining processes (Figure 4.1). These relevant

articles are spread across all search queries; in other words, they are distributed uniformly over the 8426 articles identified. It confirms that the semantic approach used to build the search terms was successful. In the following steps, the criteria IC2 is applied such that only articles about turning are analysed.

Due to the large number of articles and the risk to introduce some bias in the sorting previously described, this stage was performed in parallel simultaneously by two researchers. After the independent rating phase, in case of disagreement, a discussion took place to make a final decision for these articles as proposed by Kitchenham [72]. As it is a manual sorting step, it can introduce some bias into the choice of the selected article depending on the appreciation of the researchers. To measure the inter-rater reliability, the Cohen's kappa statistic (κ) was used [76]. This indicator measures the agreement among raters and is computed as:

$$\kappa = \frac{P_o - P_e}{1 - P_e} \tag{4.1}$$

with P_o (observed value: 0.93) the observed agreement and P_e (observed value: 0.64) the probability of random agreement. After sorting, the obtained κ value was 0.8, which indicates that the raters were in "almost perfect agreement" [76]. This means that the defined criteria were applied in the same way by both researchers. This ensures that the vast majority of relevant articles dealing with the subject of this SLR have been identified and provides quality assessment of the sorting process

4.2.4 Article Quality Assessment

The previous steps identify 149 articles about turning (Figure 4.1). These articles were selected based on their title and abstract, but their contents may still not correspond to the research question of this SLR. The quality of the content of the 149 articles is compared with quality criteria that helps to answer the research questions. The quality assessment is often used to assess differences in the execution of studies [72]. In this SLR, this list of quality indicator is proposed:

QA1 The paper correspond to the inclusion criteria.

QA2 The paper presents a clear data input description.

QA3 The origin of the data are clearly defined.

QA4 The paper clearly describes its methodology.

QA5 The pre-processing on the data are explicitly mentioned (if any).

QA6 The paper presents an indicator for the quality of their results.

If two or more of these criteria are not met, the paper is rejected. Based on the quality indicators, the rejected articles are articles that do not contribute to answering the research question or do not provide enough information to be fairly compared with others. The analysis of content and the application of these criteria results in 102 articles retained and 47 articles rejected (Figure 4.1).

4.2.5 Data Collection and Data Analysis

By following the procedure previously described, 102 articles are selected to be presented in the SLR (Figure 4.1). In what follows, two types of analyses are performed:

1. A descriptive analysis (Section 4.3). This part discusses the trends in the literature and shows the state of the art on the research question. Bibliometric information such as year of publication and corresponding approaches are discussed.
2. An in-depth analysis of the AI techniques used (Section 4.4). A detailed analysis of the content of the articles is carried out and presented in the form of tables. Each category of AI is discussed individually and relevant information in each category are presented. Criteria such as input data, pre-processing, and output are discussed.

4.3 Bibliometrics Analysis

From the 102 articles identified with the procedure described above (Section 4.2), a bibliometric analysis is performed. The aim of this analysis is to present the trend in the literature and give a meaningful picture of the state of the art. As described above, this analysis focuses on the 102 previously identified articles from the last 20 years in the context of AI applied to cutting tools in turning operation.

4.3.1 Distribution in Time

The distribution of the number of publications per year allows for finding if a subject is of interest among researchers. This makes it possible to show trends over the years and reflects the research interest in a subject. Knowledge of these trends allows researchers to align their efforts in particular topics to solve current technological challenges.

The distribution of publications between 2000 and 2020 is presented in Figure 4.2. It can be observed that there is a steady growth in the number of articles which reflects the growing interest in the use of AI in the industrial sector. This may be related with the evolution of AI techniques and also the amount of data generated by machine tools in the framework of industry 4.0. From Figure 4.2, it is possible to divide the timeline into three identical periods of seven years: from the period of 2000 to 2006, there is an average of 2.85 articles per year, from the period of 2007 to 2013, there is an average of 4.14 articles per year and from the period of 2013 to 2020, there is an average of 7.57 articles per year. It is observed that the year 2020 was particularly rich in the number of publications which may indicate the rise of a hot topic in the literature.

4.3.2 Repartition of AI Type

From the content analysis of the selected papers, seven “types” of AI stand out. The categorization is the reflection of the occurrence of these techniques in the literature:

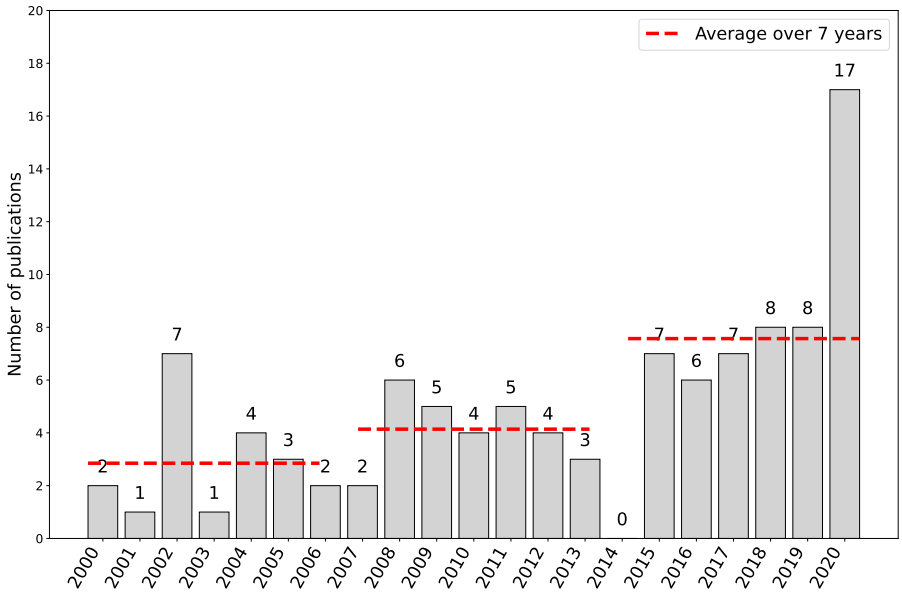


Figure 4.2: Number of publications per year in turning. The average values are computed over 7 years.

1. Artificial Neural Network (ANN).
2. Fuzzy approach: "Fuzzy Inference System" (FIS) and "Adaptative Neuro Fuzzy Inference System" (ANFIS).
3. Support Vector Machine (SVM)
4. Self Organizing Map (SOM).
5. Hidden Markov Model (HMM).
6. Convolutional Neural Network (CNN).
7. Other. This category regroups techniques that individually are not sufficiently represented to have their own category. It includes AI techniques such as: decisions trees, K-nearest neighbour classifier, etc.

These different techniques are the subject of an in-depth analysis in Section 4.4, including tables summarizing the review results. To be exhaustive and take into account articles that in addition to the AI approach addressed a non-AI approach, the category "other (not AI)" has been added.

The repartition of the type of AI used to monitor the tool wear is presented in Figure 4.3. From this figure, it is observed that the majority of articles focuses on the following AI techniques: ANN, Fuzzy and SVM. These three techniques represent about 60 % of all articles. The other techniques, i.e., SOM, HMM, CNN and "other" together represent the remaining 40 % of the literature.

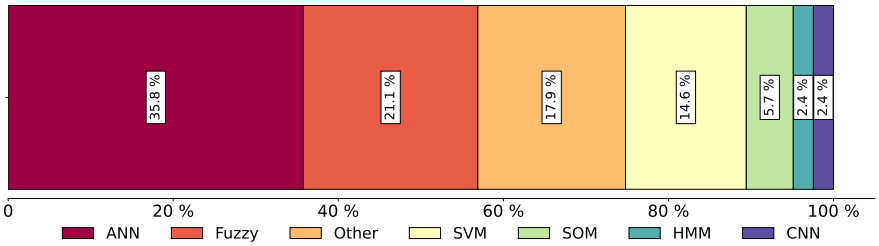


Figure 4.3: Sorted global repartition by type of AI.

The repartition of the type of AI by year is presented in Figure 4.4. It should be noted that some articles discuss several independent AI techniques, which explains why the numbers are not consistent with the repartition by year in Figure 4.2. This figure allows observing some trends in the literature:

- The interest for the ANN approach occupies an almost constant proportion of the literature through the years.
- The SVM approach has seen a growing interest in the last five years.
- The fuzzy logic approach aroused great interest up to 2011, but it seems less represented in recent years.
- Techniques such as HMM and SOM are not sufficiently represented to observe any tendencies.
- The CNN approach was first observed in 2020. Therefore, no conclusion can be drawn as to its use. It will be necessary to monitor its use in future years to observe if this is a hot topic of ongoing research.
- The other AI techniques have a growing interest too. This may indicate an interest to test new approaches.

The choice to carry out an analysis of the literature over the last 20 years therefore makes it possible to observe the evolution in interest of each of the AI techniques. This allows for identifying all the approaches that have been tried and which approaches are raising the most interest for future trends.

4.4 In Depth Analysis

In the remainder of this chapter, the main focus will be on the ANN and CNN approaches. The other approaches, namely Fuzzy, SVM, SOM, HMM, CNN, and other are described in the Appendix of this thesis. There are two types of analysis:

- The content of the articles is presented in table form presented in the Appendix of this thesis (Tables C.1 and C.8). Note that some articles present several approaches, such that they can appear multiple time in the tables of related techniques.

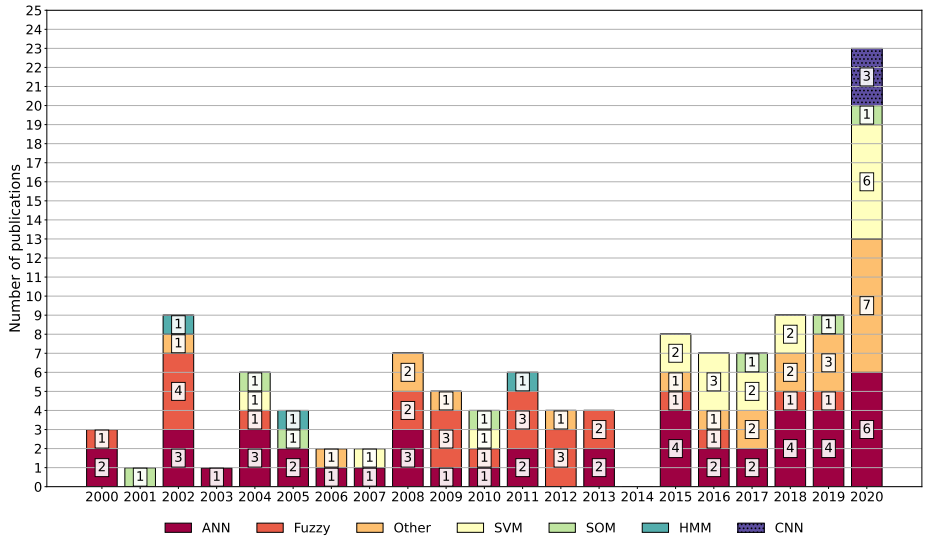


Figure 4.4: Chronological repartition of the type of AI—Note that some articles present several AI techniques.

- A general analysis of the content of the tables for each technique is carried out. In this chapter, the focus will be on the ANN (Section 4.4.1) and CNN (Appendix C.1.5) approaches. The remaining approaches are analysed in the Appendices C.1.1 and C.1.6. For each AI approach, a presentation of the most used: inputs, pre-processing and output is presented in the form of figures that relate the number of article using the corresponding approach. If there is less than four articles for a technique, a short summary of each article is presented. The articles that compare their results with others approach is also listed. Finally, a small conclusion is made highlighting the use of each AI technique.

The in depth analysis for each approach is structured according to the typical sequence of steps involved in applying AI techniques. These steps are the following:

1. **Data collection**. In this SLR, the sensors used to obtain the data are not explicitly presented, as previous reviews covered the subject extensively [77].

2. **Features extraction**. This is the pre-processing of the data, for the sake of this SLR, these techniques have been grouped into different categories:

   a) **Statistical approach**. This approach extracts descriptive statistics such as: mean, kurtosis, skewness, ..., from the temporal signal.

   b) **Frequency approach**. This approach extracts information from the frequency domain. In this SLR, this is commonly achieved by a Fast Fourier Transform (FFT).

c) **Time-frequency approach**. This approach extracts information from the time-frequency domain. In this SLR, this is generally done with a Wavelet Package decomposition.

d) **Normalization**. It consists of scaling the variables between 0 to 1. This allows the AI to learn in a more stable and faster way.

e) **Others**. This regroups specific approaches. In this SLR, the "others" category generally refers to features extracted from images.

f) **None**. An article which does not explicitly describe any pre-processing techniques is found in this category. Note that some sort of pre-processing is always needed.

More detailed descriptions of these approaches exist in the literature [78–80].

3. **Features selection**. To avoid any confusion, this part is not systematically presented in this SLR. Indeed, the articles rarely present the feature selection extensively and, in many cases, the selection is made by the authors. Since little and incomplete information is available in the articles, it is therefore not relevant to compare the articles on this point.

4. **Input** In this review, a categorization of the input data is realized, and the input is categorized as: cutting speed, feed rate, depth of cut, cutting time, spindle information, cutting forces, Acoustic Emission (AE), vibration, surface roughness, images, temperature, and others. The first three inputs listed above are data characterizing the cutting conditions of the turning operation. These data are not correlated with the health of the tool but are usually used to give context to the other measured data. In this study, these data are presented in the "input" column in the different tables of results and figures related to the approach.

5. **Inference** In this SLR, the output is categorized in: classification, estimation of VB, prediction of VB, remaining useful life (RUL) and others. The difference between "estimation" and "prediction" aims at disambiguating the word "prediction" used by authors indistinctly for "estimation" (i.e., "monitoring of the current state of the tool") and "prediction" (i.e., "prediction of the future state of the tool"). This nuance is discussed in Section 4.5.

6. **Accuracy** Comparing the accuracy of different AI techniques is complex, as different authors use different indicators (Root Mean Squared Error, Mean Absolute Error, accuracy term, ...) to evaluate their approach. Categorizing the results would be too subjective as it would be necessary to define precision intervals. As all the approaches are very different and use various datasets, it was chosen to refer to each article authors' assessment to judge the accuracy of the approach. All articles cited in this SLR have obtained a "good" performance. Objectively, the average accuracy of all classification methods is around 90%. As it is discussed in the following sections, the majority of authors that performed a comparison of an AI approach against a standard statistical approach have shown that the AI approach performed better than the statistical approach.

This approach allows for identifying the different trends in the literature for each AI selected techniques. It also allows for highlighting the difficulties and opportunities offered by each approach.

4.4.1 ANN

As presented in the bibliometric analysis (Section 4.3), the artificial neural network is the most popular decision-making method used to monitor the tool life. An explanation of this approach is presented in Chapter 2, and Table C.1 presents the 44 articles about ANN.

4.4.1.1 General Architecture of the Neural Network

The architecture of a neural network is often defined by trial and error and can have a major influence on the quality of the results. From the 44 papers about ANN, 17 propose an architecture with one hidden layer [80–96], two propose an architecture with two hidden layers [97, 98] and the rest does not give any information about the network architecture. From these papers, the most common architecture of the neural network to monitor the tool wear is presented in Figure 4.5. This architecture is composed of three layers: one input layer, one hidden layer and one output layer. The number of neurons in the input layer depends on the data used as inputs. In the hidden layer, the number of neurons is around 3 to 5. Finally, the output layer is generally composed of one neuron that gives information about VB (Figure 4.5). This architecture therefore appears to be the most efficient to resolve the subject of this SLR. Apart from the network architecture, other considerations exist. An exhaustive comparison of different types of NN is presented in [99]. The influence of the optimizer is studying in [100, 101]. A comparison between different architectures and activation functions is made in [83]. It is also worth noting that [102] proposes an approach based on a recurrent neural network.

4.4.1.2 Input

The different features used as input of the neural network, based on the 44 articles on the subject, are presented in Figure 4.6. The six most used features are: cutting speed, cutting forces, feed, depth of cut, vibration and acoustic emission (AE). In these data, only three are dependent on the state of the tool: cutting forces, vibration and AE. The three others are cutting parameters that are necessary to help the neural network to identify the cutting conditions.

4.4.1.3 Pre-Processing

To extract useful information about these signals, pre-processing techniques are employed. Figure 4.7 shows the pre-processing techniques used in this context. As the two most used pre-processing techniques are: None and Statistic, it appears that, for ANN, complex techniques are not always required. When a pre-processing is applied, it is a very simple one that consists of computing the skewness, kurtosis, etc. Note that the category “None” refers to articles that do not present the pre-processing technique used. As some sort of pre-processing is always needed,

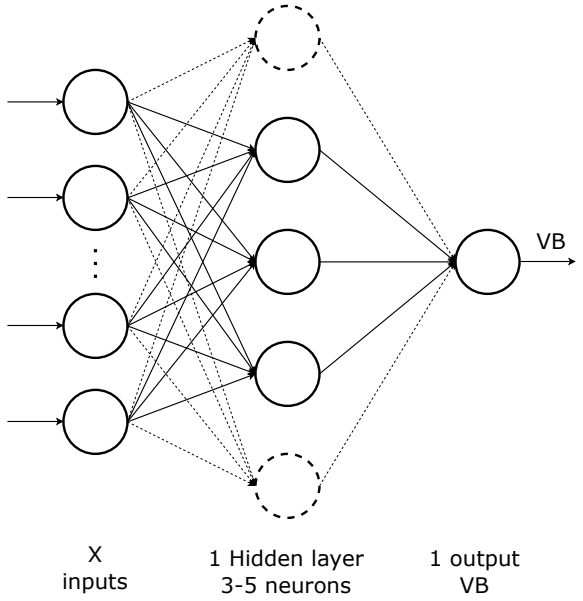


Figure 4.5: General form of an ANN used for tool condition monitoring.

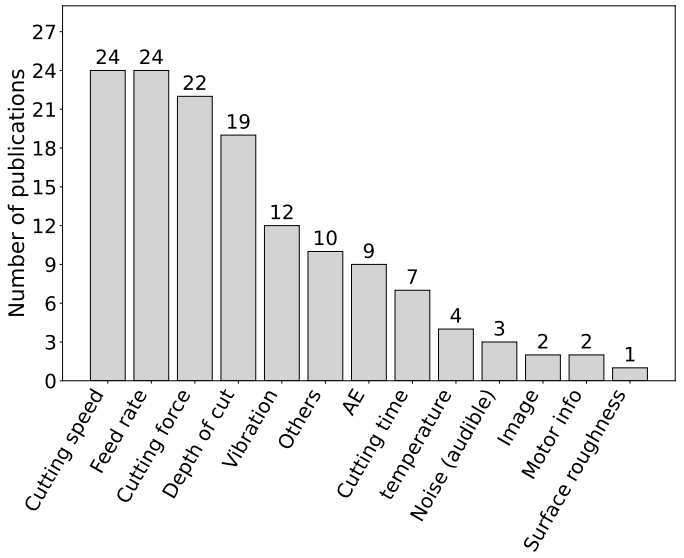


Figure 4.6: Repartition of inputs for ANN techniques.

it indicates that these articles only compute simple statistic value on the "raw" signal from the sensors such that it does not require extensive presentation of the pre-processing technique. This observation indicates that ANN is robust to noisy data such that advanced pre-processing techniques are not needed for all the inputs.

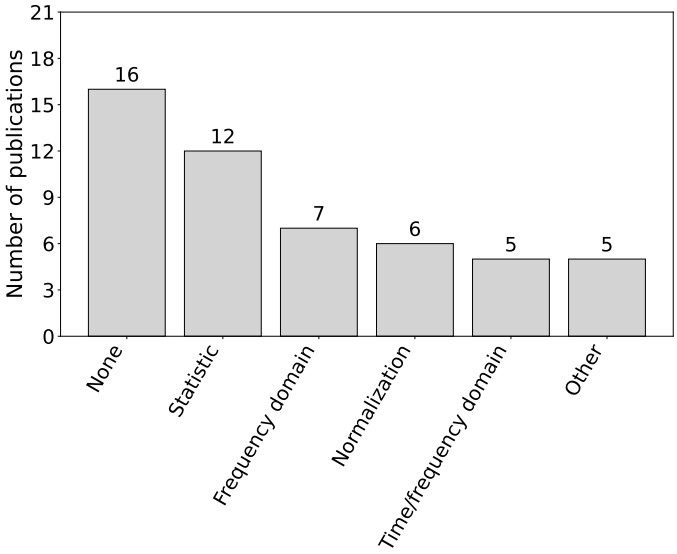


Figure 4.7: Repartition of pre-processing for ANN techniques.

4.4.1.4 Output

The output of the neural network is presented in Figure 4.8. The vast majority of papers use the ANN as a regressor to estimate the value of VB in real time ("estimation of VB" in Table C.1 and Figure 4.8). Some papers also use the neural network as a classifier, and the classification is on average performed for three states. The RUL is used in [103, 104] with the latter giving an upper and lower limit for its prediction. Some articles use a different approach: [85] monitors the percentage of tool life, [105] monitors the tool life and tool-shim interface temperature, [106] monitors the surface roughness to latter predict the flank wear and [87] monitors surface roughness and flank wear.

4.4.1.5 Comparison

Some articles present the results of the ANN by comparing its performance with those of classical regression methods [87, 89, 98, 104, 106–108]. Some of them also completely describe a regression method in their approach [87, 89, 98, 106, 107]. In all these papers, the observation is that ANN performs better than the classical regression methods. The advantage of ANN over the regression model is its ability to continuously learn from the data, making it more efficient at monitoring the tool wear.

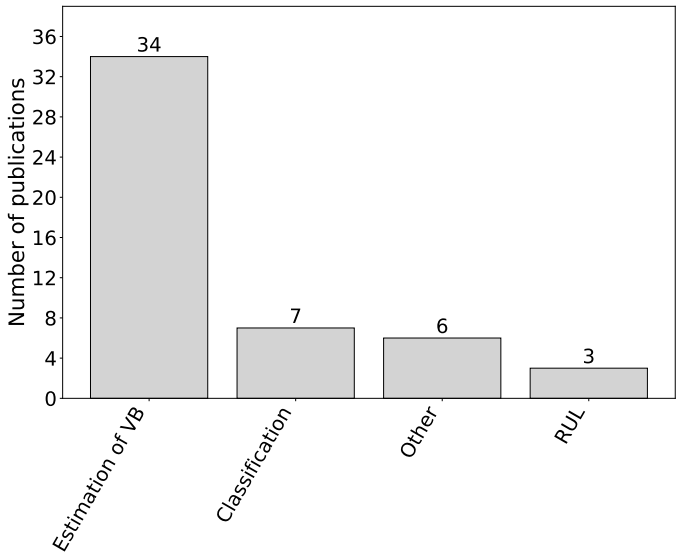


Figure 4.8: Repartition of outputs for ANN techniques.

4.4.1.6 Conclusions

The great interest in this technique can be explained by its history in the framework of tool monitoring. Indeed, before the year 2000, the majority of publications concerning the use of AI applied to cutting tools concerned the use of neural networks [75]. This can be explained by the ease of implementing neural networks and the great interest that this technique received among researchers. Today, many accessible libraries make it possible to code a neural network easily, e.g., in Python, TensorFlow [109] and Scikit-Learn [110] are two commonly used libraries. It is also well known that ANN are efficient to model highly nonlinear models which is our case of interest [35]. One of the major drawbacks of ANN is its "black box" aspect. In today industries, the explicability of AI is a subject of great interest [111, 112] but, by the number of articles about ANN, it appears that this consideration is not an obstacle to the development of this approach.

4.4.2 CNN

The use of CNN is not prominent in the results of this SLR. Only 3 articles discuss the use of CNN to monitor the tool wear: [113–115]. They are presented in Table C.7. All of these articles have been published in 2020 showing a new interest in this techniques (see Section 4.3.2). These articles almost use the same approach.

4.4.2.1 Presentation of the Articles

In [113], the authors used the image of the insert to compute the value of VB. This is achieved by identifying the wear area on the tool. The identification of this zone

makes it possible to count the number of pixels which allows finding the dimensions of the wear zone. The authors perform a data augmentation on the initial dataset to improve the robustness of the CNN. In [115], a similar approach is followed, but the images of the insert are used directly to classify the state of the tool. In this study, the classification is made within two classes: "GO/NO GO". Both articles report accuracy greater than 90%.

Based on the dependency of chip colour on the cutting tool temperature, and the influence of tool wear on tool temperature, it has been proposed to use the chip colour to monitor the tool wear [114]. Images of the chip are pre-processed to remove the background and a kernel density estimation is performed. They show that this density estimation is more efficient on the hue channel from the Hue Saturation Value (HSV) channels of the images. This is used as an input of the CNN which performs a classification around three classes depending on the class of the wear: "New", "Medium" and "High". They achieve accuracy of above 90%. A comparison is made with a functional data analysis (FDA) classifier, and it fails to beat the CNN.

4.4.2.2 Conclusions

The late interest in this technique can be explained by the complexity to obtain a significant number of images for the datasets. Indeed, this technique uses images of the tool which can be impractical to obtain in industrial applications. These approaches work greatly in laboratory conditions but still require human intervention to achieve them. Comparing this approach with the others presented in this paper, automating this approach seems more difficult. In addition, in real production applications, turning is often performed with cutting fluids which can further complicate image capture. Indeed, the conditions for taking images must be consistent and the light must be controlled so as not to influence the neural networks [114]. Today, to our knowledge, there is no automated machine allowing the introduction of a capture system of this type which may explain the low interest raised by this technique. This lack of interest can also be explained by the false belief that CNN is more difficult to implement than ANN [42]. Furthermore, this approach requires stopping the machining process to allow the picture to be taken which removes all the interest of monitoring in real time. However, the advantage of this approach is that it makes it possible to directly measure the wear on the tool rather than going through correlated indicators such as the cutting forces, for example. Automating this approach would be equivalent to automating the measurement of VB, which would provide important data for other forecasting techniques. By using the chip colour instead, it could then be possible to carry out the measurement without interrupting the process [114].

4.4.3 In Depth Analysis – Conclusions

Table 4.2 summarizes the top 3 inputs, pre-processing, and outputs for each AI techniques. From this table, the different methods are compared.

For ANN, 2 of the 3 most used inputs are cutting parameters: cutting speed and feed rate. This extensive use of cutting parameters as input signal appears to

be specific to ANN as this characteristic is not observed for other methods. All AI techniques extensively use the cutting force as input signal as it is present in the top 3 of each AI technique (except CNN). Vibration and AE signal are also commonly used. It is worth noting that, except for CNN, SVM is the second method that uses features extracted from images as input.

ANN and ANFIS/FIS techniques do not appear to require elaborate pre-processing as most of the articles do not specify the pre-processing techniques employed ("None" in Table 4.2). SVM, SOM, HMM and CNN all require special attention to the feature extraction techniques, especially SVM and HMM, which benefit from more advanced pre-processing techniques in the time-frequency domain. For the majority of methods, the computation of simple statistic characteristic on the signal is commonly used. By its nature, CNN does not require the same kind of pre-processing technique as it mainly uses images as input.

Two outputs are predominant in the use of AI techniques: estimation of VB and classification. In the analysis of AI technique by year (Section 4.3.1), it was shown that ANN, fuzzy and SVM techniques are predominant in the literature. It is observed that these three techniques are the only ones with "Estimation of VB" as the most represented output. Indeed, approaches such as SOM, HMM and CNN are mainly focused on classification. It is therefore observed that techniques monitoring the value of VB are favoured over a simple classification purpose.

Table 4.2: Comparison of the three most used: input, pre-processing and output for each approaches.

	ANN	**ANFIS & FIS**	**SVM**	**SOM**	**HMM**	**CNN**
Nb of articles	44	18 & 8	18	7	3	3
Input	Cutting Speed (24) Feed rate (24) Cutting force (22)	Cutting Force (17) AE (9) Feed rate (9)	Cutting Force (7) Image (5) AE (4)	Vibration (6) Cutting force (5) Noise (2)	Cutting Force (2) Vibration (2) /	Image(3) / /
Pre-processing	None (16) Statistic (12) Frequency domain (7)	None (12) Statistic (6) Other (3)	Time-frequency (8) Statistic (8) Other (5)	Statistic (6) Frequency domain (5) Time-frequency (2)	Time-frequency (2) Frequency (2) Normalization (1)	Image processing (3) / /
Output	Estimation of VB (34) Classification (7) Other (6)	Estimation of VB (23) Classification (5) /	Estimation of VB (10) Classification (9) Prediction of VB (1)	Classification (7) / /	Classification (3) Estimation of VB (1) /	Classification (2) Estimation of VB (1) /

As it is discussed in Section 4.5, comparing AI methods on the results they are able to achieve is not possible. Since each approach is carried out on a different dataset, with different indicators, with different inputs, etc. comparing these results would not provide a clear and unbiased analysis of the literature. Despite this fact, it is described in the previous sections that articles that compare the AI techniques with classical methods such as statistical model, show that the AI techniques outperform the classical models.

4.5 Discussion

As indicated by the various authors of the identified publications, all approaches presented in this SLR were able to obtain good results in monitoring the cutting tool health, some authors were even able to obtain more than 90% accuracy. The majority of these approaches have the common point to monitor the value of VB to assess the tool state. Unfortunately, there are few other approaches that try to evaluate the tool health differently. Some papers try to monitor the tool wear with other indicators such as its percentage of elapsed tool life, and others try to predict the RUL. This lack of other approaches may come from the difficulty to define a clear end-of-life criterion to cutting tools. In the ISO 3685 standard [27], the maximum value for VB is set to 0.3 mm as the end-of-life criterion, but, in real application, this value may not represent the cutting tool end-of-life, depending on the industrial applications and its tolerances. As this end-of-life can be very variable, having approaches based on the RUL or other time indicator can be very difficult to adopt for industrial applications. It is therefore understandable why researchers are more interested in the value of VB rather than trying to define a precise end-of-life criterion. It is up to the AI user to define the criteria corresponding to its application. By proposing an approach that monitors VB, the researchers ensure that their method can be applicable in real applications.

Knowing the current condition of a tool is obviously of great interest in condition monitoring. However, as the industry evolves, predictive monitoring tends to be the preferred approach. The current state of the art shows that there is too little research done to meet the future needs of industries in predictive maintenance, i.e., to predict the coming evolution of the tool condition. In recent years, advanced statistical models tend to answer this problem but still require a lot of tuning to obtain results. The use of AI could perhaps meet these needs of industries while guaranteeing models robust enough to be implemented on shop-floor applications. At this stage, it is not possible to assess whether this lack of interest in predictive maintenance is due to a lack of interest from the researchers or if it comes from a publication bias. Indeed, in this SLR are only listed articles that present relatively good results, and this may either be due to the fact that all techniques were always able to perform well, or that the publications which do not get good results are not published or not submitted for publication. It is therefore possible that, currently, research in predictive maintenance is not published due to a possible lack of results.

Due to the different nature of the AI techniques, it is not possible to make a comparison between each of them. Many articles compare their approach with other methods. In this SLR, these articles are listed and their results presented, although they are often contradictory. Indeed, an approach presented in any of

the selected articles is often presented to be the most effective compared with others. Given these contradictions, it is impossible to identify the best approach, if one exists. This may come from two reasons: (i) the publication bias previously discussed (the papers are not submitted or published if their approach is not better than another one), (ii) there is no best approach. Moreover, out of 102 articles, 98 performed a test campaign to obtain their dataset. As these data have been collected with different cutting conditions, materials, tools, ... it is not relevant to compare the results of these approaches. In turning, there is no benchmark dataset allowing for fairly comparing all the approaches, whereas benchmarks exist for other categories of machining. In milling, for example, some datasets are available on the NASA website [116], on the PHM website [117], ... This lack of unified benchmarking in turning does not allow a clear view of the performance and is a clear limitation of the comparison of performances in all these articles, which can therefore only be analysed objectively on the bases of their methodology instead of their results. The results obtained are therefore highly dependent on the quality of the data recovered during the tests and some data may be more fitted to certain approaches than others. Consequently, the pooling of various test data would allow the establishment of benchmarks. These benchmarks should aim to be the most representative of industrial reality.

It is also observed that the title of the articles could be confusing. Indeed, many articles incorrectly used the word "prediction" to talk about "monitoring". As a matter of fact, out of the 102 identified articles, 34 wrongly use the word "prediction" in their title and 62 in their title or abstract. The word prediction, in the English language, normally evokes the future state as defined in the Oxford dictionary: "*Prediction is a statement that says what you think will happen*" [30]. As shown in this SLR, the majority of articles actually use the AI technique as a monitoring system (i.e., to produce an estimate in real time). There is no forecasting of the future state. This abuse of language likely leads to confusion given the previous remark on the emergence of predictive maintenance.

4.6 Answers to the Research Questions

This section presents an answer to the research questions identified in Section 4.2.1. The answers to these questions are distributed through this SLR, but the main results are summarized here.

RQ1 How is AI used in the framework of monitoring/predicting the condition of tools in stable turning condition?

This question is answered throughout this SLR and RQ2 to RQ5. Several approaches have been mentioned and there is not one that stands out from the others (mainly due to the lack of unified benchmark). Each approach can find its place depending on the means and equipment made available to the monitoring system. However, as discussed in Section 4.5, it appears that, in the current state of the art, too few papers concern the evolution of VB, as they rather monitor the current state of the tool, which could be a problem in the framework of predictive maintenance in industry 4.0.

RQ2 What are the most used AI techniques used in this context?

In Section 4.4 of this SLR, all the AI methods were presented. Among them, there does not appear to be a more efficient approach than another. Each approach has its advantages and disadvantages. However, it appears in Section 4.3.2 that approaches from neural networks, fuzzy logic and support vector machines are particularly present in the literature. They all have the ability to be integrated into workshops. Neural network approaches have the advantage of having been widely studied but have a "black box" aspect which could cause reluctance in practice. Fuzzy logic seems to no longer be considered even though they had the advantage of presenting the results in an easily understandable way. In addition, finally, SVM approaches seem to perform as good as ANN and fuzzy approaches but require special attention to data pre-processing and the choice of parameters.

RQ3 What is the forecasting horizon of each of the identified AI techniques in this context?

Figure 4.9 presents the outputs of all the AI techniques presented in this chapter. It appears that the majority of articles use AI to monitor the current state of tool wear (estimation of VB). A classification purpose is also realised to detect the actual state of the tool. Some papers also present other kinds of output such as the percentage of tool life. Only five articles propose an approach that consist of a prediction of the future state. This is achieved by computing the remaining useful life (RUL) or a prediction of VB at a future time (generally called "T+1"). As discussed in Section 4.5, there is a lack of application in predictive maintenance and the actual state of the art is limited to the actual monitoring of the tool.

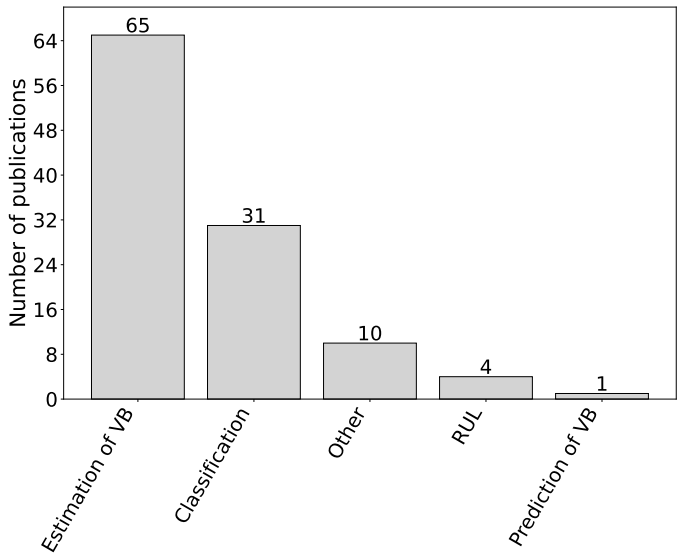


Figure 4.9: Output variables of all the approaches.

RQ4 What are the most common inputs used with the different identified AI techniques in this context?

As AI techniques are data-driven, the most common features have a high correlation with the cutting tool state. The repartition of all input data for all techniques is presented in Figure 4.10. The cutting force is the main input as it appears in 54 of the 102 articles. Features that correspond to the cutting conditions such as: feed rate, cutting speed, and depth of cut are also used a lot. These features are generally constant through the cutting process but are important to give context to the correlated data such as the cutting force, vibration signal, ... These cutting parameters thus have a significant impact in the ability of the AI technique to perform the monitoring of the tool.

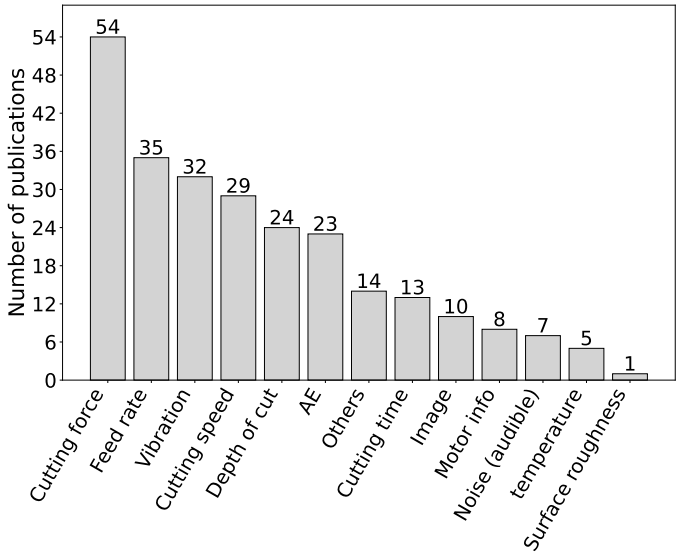


Figure 4.10: Input variables of all the approaches.

RQ5 What are the most used features extraction techniques in this context?

The features extraction technique for each AI approach is presented in Section 4.4. Figure 4.11 presents the repartition of all pre-processing techniques in this SLR. It is important to note that, depending on the approach, the pre-processing technique can be different. For example, it is observed that using ANN does not need much pre-processing while, for SVM, the pre-processing technique is necessary.

4.7 Conclusions of the Systematic Literature Review

This chapter presents a systematic literature review covering the use of AI techniques in the monitoring of cutting tool health in stable turning condition. The review

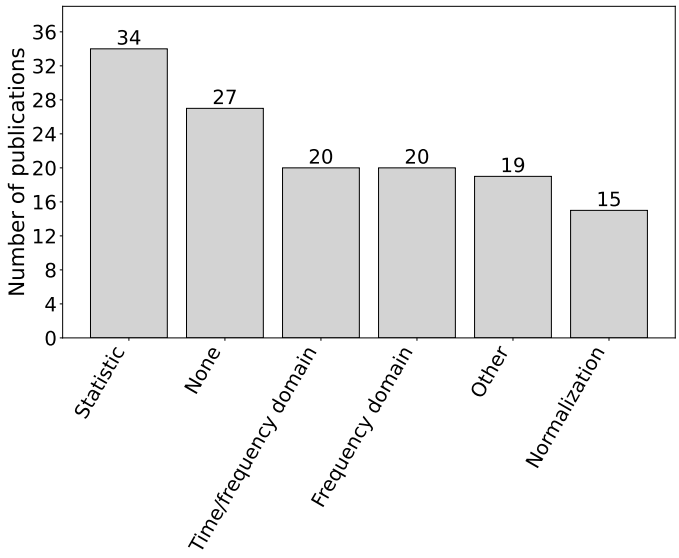


Figure 4.11: Feature extraction techniques of all the approaches.

protocol allowed for identifying 8426 articles in which 102 articles are corresponding to criteria of interest (Section 4.2). The review protocol followed the procedure recommended by Kitchenham [72]. During the definition of the research procedure, a new approach that aims to improve the quality of it was proposed. Some conclusions can be made about them:

- In the research procedure for this SLR, the choice to use a semantic approach to build the keywords was successful. It was chosen to construct each search query with a combination of four keywords so that each correspond to one part of the semantic approach (Section 4.2.2.2). The results show that this approach allows for browsing all of the literature on the subject of interest. However, it needs some additional sorting steps such as removing the "duplicated" items. Due to the number of articles identified, this last step cannot be performed manually. This approach is particularly interesting in the context of a systematic literature review.

- In the research procedure, to evaluate the inter-rater reliability, the Cohen's kappa [76] was used to evaluate the agreement among raters (Section 4.2.3). This gives an unbiased judgement of the quality of the sorting which ensures that the research was carried out in an unbiased manner.

The bibliometric analysis of the 102 articles was performed in Section 4.3 and shows the trends in the literature over the last 20 years. Several aspects were discussed including the distribution of the number of articles per year (Section 4.3.1). Some general conclusions can be drawn:

- There is a growing interest in the use of AI techniques to monitor the tool wear in turning operation. This is observed by the increasing number of publications by year in this domain (Section 4.3.1).

- Approaches using neural networks are predominant in the subject of this SLR with 44 articles (Section 4.3.2). This may be linked to the ease of implementation of this type of technique and to its robustness to handle noisy data. Fuzzy logic and SVM approaches are also often used by the researchers with 26 and 18 articles, respectively.

An in-depth analysis of the six AI approaches identified in this SLR was carried out and presented in the form of Table C.1–Table C.8. A discussion for each approach was performed in Section 4.4. Some general comments can be provided:

- Some articles compare the AI techniques and the classical statistical approach. It appears that the AI techniques outperform the classical approach. The main advantage of the AI technique over the classical approaches is that the AI are able to easily adapt to multiple cutting conditions without any further researcher expertise. This makes AI an interesting approach in the industrial context

- Almost all the articles presented in this systematic literature review perform a dedicated test campaign to obtain their own data. This makes it difficult to determine the best approach. An establishment of a benchmark dataset would allow for comparing the different approaches on the same data allowing the researchers to have a clear view of the best approach.

- There is a vast variety of approaches to monitor the tool wear, but the accuracy is usually above 90%. It indicates that, no matter the approach, AI techniques can be used to monitor the tool health. Due to the wide variety of approaches and the differences in the AI techniques employed, producing an effective horizontal comparison between the methods is not feasible.

- The forecasting horizon is actually limited to the real-time monitoring with only a few applications corresponding to predictive maintenance.

To conclude this SLR, some future scopes that need to be addressed in the use of AI to monitor the tool wear are highlighted:

- The implementation of a common benchmark dataset is needed to allow an objective comparison of the performance of proposed approaches.

- Since all the approaches mainly focus on real-time monitoring of the tool state, the predictive maintenance aspects need to be considered a priority.

- Since all ANN-based approaches are considered “black-box” models, this can lead to reluctance in their adoption. Therefore, techniques such as confidence interval estimation or explainability should be considered.

4.8 Literature Evolution since the SLR

As a literature review always covers a specific time period, it is useful to reflect on how the literature has evolved since the publication of the SLR.

An initial consideration is whether the topic of Tool Condition Monitoring (TCM) continues to hold significant relevance within the current research landscape. In the original SLR, a growing interest in the subject was observed through the increasing number of publications. During the search process of the SLR, the following search query was found to yield the highest number of results on *Scopus* [74]: "machining" AND ("prediction" OR "monitoring") AND ("artificial intelligence" OR "machine learning") AND ("condition" OR "data" OR "wear"). This query was repeated in September 2025, and the updated number of articles is shown in Figure 4.12. Although not all retrieved articles are directly related to the scope of the SLR, it is evident that interest in AI-driven TCM is rapidly growing, with nearly twice as many publications in 2024 compared to 2020.

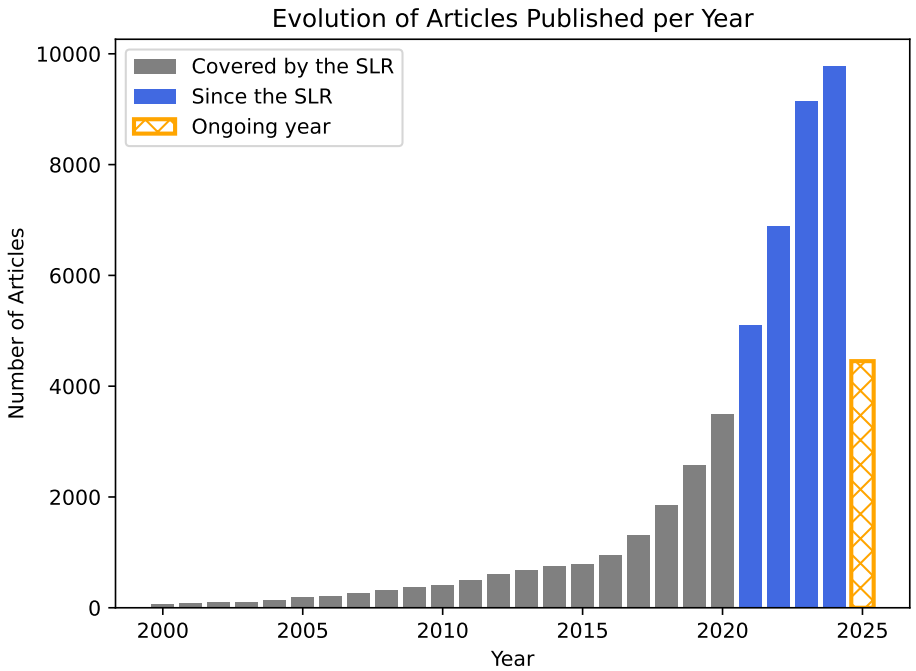


Figure 4.12: Evolution of articles about TCM published by year on the *Scopus* database. Data retrieved in September 2025.

Given the significant number of new publications, it is important to question whether the findings highlighted in the original SLR are still valid. Recent literature reviews on the topic of TCM have reached conclusions that align with those of the original SLR [118]. Artificial Neural Networks remain the most widely used approach among researchers, followed by applications of Support Vector Machines [118]. Moreover, both the data preprocessing techniques and the primary objectives

of the models have remained largely unchanged since the SLR. The models are still predominantly used to estimate the tool's condition, typically through the VB (flank wear) indicator [118].

As noted in the SLR, deep learning methods are also emerging, particularly to handle increasingly complex data generated by machining processes, whether from sensors or cameras [119]. The concepts of Industry 4.0 and 5.0 are central to current developments, with concrete methods being proposed to deploy models via edge and fog computing [120]. Ongoing research suggests that, as Industry 5.0 emphasizes human-machine interaction, there is a growing focus on explainable AI and more intuitive AI systems [120]. These limitations in interpretability were also discussed in the SLR.

In parallel to the methods developed specifically for TCM, it is also valuable to consider recent advances in artificial intelligence. Since 2020, a new class of models known as transformers has emerged and is gradually replacing traditional approaches in many domains [44, 121]. The use of transformer-based methods in TCM remains scarce, with only a handful of studies on *Scopus* addressing this topic. Given the rapid progress in this area, it is important to recognize that these approaches could become dominant in the coming years.

4.9 Integration of the Literature Findings into the Thesis

The findings and trends highlighted by the SLR played a key role in shaping the direction of this thesis and the topics explored throughout. The first step of the thesis was to verify whether ANNs truly represented the most promising approach. Although this investigation is not detailed in the present document, it led to the publication of two research articles, one focused on classification [122] and the other on regression [123]. Below, each chapter of the document is briefly introduced, along with an explanation of how the SLR impacted its development.

- Chapters 7 and 15 present an indirect monitoring regression approach based on neural networks, as identified in the SLR. The SLR highlighted the "black-box" nature of this approach, which can lead to reluctance in practical adoption. To address this limitation, these chapters introduce two novel methods: confidence interval estimation for tool condition monitoring and the integration of explainable models. Both issues are discussed in detail and serve as key motivations for the proposed methods.

- Chapters 8 and 16 explore a topic that is rarely addressed in the literature: indirect forecasting regression. While most studies focus on estimating the current state of the tool, these chapters aim to develop models capable of predicting not only the current condition but also the future evolution of tool degradation.

- Chapters 9 and 17 focus on direct monitoring classification using image analysis to classify tool wear states. Given the limited literature on this topic, the use of transfer learning is explored. Additionally, the literature highlights that one of the main challenges in applying these methods lies in image alterations during industrial acquisition. To take that into account,

data augmentation techniques aiming to represent these images alterations are used to evaluate the robustness of the developed models. In addition to that, vision transformers are also tested to evaluate their performance for tool condition monitoring.

- Chapters 10 and 18 extend the use of image analysis to direct monitoring regression. These chapters evaluate model robustness against image alteration, similar to the classification approach.

Part II

Development of Tool Condition Monitoring Methods on Turning Operation

Chapter 5

Introduction to the Development of Tool Condition Monitoring Methods in Turning

In the previous chapters, the relevance of tool condition monitoring (TCM) was introduced. In an idealized scenario, it was shown that TCM methods can help reduce production costs and CO_2 emissions (see Chapter 3). The literature review conducted on the topic highlighted various TCM strategies, but more importantly, it revealed several gaps in the existing research (see Chapter 4). This part of the document therefore focuses on the development of different TCM techniques to fill these gaps. The application case for these developments is the turning operation, chosen for both practical and physical reasons. From a practical standpoint, a turning dataset was already available within the lab at the beginning of the thesis. This dataset, originating from a previous doctoral work (see [17]), provided a valuable starting point for developing various TCM methods.

Moreover, turning is an ideal case study because the tool remains continuously engaged with the material, resulting in simpler cutting dynamics compared to milling. This leads to cleaner signals and easier data processing, which is advantageous for developing and testing data processing techniques on more manageable datasets

This Part of the document is therefore organized as follows:

- Chapter 6 describes the database that will be used throughout this part of the thesis.
- Chapter 7 explains how the data is used for indirect monitoring regression. This chapter directly addresses the limitations identified in the literature review, particularly regarding confidence intervals and model explainability in AI.

- Chapter 8 focuses on indirect forecasting regression. It introduces a method based on recurrent neural networks to predict the future evolution of the tool. The turning dataset is used to illustrate the relevance of this approach. As with the monitoring chapter, this work stems from limitations observed in the literature.
- Chapter 9 presents how tool images are used to classify their condition. To best represent and evaluate the robustness of models against various image alterations that may occur in industrial environments, the images are modified and the model is assessed on its ability to classify the tool condition based on the altered image. This chapter also applies transfer learning techniques to train the models. Both aspects: image alteration and transfer learning, address limitations identified in the literature.
- Chapter 10 also uses images similarly to Chapter 9, but this time to directly estimate the wear area on the tool image. The conclusions of this chapter are limited due to constraints in the dataset, which will be addressed in the milling section of the thesis.
- Chapter 11 presents the general conclusion of the methods developed throughout the thesis.

Chapter 6

Description of the Turning Database

6.1 Introduction

For the whole Part II of this document, the dataset used is from Lucas Equeter's PhD thesis [17]. This dataset serves as the foundation for the development of the artificial intelligence methods discussed in the subsequent chapter of this Part of this thesis.

This chapter therefore introduces the database developed for this study. Its purpose is to provide a structured dataset that will be used throughout this Part of the document to link the signals recorded during the experimental turning tests to the condition of the cutting tool. It presents the complete data preparation process for use with AI methods (see Figure 6.1 and Chapter 2). Starting from the raw signals collected during the experiments, a set of indicators is computed. These indicators, derived either from the time domain or the frequency domain, serve as representations of the signal characteristics. However, not all indicators are necessarily related to tool wear. Therefore, a correlation analysis is performed to identify those most representative of the tool's condition. Finally, the chapter describes the split of the training, validation, and test datasets that will be used for subsequent AI modelling. This descriptive chapter is organized as follows (see Figure 6.1):

- Section 6.2 describes the experimental setup.
- Section 6.3 explains how time-domain and frequency-domain indicators are extracted from the dataset.
- Section 6.4 examines the correlation between the extracted indicators and tool wear. This analysis helps identify the most relevant indicators to be used as input features for the models developed in this part of the thesis.

- Section 6.5 outlines how the data is split for training and testing the AI methods. The test dataset is selected to include the most diverse degradation trajectories possible.

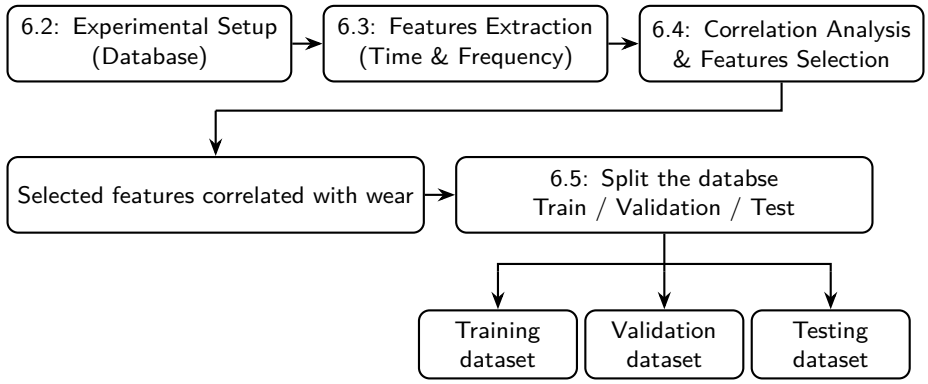


Figure 6.1: Flowchart illustrating the approach developed in this chapter, structured according to the data preparation logic outlined in Chapter 2.

6.2 Experimental Setup

The dataset used in this part originates from experimental straight turning tests. The objective of these tests was to create a database that correlates cutting tool wear with cutting forces. The material used during the experiments consists of C45 steel bars (also known as EN 1.0503). Figure 6.2 shows the C45 bars before and after machining. As shown by the Figure, the machining operation consist of 10 pass that will create 9 steps on the workpiece. Table 6.1 shows the chemical composition of this alloy. Every bar received a heat treatment at 850°C during 8 hours (annealing) to achieve a constant hardness of 179 HV30 through their section. The bars were machined on a Weiler E35 CNC lathe (Figure 6.3). The cutting insert (CNMG120404-M3 TP40) and tool holder (DCLN L 2020K12-M) were selected based on SECO TOOLS recommendations [124]. The insert was selected to have the shortest possible lifespan, thereby minimizing material waste during testing.

The machining process for each tool in the experimental campaign consists of several steps. First, a bar is mounted in the machine and machined for an average duration of 2 minutes and 45 seconds. During the final pass of the tool through the material, cutting forces are recorded using a Kistler 9257B force sensor. This final pass lasts slightly less than 15 s, and the signal is sampled at 10 kHz.

Once the bar is machined, the tool holder is disassembled, and tool wear is assessed according to the ISO 3685 standard [126], using a Bysameyee EU-1000X 3 digital portable microscope. The bar is also dismounted and the arithmetic roughness (R_a) of the machined surface is measured with a Diavite DH-6 roughometer. These operations are repeated until tool end-of-life is observed. On average, tools

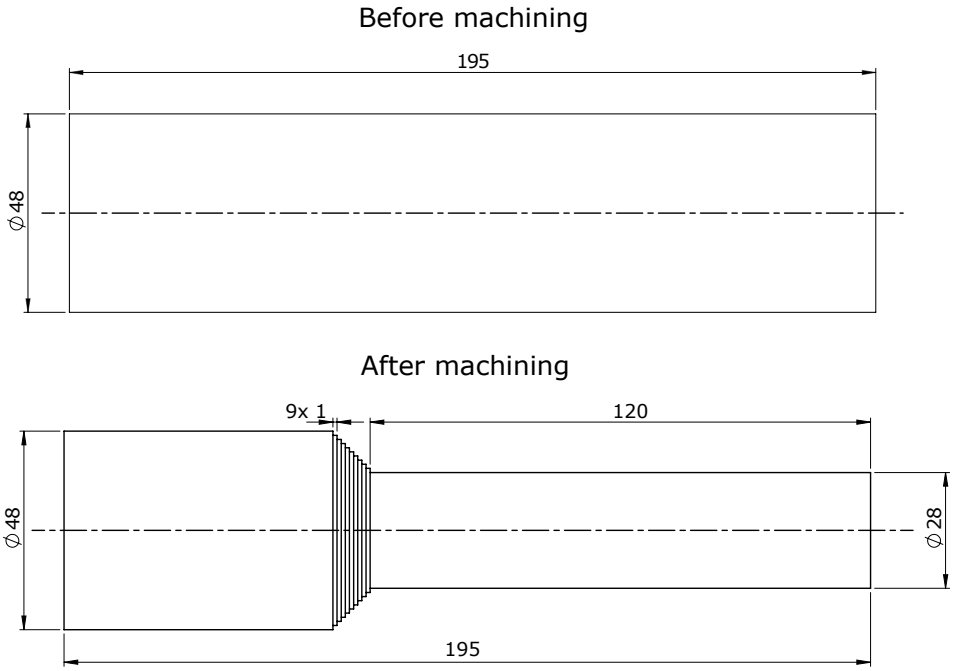


Figure 6.2: Technical drawing of the C45 steel bar before and after machining during the turning test.

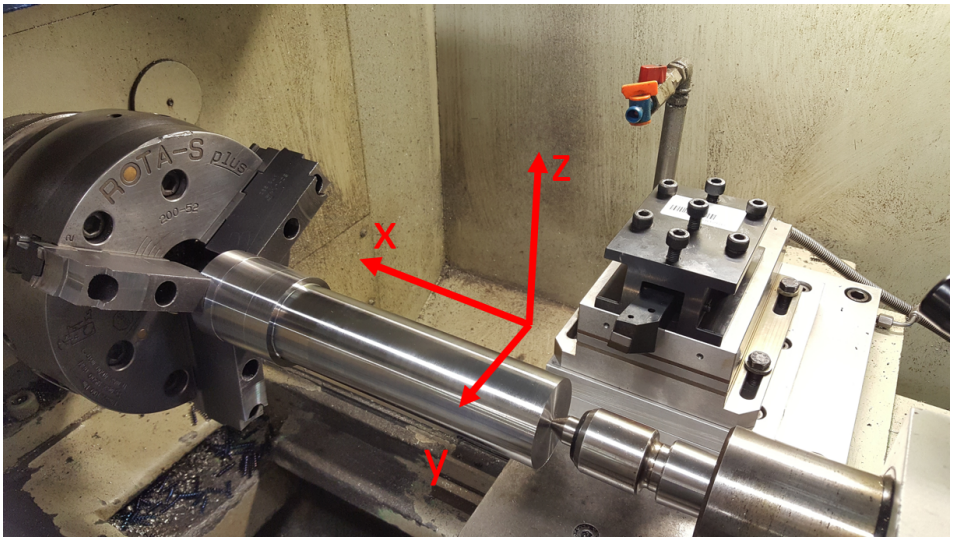


Figure 6.3: Experimental straight turning setup. The reference frame of the machine is displayed in red.

Table 6.1: Chemical Composition of C45 Steel (EN 1.0503) [125].

Element	**Content (%)**
Carbon (C)	0.42 – 0.50
Silicon (Si)	≤ 0.40
Manganese (Mn)	0.50 – 0.80
Phosphorus (P)	≤ 0.045
Sulfur (S)	≤ 0.045
Chromium (Cr)	≤ 0.40
Nickel (Ni)	≤ 0.40
Molybdenum (Mo)	≤ 0.10
Cr + Mo + Ni	≤ 0.63

reach their end-of-life of 300 µm after machining 6.4 bars which corresponds to around 20 min of machining.

The complete dataset consists of 30 tools operating at various cutting speeds for a total of 192 measurements. This number of tools is higher than what is generally reported in the literature. Indeed, about ten tools are generally used to test the methods as in [127–129]. In the presented database, more tools are considered showing dispersion in degradation even under identical cutting conditions. Table 6.2 summarizes the cutting speed variation for each tool. The feed and depth of cut are constant and fixed to $0.2\,\mathrm{mm\,tooth^{-1}}$ and 1 mm respectively. Trajectories 21 to 30 saw variable cutting speed. It means that during the life of the tool, the cutting speed was changed at the beginning of machining a new bar.

Table 6.2: Cutting conditions for each turning test. All tests were performed on C45 steel, with a constant feed of $0.2\,\mathrm{mm\,tooth^{-1}}$ and a depth of cut of 1 mm.

Tool N°	**Cutting Speed v_c [m min^{-1}]**
1 to 10	260
11 to 15	250
16	240
17 to 20	265
21 to 30	Variable: 240 to 260

Table 6.3 provides a summary of the measurements collected during the tests. In the reference frame used (Figure 6.3), the cutting force components are defined as follows: F_x corresponds to the feed force, F_y to the radial force, and F_z to the main cutting force.

Images acquired with the microscope (Figure 6.4) are processed to measure the degradation of the tool. The ISO 3685 standard specifies that VB should be measured using the average of the flank wear [126]. However, the standard does not specify a minimum number of measurements for estimating this average. This problem is usually not discussed in the literature but can impact the value of VB. The selected measurement method consists of identifying the wear area using image processing software and then calculating the value of VB along the entire length

Table 6.3: Measured parameters and their sensor during the turning tests.

Parameters measured	Sensor	Associated variables
Cutting force and Torque	Kistler 9257B	F_x, F_y, F_z, M_x, M_y, M_z
Picture of the tool	Bysameyee EU-1000X 3	VB and notch wear
Roughness	Diavite DH-6	R_a

of the area. This allows the estimation of the mean value of VB as accurately as possible. This method provides an estimate of VB that is better than the average of a few points. It also averages out any errors that may have been made when selecting pixels at the border of the wear area in the image.

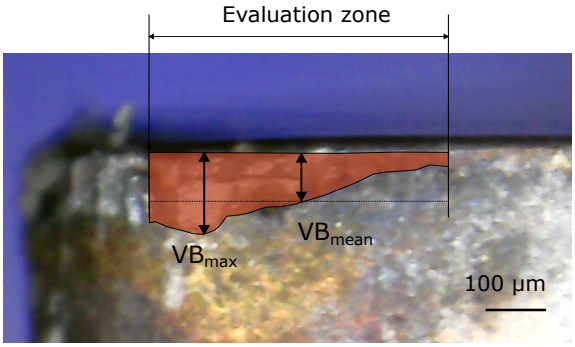
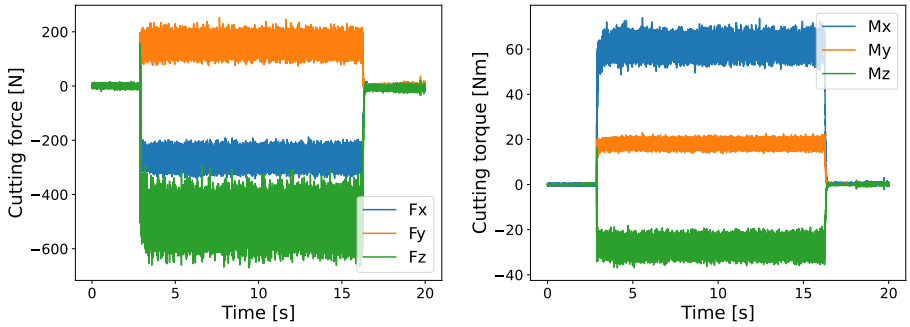


Figure 6.4: Example of VB measurement according to the ISO 3685 standard [27] for tool number 14 bar number 5. VB_{max}: 135 µm and VB_{mean}: 78 µm.

6.3 Features Extraction

The signal measured during the last pass for each bar is a temporal signal similar to the one presented in Figure 6.5. These temporal signal are not directly usable with some neural networks, so it is necessary to extract relevant features from them. There are several techniques for feature extraction, the most common being statistical and frequency analysis of the signal. In the following, 5 statistical and 2 frequency features are computed for each cutting forces and torques (F_x, F_y, F_z, M_x, M_y and M_z). The different statistical feature extraction techniques have been chosen based on the literature on the topic [33]: Root Mean Squared (RMS), mean, variance, kurtosis and skewness. Figure 6.6a shows the evolution of the RMS value of the feeding force (F_x) for different tools. If the temporal signals from the sensors are represented as: t_1,...,t_N, and $\bar{t}$ the mean value of this temporal signal. The statistical features can be computed as presented in Table 6.4 with the central moment (m_i) defined in Eq. (6.1).

$$m_i = \frac{1}{N}\sum_{j=1}^{N}(t_j - \bar{t})^i \tag{6.1}$$



(a) Cutting force **(b)** Cutting torque

Figure 6.5: Example of raw temporal signal acquired during the turning for the tool number 16 and bar 5.

Table 6.4: The 5 statistical features processing methods and their mathematical equations.

Feature processing method	**Equation**
Mean	m_1
Standard deviation (std)	$\sqrt{m_2}$
Skewness (skew)	$\frac{m_3}{m^{3/2}}$
Kurtosis (kurt)	$\frac{m_4}{\sigma^4}$
Root Mean Square (RMS)	$\sqrt{\frac{1}{N}\sum_{i=1}^{t} t_i^2}$

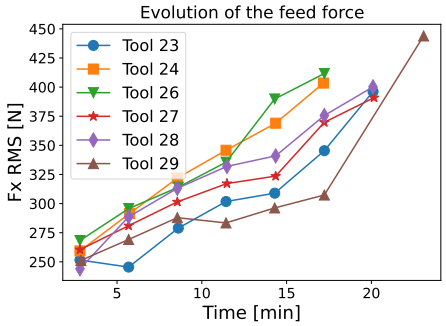
In the frequency domain, the Welch's method [130] is used to extract the Power Spectral Density (PSD) on each signal. Figure 6.6b shows the PSD under 3 kHz computed with the Welch's method on the feed force (F_x). The amplitude (denoted as PSD) and the associated frequency of the highest peak (denoted as PSD freq.) are used as indicators.

Other features can also be computed such as the total length machined by the tool that can be computed as (Eq. (6.2)):

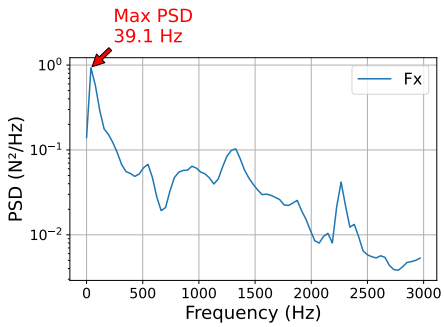
$$\text{total length} = \int v_c(t)dt \tag{6.2}$$

6.4 Correlation Analysis and Features Selection

The previously described feature extraction method enables the computation of numerous indicators across the entire dataset. However, not all extracted features exhibit the same level of correlation with cutting tool wear. To identify the most relevant candidates for use as inputs to AI models, a correlation analysis is performed [127]. A correlation analysis is a statistical method used to evaluate the



(a) Example of statistical indicator: RMS of the feed force for several tools



(b) Frequency indicator: Power Spectral Density on the feed force estimated with the Welch's method

Figure 6.6: Example of features extracted with the statistical and frequency analysis.

relationship between two variables. A correlation coefficient equal to +1 implies perfect correlation between the two values while a value of 0 indicates no correlation. A correlation −1 indicates a perfect negative correlation, meaning the variables move in opposite directions. Multiple types of correlation analysis exist. In this application the Spearman's correlation analysis is used as it is the most adapted to the size and non-normality of the dataset [131, 132]. Figure 6.7 shows the correlation of the data with respect to VB. As the database is exclusively composed of cutting force signals, it does not allow a comparison of the correlation of this signal with other signals such as vibrations which are regularly used in this type of study [33, 133]. This limitation will be overcome in milling (see Chapter 13). Nevertheless, a strong correlation between cutting forces and tool degradation state is observed, which is also observed in the literature [134, 135].

Table 6.5 shows the 5 selected features and their correlation with tool wear. These features were chosen because they show the strongest correlation with wear while avoiding redundancy. For example, F_x RMS and F_x Mean have the same correlation magnitude (0.87 and −0.87, with opposite signs due to the reference frame used), therefore, only F_x RMS was retained. The quantity of 5 inputs is chosen after tests where the addition of more inputs did not improve the results presented in the next chapters. Therefore, in order to keep the network as small as possible, the smallest number of inputs is chosen. Of these 5 signals, 3 are related to the cutting forces (M_z RMS, F_x RMS and F_z RMS) and 2 are independent of the tool condition (machining duration and total machined length) but provide information about its past life. The machining duration gives the age of the tool. Total machined length indicates the amount of material removed, and when considered alongside machining duration, it also reflects the mean cutting speed.

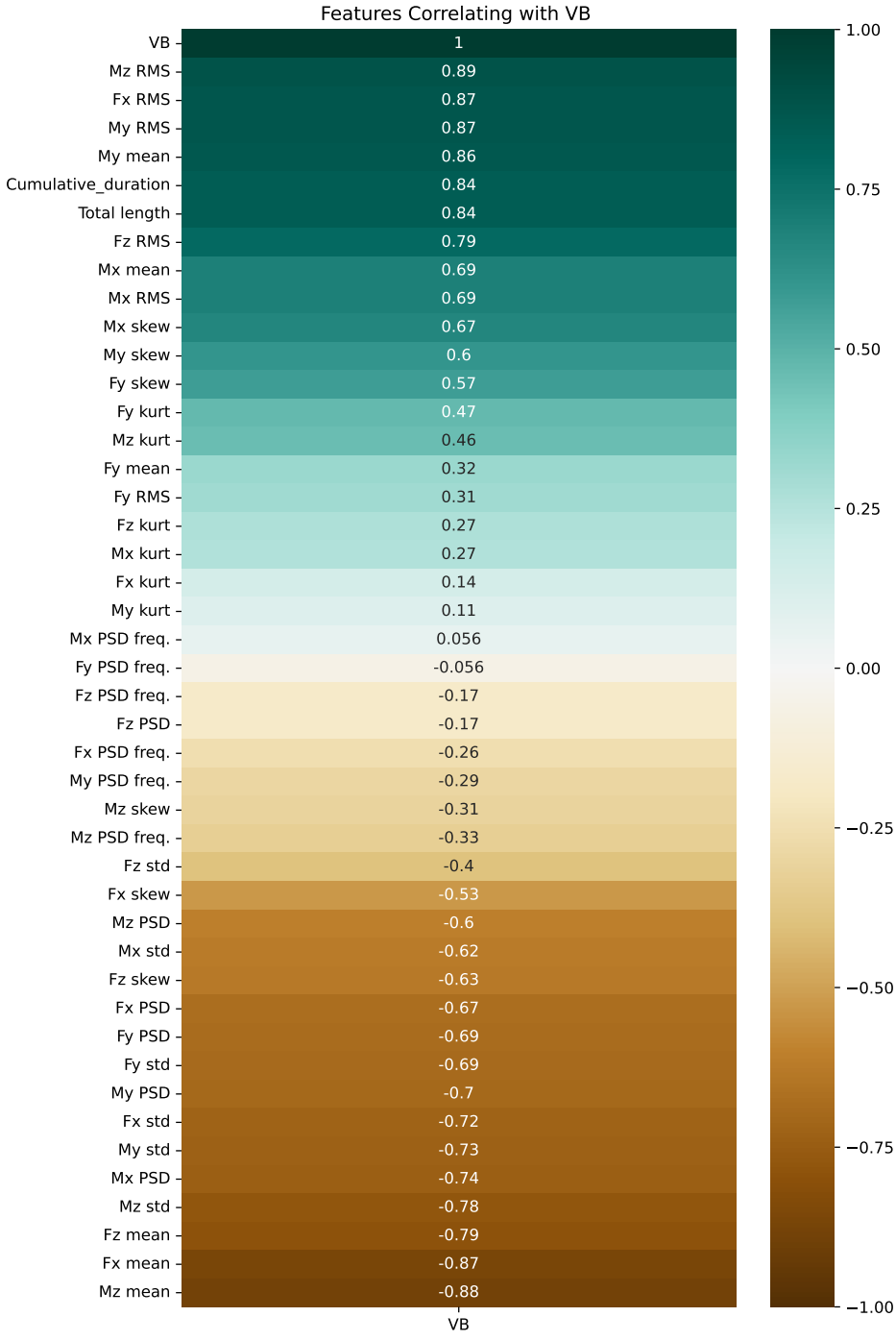


Figure 6.7: Correlation analysis on the whole dataset with respect to VB.

Table 6.5: Most correlated features with respect to the tool wear.

Features	Spearman's correlation coefficient with respect to VB
M_z RMS	0.89
F_x RMS	0.87
Machining duration	0.84
Total machined length	0.84
F_z RMS	0.79

6.5 Training and Testing Dataset

The goal of any AI model is to learn from a training dataset in order to generalize its knowledge to previously unseen data. As discussed in Chapter 2, the dataset must be divided into distinct subsets to monitor training performance and mitigate the risk of overfitting [53]. These subsets include a training set, a validation set, and a test set. The testing dataset is selected to ensure that the models are able to generalise their results. Therefore, the test set consist of the tool number: 23, 24, 26, 27, 28 and 29. These tools were selected as the testing dataset from those who saw variable cutting speed between two measurements (tool 21 to 30). Figure 6.8 shows the degradation of the selected tool. Each tool has a different degradation trajectory, some, such as tool 23, have an almost linear degradation, whereas tool 26 has a brutal degradation after 11 minutes of machining. These particularities in their degradation trajectories make them an excellent testing set. This testing set represents 20 % of the entire dataset. The remaining 80 % is further split into 75 % for training and 25 % for validation. This division ratio falls within the standard range for the field of artificial intelligence. Additionally, all data are normalized to facilitate faster convergence and improve model performance.

6.6 Conclusions

The turning database presented in this chapter serves as the foundation for training the various AI methods for the remainder of this thesis Part. This database consists of the evolution of the cutting force with respect to the tool wear and is therefore an ideal starting point to develop the different methods to fill the gaps identified in the literature.

In this database, the identified features most correlated with the wear will be used as entry of the model to evaluate their performance to monitor or forecast the cutting tool condition. These features are: M_z RMS, F_x RMS, F_z RMS, the machining duration and the total machined length.

To ensure that the future results will be evaluated as rigorously as possible, the training, validation and testing dataset have also been defined to ensure a testing set representative of the whole database.

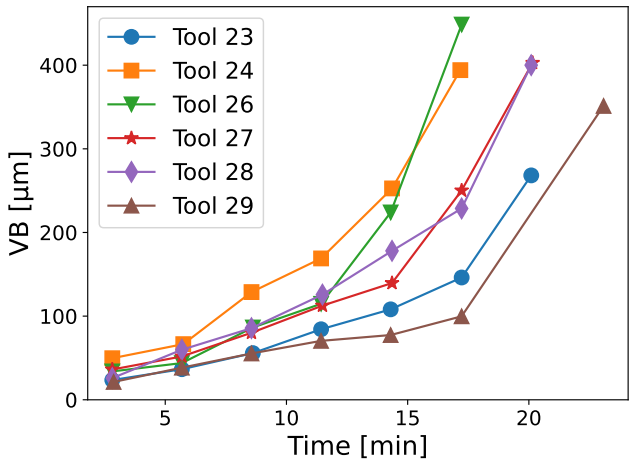


Figure 6.8: Degradation trajectory for tools: 23, 24, 26, 27, 28 and 29.

Chapter 7

Monitoring the Cutting Tool Degradation from Sensors Signals in Turning

Indirect Monitoring with Regression

The following results are mainly based on four articles published as part of this thesis. The selection of ANN is from [122, 123], the confidence interval estimation is from [69] and the interpretability is from [136]

7.1 Introduction

The turning dataset (Chapter 6) enables the evaluation of various AI methods on a comprehensive database that includes different cutting conditions, but with only a single sensor. In this chapter, this dataset is used to focus on "indirect" "monitoring" approach using "regression". The objective is therefore to use the sensor measurements collected during machining to estimate the current state of the tool (represented as VB).

In Chapter 4, three gaps in the literature were identified regarding indirect monitoring regression. The first gap concerned determining whether neural networks are the most suitable approach for monitoring the cutting tool state. Although this aspect is not directly detailed in the present chapter, it has been addressed in publications produced during the thesis, which demonstrated that neural networks are the most appropriate method for both classification [122] and regression [123]. Consequently, neural networks will be primarily used in this chapter and the results with this approach will be presented. The second gap relates to the absence of confidence interval estimation around the results, while the third and final gap concerns the lack of interpretability, as models generally operate as black boxes.

To present the results and respond to each of these gaps, this chapter is structured as follows (Figure 7.1):

- Section 7.2 describes the neural network architecture best suited to the dataset. As mentioned earlier, the decision to use a neural network is not discussed in detail in this thesis. However, the architecture and its results are presented here to address the first gap identified in the literature. This architecture will serve as the basis for all neural network–based results presented in the remainder of the Chapter.
- Section 7.3 presents the results obtained on the testing dataset (defined in Chapter 6) using the architecture defined in Section 7.2.
- Section 7.4 complements the results by introducing a bootstrap method to estimate a confidence interval around the tool monitoring predictions. This section is a direct answer to the second identified gap and was published in [69].
- Section 7.5 introduces an approach based on Explainable Boosting Machines (EBMs), offering a transparent alternative to the black-box nature of neural networks. This section is a direct answer to the third identified gap and was published in [136].
- Section 7.6 concludes the chapter and recall all the contributions.

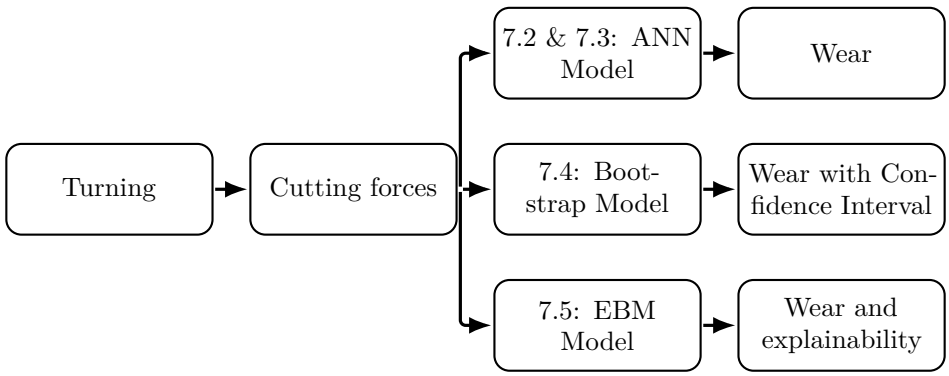


Figure 7.1: Summary of the approaches developed in this chapter.

7.2 Determination of the Best Network Architecture

The training of neural networks depends mainly on the choice of hyperparameters and the architecture of the network. For database similar to the one used in this chapter, the literature on the subject generally uses neural networks with a shallow architecture: 1 to 2 hidden layers with a few neurons inside [33] (Chapter 4). Therefore, similar combinations of architectures, containing from 1 to 4 hidden layers each with between 5 and 15 neurons, were tested. To ensure that more complex models would not be more appropriate for the database, models with up to 40 hidden layers were also tested. As these deeper model turned out to not provide any gains of performances, they will not be presented in the following. The different

architectures are tested on the same test dataset (defined in Chapter 6 Section 6.5) and their performance are compared. Different combinations of activation functions are tested: Sigmoid, tangent hyperbolic (Tanh), Rectified Linear Unit (ReLU) (Defined in Chapter 2).

Table 7.1 compares the relative performance of different architectures on the test dataset. Every tested architecture is trained with a maximum of 1000 training epochs, a starting learning rate of $1e-4$ and a bach size of 8. The training of these network follows the logic describes in Chapter 2. The notation [5, 6, 6, 1] is read as there are 5 inputs, 2 hidden layers each containing 6 neurons and 1 output. The architectures presented in the table were selected because they are the most representative of each category tested. Several indicators are presented to compare the performance: the Mean Squared Error (MSE), the coefficient of determination (R^2 score) and the Mean Absolute Percentage Error (MAPE). These indicators are defined as follows:

- The MSE is a measure of how close a fitted line is to data points. For each point, it calculates the square difference between the observed and predicted values and then averages these values. The lower the MSE, the better the prediction performance. The MSE is defined as (Eq. (7.1)):

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{7.1}$$

  where:

  - N is the total number of observations,
  - y_i is the actual value of the i^{th} observation,
  - $\hat{y}_i$ is the predicted value of the i^{th} observation.

- The R^2 score is a statistical measure that represents how well the predicted value fit to the real one. A value of 1 indicates a perfect fit, while a value of 0 shows no fit. The R^2 score is defined as (Eq. (7.2)):

$$\text{R}^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} = 1 - \frac{SS_{res}}{SS_{tot}} \tag{7.2}$$

  where:

  - $\bar{y}$ is the mean of the actual values,
  - SS_{res} is the sum of squares of the residual errors,
  - SS_{tot} is the total sum of squares.

- The MAPE is a statistical measure used to determine the accuracy of a prediction. It calculates the average of the absolute percentage differences between the actual and predicted values. Lower MAPE values indicate better accuracy. A MAPE value of 0 indicates perfect predictions. It is expressed as a percentage, and is defined by the formula (Eq. (7.3)):

$$\text{MAPE} = \frac{100\%}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{7.3}$$

Table 7.1: Comparison of the relative performance of several neural network architectures after a maximum of 1,000 training epochs, sorted by performance. The best architecture is highlighted in bold.

Architecture	MSE	R^2 Score	MAPE	Activation function
[5, 6, 6, 1]	**850**	**94%**	**15%**	**Tanh and ReLU**
[5, 6, 10, 6, 1]	1200	91 %	19 %	Tanh, ReLU and ReLU
[5, 6, 6, 1]	1250	90 %	25 %	Sigmoid and ReLU
[5, 5, 13, 9, 1]	1330	90 %	23 %	All ReLU
[5, 11, 1]	1550	89 %	22 %	All ReLU
[5, 6, 1]	1600	88 %	27 %	All ReLU
[5, 12, 5, 1]	1630	88 %	25 %	All ReLU
[5, 6, 6, 1]	1700	87 %	26 %	All ReLU
[5, 6, 6, 1]	12000	14 %	30 %	All Tanh
[5, 6, 6, 1]	12000	14 %	30 %	All Sigmoid

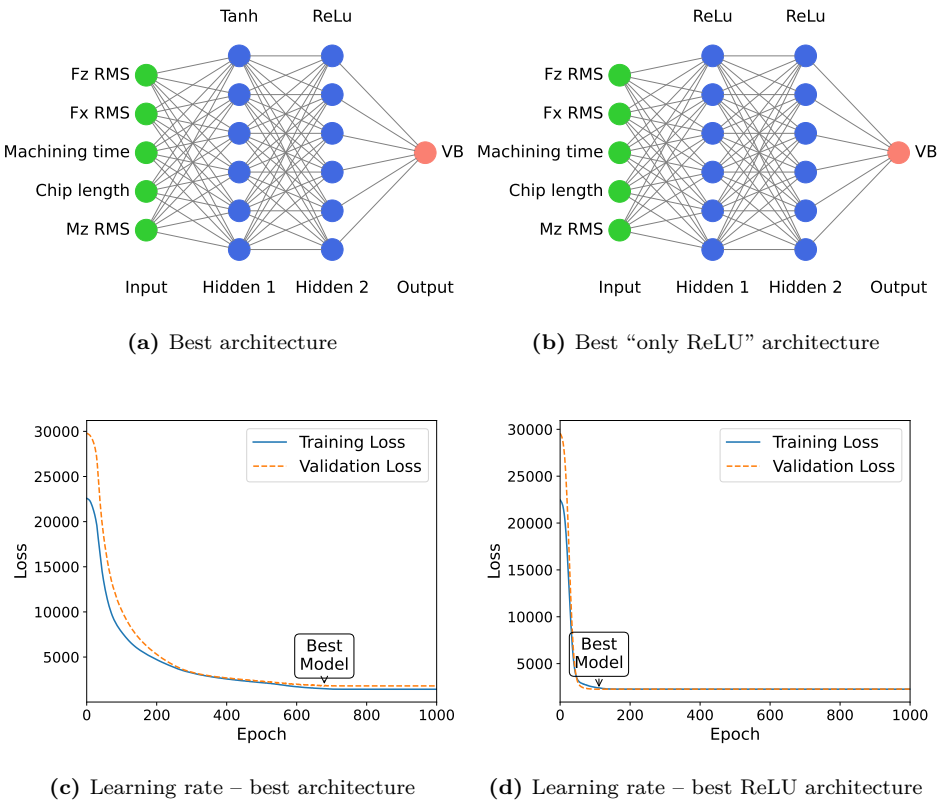
These scores show how accurate the model is on average (MAPE), and how well it captures the variance on the testing trajectories (R^2 score). Several observations can be made about Table 7.1:

- All architectures containing only "ReLU" layers converge to similar performance, as shown by their comparable performance indicators. The mean R^2 score stands around 88 %, while the mean MAPE is around 23 %. The architecture seems to have minimal impact on the results obtained. Consequently, it is recommended to use less complex network architectures, as more complex ones do not offer significant advantages for databases such as the one used in this chapter.

- The architectures with only Tanh or Sigmoid activation function fail to obtain good results in only 1000 epochs. This is largely attributed to the data scale. Specifically, Tanh and Sigmoid functions can only output values ranging from -1 to 1 and 0 to 1, respectively. Given that the VB values are expressed in microns (ranging from 0 to 450 µm), a saturation effect is observed. It is important to note that even when the VB values are scaled to better suit these architectures, the results do not show any significant improvement over those discussed in the remainder of this section.

- The optimal architecture is composed of two hidden layers, each with six neurons. The first layer utilizes a Tanh activation function, while the second uses a ReLU activation function. Compared with approaches using only ReLU, this architecture improves the R^2 score by an average of 5 % and MAPE by 8 %.

It should be noted that the network with a Tanh layer (Figure 7.2a) takes more epochs to train than a network with only ReLU layers (Figure 7.2b). This is observed in Figures 7.2c and 7.2d where the best model takes 4 times longer to converge but achieves better results on the testing set. These figures also show that there is no overfitting as the training and validation losses have a similar evolution.



(a) Best architecture

(b) Best "only ReLU" architecture

(c) Learning rate – best architecture

(d) Learning rate – best ReLU architecture

Figure 7.2: Learning comparison between the best identified model and an uniform network – Note that the "EarlyStopping" is disabled to allow to have 1000 epochs and not be stopped after convergence.

7.3 Results with the ANN Approach

Figure 7.3 illustrates the performance of a neural network, using the previously defined architecture, across the testing trajectories (23, 24, 26, 27, 28 and 29). In blue is represented the real cutting degradation that was measured during the turning test. In red, the prediction made by the model and in orange the relative error between the prediction and the degradation. This error is computed as the difference of the prediction (red line) and the real degradation (blue line). The pink area on the figure shows the expected end-of-life of the tool associated with wear above 300 µm. The figures show that the monitoring system is able to track tool wear with an accuracy within a few tens of microns for nearly all tools. Only tool 26 (Figure 7.3c) exhibits a significant overestimation of wear around the 15-minute mark, where the predicted wear exceeds 300 µm, even though the tool is still in usable condition. One important point for this monitoring is when the red line attains the end-of-life region. Because this is the moment that should trigger a tool

replacement. For all other trajectories, the error in estimating the tool's end-of-life remains below one minute.

7.4 Results with the Confidence Interval Approach

7.4.1 Motivation of the Confidence Interval Estimation

Although, in theory, neural networks have an excellent ability to approximate non linear behaviour, they only provide a point estimate of the tool state. Thus, the prediction is less reliable if the data are sparse or affected by probabilistic events. To interpret the results correctly, the uncertainty of the network on the prediction must be taken into account [137]. In real-world applications, relying solely on a single value estimate can leave practitioners uncertain about the prediction's accuracy [138]. Therefore, providing a confidence interval or a probability distribution is crucial for assessing the reliability of the output [139]. This additional information enables practitioners to estimate the likelihood of a tool failing to meet quality standards in upcoming machining operations, and to make informed decisions about whether immediate tool replacement is necessary.

7.4.2 Bootstrap Approach

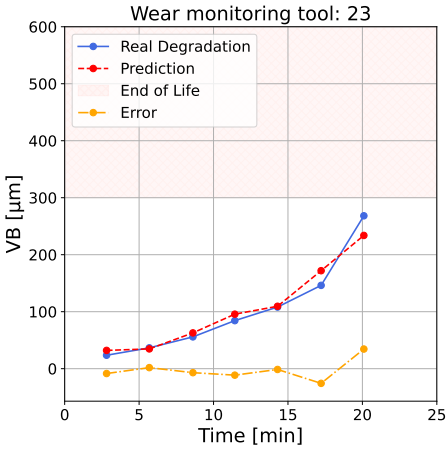
There exist multiple approaches to estimate a confidence interval around the prediction. In this chapter, a bootstrap method is chosen as it was never applied to cutting tool health monitoring prior. Nonetheless, this method has already been used in the literature for different applications [140, 141]. The bootstrap approach involves training a set of neural networks in parallel and querying all the networks to estimate a confidence interval (Figure 7.4). The collective decision of the networks is less likely to be in error than any individual neural network. As each network does not converge in the same way due to the random initialization of the network and the training process, it will be shown that the predictions follow a normal distribution. Thus, if N is the total number of networks and $\widehat{y_{pi}}$ the i^{th} estimation of the tool wear, the average regressed value $\bar{y_p}$ is (Eq. (7.4)):

$$\bar{y_p} = \frac{1}{N} \sum_{i=1}^{N} \widehat{y_{pi}} \tag{7.4}$$

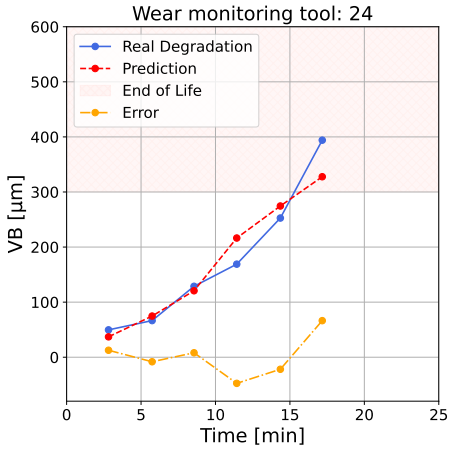
The normality of the results imply that knowing the mean and standard deviation (σ), it is possible to create a 95 % confidence interval (CI) around the prediction (Eq. (7.5)):

$$\text{CI} = \bar{y_p} \pm 1.96\sigma \tag{7.5}$$

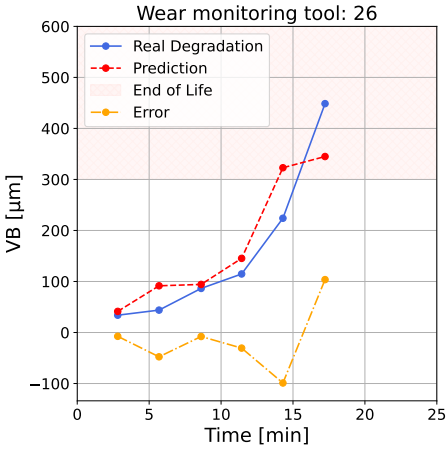
The normality assumption can visually be proved by a quantile-quantile plot (Q-Q plot) [142]. Figure 7.5 shows the Q-Q plot as well as the histogram of the wear estimate for a cutting tool degradation with the network architecture previously presented. In order to have a significant number of points and thus a sufficient statistical distribution, this figure is made with 100 estimates from 100 networks trained with the bootstrap method. On the quantile-quantile graph, the red line represents the theoretical Normal distribution, the blue points represent



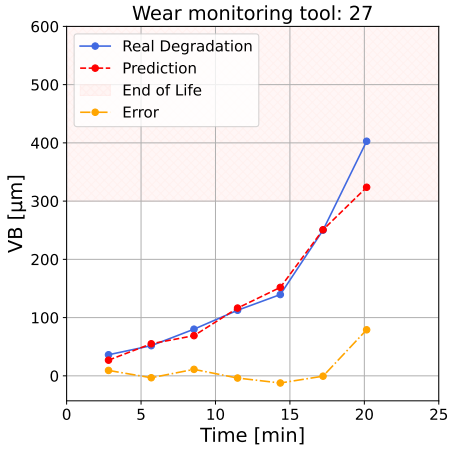
(a) ANN – Result for tool: 23



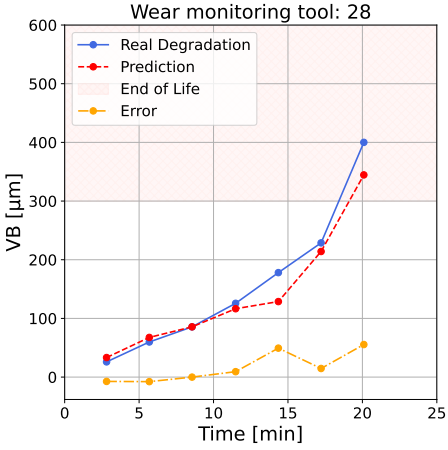
(b) ANN – Result for tool: 24



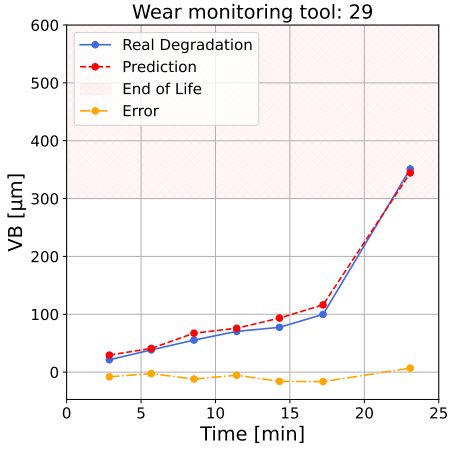
(c) ANN – Result for tool: 26



(d) ANN – Result for tool: 27



(e) ANN – Result for tool: 28



(f) ANN – Result for tool: 29

Figure 7.3: Monitoring: Estimated tool wear on the testing dataset.

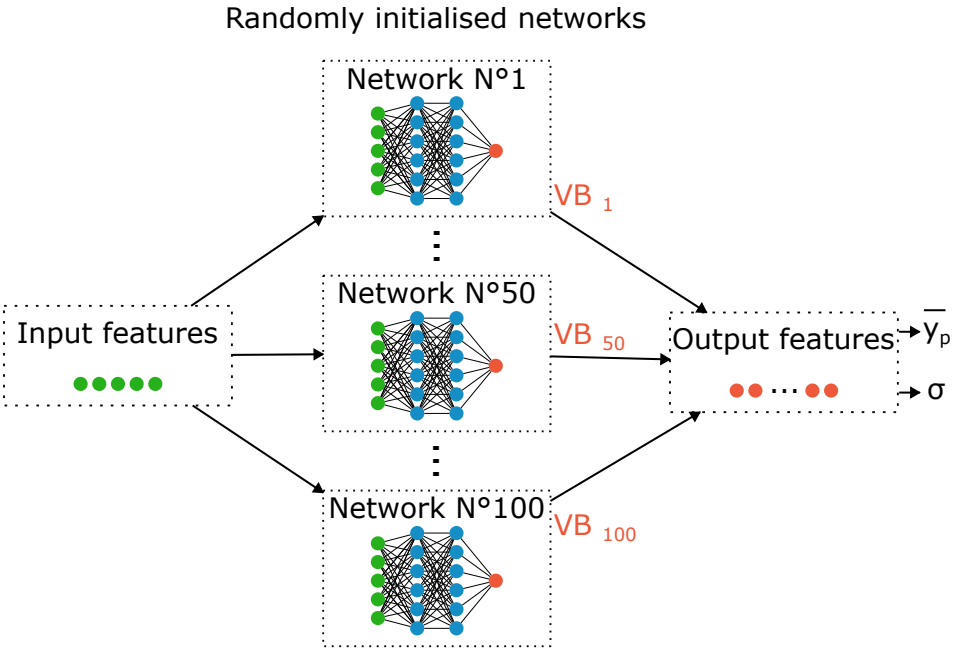


Figure 7.4: Bootstrap approach with 100 randomly initialised identical neural networks.

the different neural network estimates. If the distribution was perfectly Normal, all points would lie on the red line. In Figure 7.5, outliers are present at both ends, but it is observed that the majority of the points approximate the theoretical Normal distribution. This confirms that the hypothesis of a Normal distribution is respected. The second plot in Figure 7.5 is a histogram that allows to visually represent the distribution, a bell-shaped curve is visually observed, confirming the Normal distribution.

In the following approach, this bootstrap method will be used to identify the confidence interval on the tool wear estimate. 100 neural networks will be trained in parallel to create the confidence interval estimate. This number was identified by trial and error, the number of 100 is a compromise between the computational time for training and the behaviour of the confidence interval. Beyond 100 networks, the confidence interval remains the same.

Compared to a classical neural network approach, this bootstrap approach requires more computing resources, especially during training. As mentioned above, the networks are relatively small, so they individually do not require high computing power. Training a neural network on a Intel I7-9750h (6 cores, 12 threads at 2.6 GHz to 4.5 GHz) takes less than a minute on a single core, training 100 networks in parallel takes around 10 minutes of computing time. Once the network is trained, integrating the 100 networks takes less than a second, which is not critical for real-time monitoring even with low computing resources.

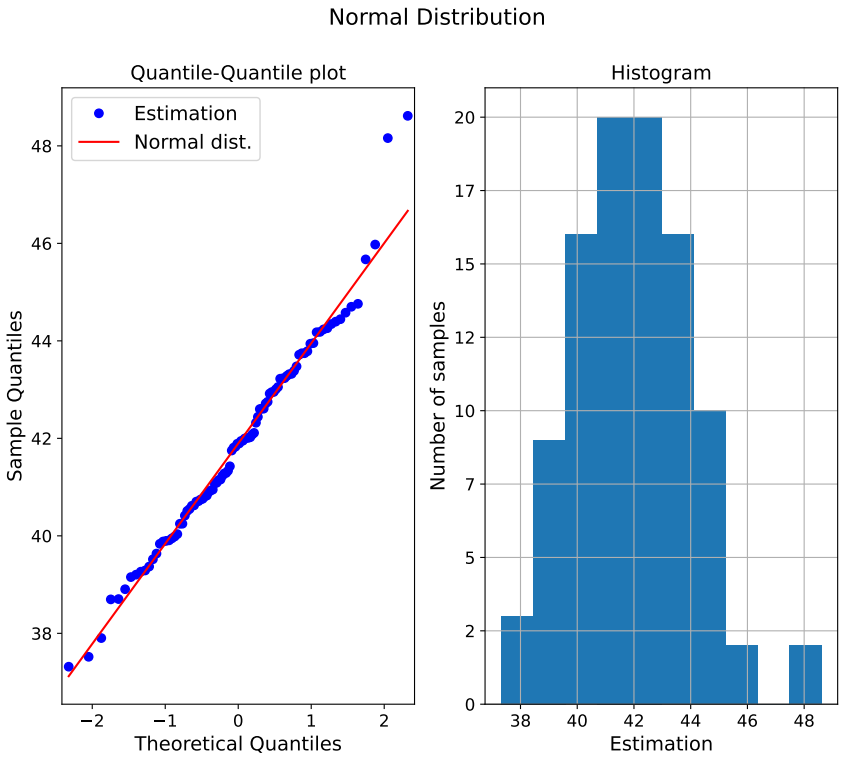


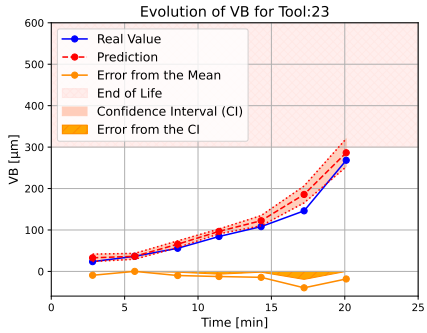
Figure 7.5: Quantile-quantile plot and histogram of the estimation of the neural networks.

7.4.3 Results with the Bootstrap approach

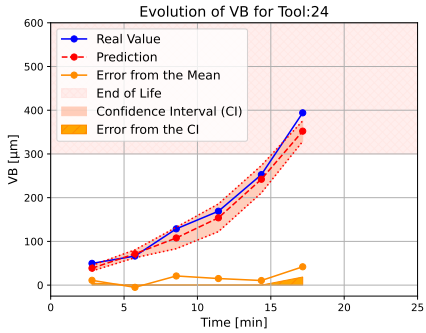
Figures 7.7a to 7.7f show the results of the tool wear estimation with 100 neural networks with the architecture presented previously. The red line corresponds to the mean estimate by the neural network while the red area corresponds to the 95 % confidence interval around the estimate. The orange area is the error between the confidence and the measured value whereas the orange line is the error between the mean estimate and the measured value. The results for the different tools are:

- For tool 23 (Figure 7.7a), the estimate is consistently slightly higher than the actual observed wear, but the confidence interval matches almost consistently the real value. A more significant overestimation is observed around 17 minutes that is around 40 µm but the confidence interval is larger than before and the error on this interval is smaller. For this tool, the end of life has not been reached.

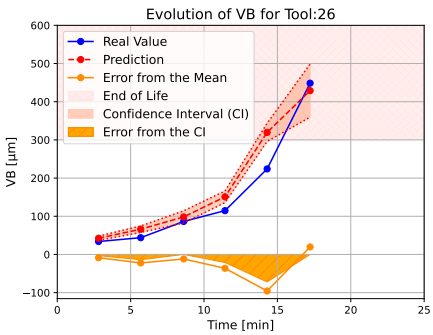
- Compared to the rest of the test database, tool 24 has a higher initial degradation rate. The network estimate has a slightly larger confidence interval over this part of the tool's life, which is a unique case since generally the confidence interval increases with time (Figure 7.7b). Apart from this sudden change in tool wear, the estimator is able to track the degradation



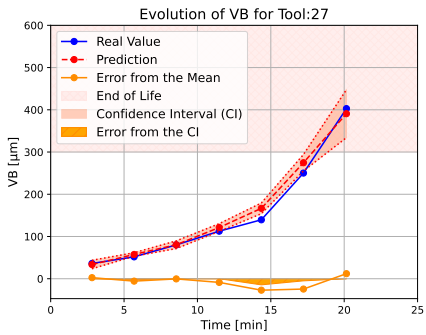
(a) Bootstrap – Result for tool: 23



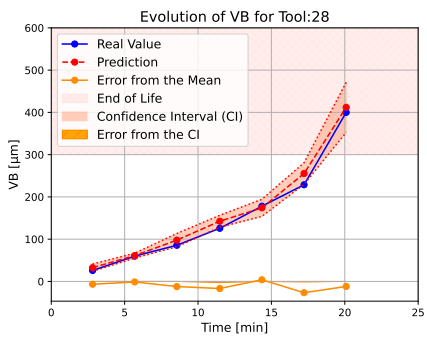
(b) Bootstrap – Result for tool: 24



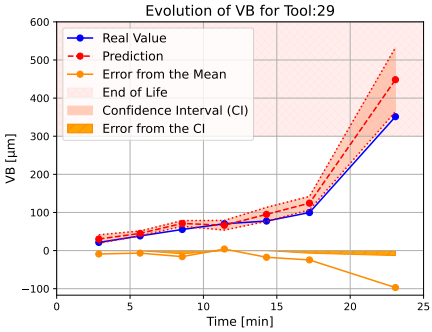
(c) Bootstrap – Result for tool: 26



(d) Bootstrap – Result for tool: 27



(e) Bootstrap – Result for tool: 28



(f) Bootstrap – Result for tool: 29

Figure 7.6: Estimated tool wear on the testing dataset with a confidence interval estimated with the Bootstrap approach.

correctly but tends to slightly underestimate the value of the degradation. This is a unique case in this test dataset. For this tool, the observed end of life is reached less than 1 minute before the predicted end of life.

- Tool 26 is the tool with the fastest end of life degradation (Figure 7.7c). At the beginning of the tool's life, the network is able to track the degradation correctly with a relatively low confidence interval, but then, due to the sudden change in the wear rate, the estimator strongly overestimates the tool's degradation. In this case, it even indicates that the tool has reached the end of its life because the prediction around 15 minutes is higher than 300 µm. Then the confidence interval increases strongly and the estimate is closer to reality. For this tool, the end of life is reached around 2 minutes after the predicted end of life. This tool also provides a good example of the value of confidence intervals, which becomes evident when comparing its performance with the model without confidence intervals. Without the confidence interval, the tool state is significantly underestimated (see Figure 7.3c). However, this represents only a single estimate of the tool condition that in this particular case is far from measured degradation. The addition of the confidence interval, captures the uncertainty range within which the true estimate is more likely to lie.

- The degradation of tools 27 and 28 are fairly similar (Figures 7.7d and 7.7e), there is simply a slightly earlier change in the degradation rate for tool 27 than for tool 28. For both tools, the neural network correctly tracks the degradation and the confidence interval increases slightly at the end of the tool's life. For these tools, the end of life is reached almost at the same time as the predicted ones.

- Tool 29 has a lower degradation rate than the others and a longer tool life (Figure 7.7f). The estimator is able to follow the degradation of the tool correctly, but its end of life is overestimated and the confidence interval is significantly larger than for the other tools. For this tool, the end of life is reached 2 minutes after the predicted end of life.

The results on particular degradation trajectories with either variable degradation rates or variable duration and with variation of the cutting speed show that the approach is able to follow the tool degradation correctly. On average, it is observed that monitored value of the wear cross the end-of-life zone (300 µm of wear) around the same minute as the measured one. In the worst cases, the error produced on the estimation of the tool is not very significant and corresponds to an error of a few microns. There are nevertheless 2 erroneous points, one for tool 26 and the second for tool 29. The origin of this error is not clearly identifiable. But we can note that following these errors the interval of confidence has increased showing the interest of this approach. This interval is particularly useful when the tool sees sudden variations in wear and a single estimate would not allow an assessment of the network's confidence. As the database does not contain significant variation in cutting conditions, the size of the confidence intervals is consistently small. This is interesting, and unexpected, because too large confidence intervals would not allow

to clearly estimate the state of wear of the tool. Too large confidence intervals in this chapter would have shown that the method would not be applicable in more complex cases with greater variations, but this is not the case here. It's important to note that these observations hold true for the case of 100 parallel networks, but they may be influenced by the number of networks used. In particular, reducing the number of networks tends to widen the confidence interval, while increasing it has the opposite effect. The standard deviation varies with the number of networks (n), following a proportional relationship of $\frac{1}{\sqrt{n-1}}$ [143]. Therefore, the choice of the number of networks represents a trade-off between available computational resources and the desired level of accuracy.

7.5 Results with Explainable Model

7.5.1 Motivation to Explainability

In the previous sections, neural networks were used to solve the monitoring problem. Although these methods have proved effective, their practical application remains limited. This is mainly due to the fact that most AI models still operate as "black boxes", offering little or no insight into how decisions are made or why certain predictions are produced. This lack of interpretability is a significant barrier to their adoption in real industrial environments, where understanding and confidence in system behaviour is crucial. Therefore, the manufacturing industry is not keen to trust models blindly as it can have a serious impact on their production and operating costs. For a long time, the trade-off between explainability and performance remained a persistent issue in most AI methods [144]. In recent years, significant research efforts have been made to develop so-called "glass-box" AI methods. These methods aim to be explainable while being as accurate as "black-box" approaches. An example of this is the Explainable Boosting Machine (EBM), which is a type of Generalized Additive Model (GAM) that uses cyclic gradient boosting to enhance interpretability and accuracy [45]. This model can provide a global and a local explanation of its results and can be used in several applications where interpretability is critical for decision-making. A global explanation describes how the data is used to perform the model's task, offering insights into general trends and data interactions. Local explanation clarifies why a model made a specific decision, making each individual prediction explainable. Explainability's method can be either specific to a model or applicable to various models [145].

In the context of cutting tool wear monitoring, some works on interpretability exists. In 2022, Guo et al. [146] used a deep learning method with attention mechanisms to provide explainability for model predictions based on cutting signals. They observed that due to the randomness of each training session, the interpretations varied. By averaging the results, they were still able to identify certain characteristic explanations [146]. However, this fluctuation in interpretability remains a limitation. In 2024, Kumar et al. [147] used deep learning models for image segmentation on tool images, incorporating a GRAD-CAM (see Chapter 9) analysis to explain the model's areas of interest. They combined this approach with human validation to guide the model in identifying the correct wear zones. They demonstrated that their approach was more effective than without interpretability

analysis and human-AI collaboration [147]. However, this method is only applicable to models that process images of cutting tools and requires a person capable of performing this operation. In general, in the field of cutting tool monitoring, either explainability is not considered, or the methods used are not sufficiently transparent to be used industrially. In addition, an analysis of a "glass-box" method such as EBM has not been applied to cutting tool monitoring yet.

Therefore, the objective and novelty of this approach is to utilize an EBM method and demonstrate its value in monitoring cutting tools degradation. To evaluate the model's performance, it is tested on the turning database. The model's hyperparameters are fine-tuned using a grid-search algorithm. Then, the monitoring performances are compared to the ANN approach previously developed. The model's explainability is then demonstrated, providing both global and local explanations of the results to showcase the capabilities of these models.

7.5.2 Comparison of ANN results with Explainable Model

Compared with ANN, EBMs do not require a pre-selection of features as they can dynamically determine the importance of their inputs. Therefore, all calculated indicators as well as the cutting conditions are directly provided as inputs to the EBM model. The data are not normalized because the model can handle non-normalized input data. This simplifies the explanations offered by the model, as it retains the physical meaning of the data.

The EBM model was developped using the InterpretML library [148]. The model includes several hyperparameters that must be carefully selected to achieve the best interpretability and monitoring performance. In this study, hyperparameter selection was performed using a grid search algorithm. Table 7.2 summarizes the selected values alongside the recommended defaults from the InterpretML library [148] (see Chapter 2, Section 2.7 for the full explanation of the hyperparameters). The first parameter, max leaves, controls the maximum number of leaves in the decision trees used to model each feature. A low value encourages simpler models and reduces the risk of overfitting. In this case, the value was set to 3, which is more conservative than the default and promotes interpretability. The learning rate determines how quickly the model updates its predictions during training. A lower rate ensures more stable convergence and reduces the likelihood of overfitting. The selected value of 0.01 is slightly below the default (typically 0.015 for classification), reflecting a cautious approach. The smoothing rounds parameter governs the smoothing of contribution curves for continuous variables. While the recommended values are around 75 for classification and up to 500 for regression, the optimal model was obtained with a value of 200, offering a good balance between curve readability and model flexibility. The interactions parameter defines how many pairwise feature interactions the model should explore. In this case, it was set to 0, meaning the model is purely additive. This decision was based on the absence of significant interactions in the data, even when the parameter was increased during testing. The EarlyStopping rounds parameter sets the patience for stopping training when no improvement is observed. A value of 200 was chosen, allowing the model sufficient time to converge while avoiding unnecessary computation. The inner bags parameter specifies the number of subsamples used to train each

feature model. A high value such as 20 improves robustness by reducing variance in the feature contributions, though it increases training time. Finally, the outer bags parameter determines the number of independent models used to estimate prediction uncertainty. The selected value of 14 aligns with recommended practices, providing reliable confidence intervals without excessive computational cost. All the other hyperparameters have been set to their default value.

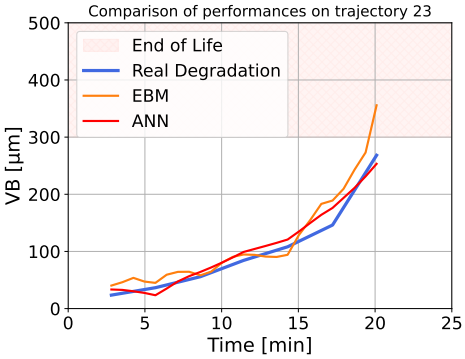
Table 7.2: Hyperparameter of the EBM model with the recommended value by the InterpretML library [148] and the value selected through a grid search algorithm.

Hyperparameter	**Recommended [148]**	**Selected**
Max leaves	2 to 3	3
Learning rate	0.0025 to 0.2	0.01
Smoothing round	75 to 500	200
Interaction	Data specific	0
EarlyStopping rounds	100 to 200	200
Inner bags	0 or 20	20
Outer bags	14	14

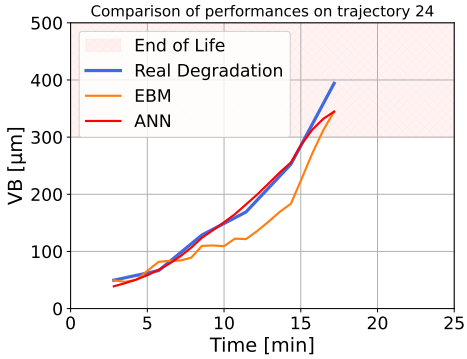
Figure 7.7 compares the performance of the EBM approach developed in this section with the ANN approach developed in the previous section (Section 7.3) across the different testing trajectories. Note that due to random initialisation of the ANN, the monitored value presented on this figure can vary from the ones presented earlier. Both models are trained on the same training dataset. This figure shows the target trajectory (measured during the tests) and the monitoring of the two methods. Table 7.3 summarizes the Mean Squared Error (MSE) and the coefficient of determination (R^2) metrics across the same testing trajectories between the EBM and ANN approaches. From the Figures and Table, it is observed that the performances are often similar, as illustrated by trajectories 27 and 28. However, trajectories 23 and 24 are less accurately tracked by the EBM model, especially during the mid-to-end life of the tool, where monitoring is crucial due to the higher likelihood of triggering a tool replacement. In contrast, trajectories 26 and 29 are both better tracked by the EBM approach, but it should be noted that on these trajectories, the ANN has the highest deviation compared to the other trajectories.

On average, according to the dataset and testing trajectories described in this chapter, the EBM approach is less effective than the ANN approach. However, despite being less effective, its performance can still be satisfactory, as the figures indicate that the general trend of degradation is accurately followed. Additionally, the table shows that the average metric values are similar, with an average R^2 difference of only 2 %.

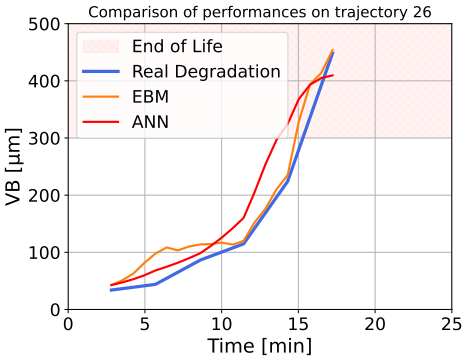
Although the EBM approach is less accurate than the ANN approach, the difference remains relatively small and is highly dependent on the trajectory. One advantage of the EBM approach is that it does not require feature selection to perform well. All database data is input into the model, which automatically ranks the most correlated features with tool wear. In contrast, ANN requires feature selection, making it more time-consuming. This is beneficial in applications where



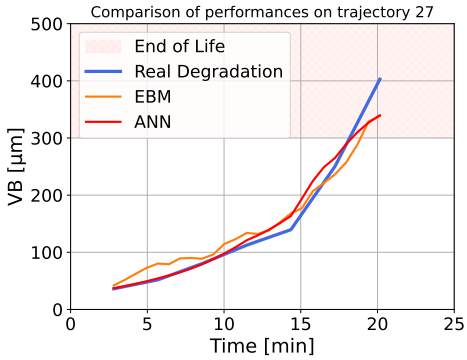
(a) EBM – Result for tool: 23



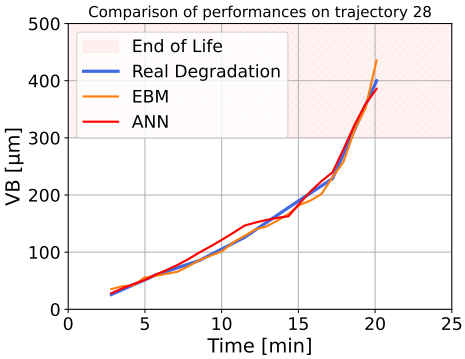
(b) EBM – Result for tool: 24



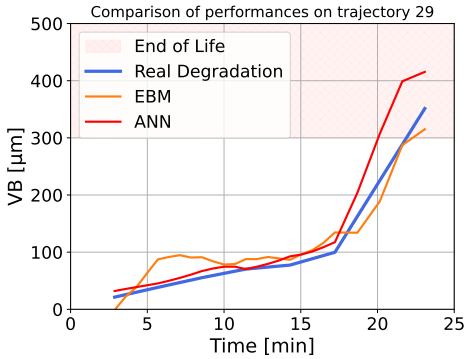
(c) EBM – Result for tool: 26



(d) EBM – Result for tool: 27



(e) EBM – Result for tool: 28



(f) EBM – Result for tool: 29

Figure 7.7: Comparison of performances of the EBM model and the ANN model trained and tested on the same dataset.

Table 7.3: Comparison of performance metrics from the trajectories presented in Figure 7.7.

	EBM		ANN	
	MSE	R^2	MSE	R^2
Traj. 23	768	0.82	190	0.95
Traj. 24	1591	0.84	194	0.98
Traj. 26	911	0.93	2412	0.83
Traj. 27	569	0.94	380	0.96
Traj. 28	96	0.99	102	0.99
Traj. 29	763	0.88	1082	0.83
Overall Results				
Average	783.00	0.90	726.67	0.92

relevant data is difficult to identify and the interaction between data points is not obvious, such as machining with multiple sensors.

7.5.3 Global Explanation

Figure 7.8 illustrates the average contribution of the input features to monitor tool's degradation. The Figure shows the first and last 5 features to illustrate the range of their average contribution to tool degradation. This hierarchy of importance is established by the EBM model during training, ensuring transparency about which features are most useful for predictions. The values shown on the y-axis of this graph represent the mean absolute contribution of each variable in estimating tool wear. The higher the contribution, the more important the parameter is relative to the others. This score is therefore a relative measure used to compare the variables with one another. Among the most significant values are: the torques around the Z axis (M_z) and the feed force (F_x RMS). This observation that the cutting force is highly correlated to the wear is a well-documented observation in the literature [134] and was also observed during the correlation analysis in Chapter 6. The EBM model automatically detected this correlation. Due to the database having only minor variations in cutting speed, this parameter contributes less to the estimation of the tool's condition and is among the least significant indicators. A Spearman's correlation analysis on this dataset also yields similar observations (Chapter 6, Section 6.4). The model can also identify interactions between features. However, since the database contains data from only one sensor, interactions between signals from the same sensor do not provide any benefit. This is confirmed by the results, which show that the model did not detect any interactions within the database.

Figure 7.9 shows the global importance of a single feature. Here, the global contribution of the feed force (F_x RMS) is selected to be shown. This feature is chosen due to its strong correlation with the tool's condition and its straightforward physical interpretation. The score (y-axis) is expressed in µm. This score represents the contribution of F_x RMS to the tool's wear estimation for different values (x-axis). The figure illustrates that depending on the value of the feed force, its contribution

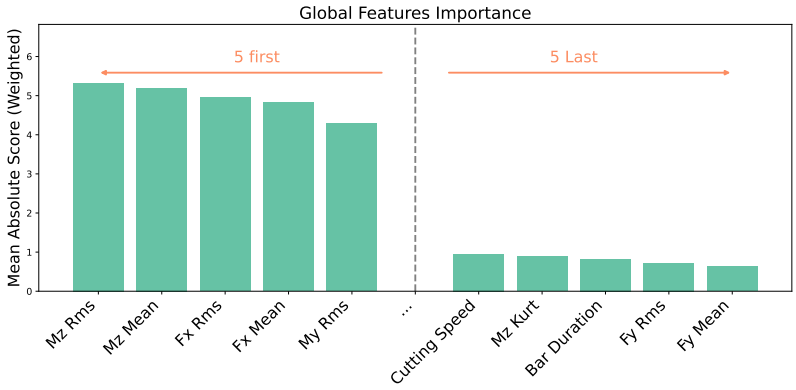


Figure 7.8: Global explanation showing the 5 top and 5 last average contribution of all the input of the model.

to the wear estimation changes. Higher cutting force values correspond to a greater impact on wear estimation, aligning with the physical phenomenon where a worn tool requires higher cutting force [134]. For example, when the F_x RMS value is 375 N, its contribution to the tool's condition estimation is approximately 15 µm. The model uses this value, along with all other feature values, to estimate the state of the tool. This figure therefore provides an overall summary of the contribution of F_x RMS for different input values. It should be noted that the trend observed in Figure 7.9 is generally seen in other features as well. Details about local explanations and wear estimation are described in Section 7.5.4. Around the value (referred to as the score), there is a confidence interval indicating the variance of this variable. The density graph below shows the data distribution for that particular features. The height of the bar indicates the density of this value in the database (DB). This visualization is important as a model can perform very differently with large and small samples [148]. It is therefore important to know the density of the data as larger samples tend to provide more statistically significant interpretations.

7.5.4 Local Explanation

In addition to global interpretation, local explanation is achievable with EBMs. Instead of looking at trends, the focus is on examining how a particular prediction is made. Figure 7.10 represents the local explanation of the last wear prediction of trajectory 29 (Figure 7.7). On this figure, the y-axis represents the feature's contribution expressed in µm and the x-axis corresponds to the feature name. In this example, the actual measured wear is 288.4 µm, and the model's prediction is 287.4 µm To obtain this wear estimation, the model first provides the "intercept" value (light green bar in Figure 7.10), which is a constant determined during the training phase. Each input feature then adds a positive or negative contribution to this value. A positive contribution indicates that the tool is more worn compared to the average value (intercept), while a negative contribution indicates less wear. The sum of all contributions represents the wear estimation of the model expressed in µm.

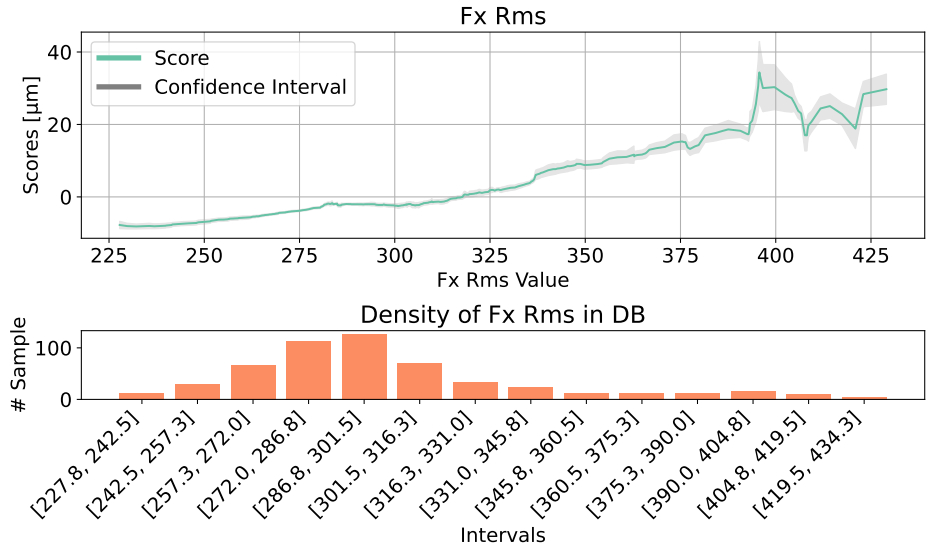


Figure 7.9: Global explanation of the selected features F_x RMS and its contribution to estimation of the tool wear with respect to the input value. The graph below shows density of each interval inside the database.

In this example, with a worn tool, features are expected to indicate high wear to the model, which is observed here. Indeed, most features have a positive contribution to the tool wear estimation. There are still some features that negatively influence the estimation, but their number and contribution are less significant compared to the others.

7.6 Conclusions

This chapter presented the contributions to indirect monitoring with regression based on the turning dataset. These developments highlight the use of neural networks for monitoring tool wear. In the context of turning operations, it is possible to monitor tool degradation effectively, enabling accurate identification of the tool's condition and end of life.

Two original approaches were presented: the estimation of confidence intervals around predictions and the interpretability of the results. In this chapter, the method for providing confidence intervals is based on a bootstrap approach, while interpretability is achieved through the Explainable Boosting Machine (EBM) method. Both approaches aim to support practitioners in making informed decisions based on model predictions, thereby increasing trust in those decisions.

It is important to note that, in the current developments, no single model is capable of providing both confidence intervals and interpretability simultaneously. This remains an active area of research, with ongoing efforts to integrate uncertainty estimation into interpretable models.

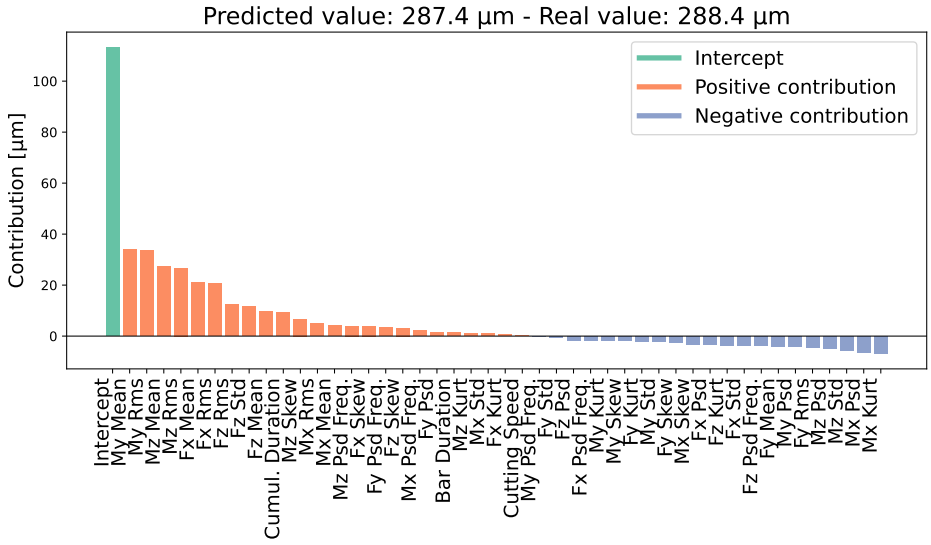


Figure 7.10: Local explanation of one prediction of the EBM model showing the contribution of each input features to estimate the tool wear for the penultimate wear measurement of trajectory 29 (Figure 7.7).

Therefore, it is proposed that further research use these methods in combination: employing a technique such as bootstrap to estimate the prediction and its confidence interval, and supporting the decision with an interpretable model. By combining both approaches, the robustness of tool wear prediction is enhanced, making it easier to decide whether to replace the tool to meet the required quality standard.

Chapter 8

Forecasting the Cutting Tool Degradation from Sensors Signals in Turning

Indirect Forecasting with Regression

This chapter presents results that are being prepared for submission to a journal

8.1 Introduction

In Chapter 7, the objective was to estimate the condition of cutting tools at a given moment. While this is useful for tracking the current state of tool wear, monitoring alone does not provide any insight into the degradation trajectory of the tools. In practice, being able to forecast this degradation is valuable, for instance, to determine whether the tool will be able to complete the upcoming machining operations or if it would be better to replace it beforehand. Although monitoring approaches track tool wear, they do not address this forecasting need.

In the literature, over 98 % of cases focus on using monitoring to estimate the current condition of the tool [33], while only the remaining 2 % attempt to predict the future progression of tool wear. For instance, in [149], researchers use previous VB measurements and signals acquired during machining to predict future degradation with great accuracy. Unfortunately, this still requires stopping the machining process to measure degradation. Another approach performs multi-step prediction in ball end milling using unsupervised learning to identify the most relevant features from vibration and power signals [150]. These features are then used to forecast tool condition with a model combining Long Short Term Memory (LSTM) and convolutional layers, outperforming AutoRegressive Integrated Moving Average (ARIMA) estimates. However, the author suggests that real-time monitoring performance could be improved [150]. The authors of [145] also use a CNN Bidirectional LSTM model for up to 15 steps ahead prediction,

achieving better performance compared to another model from the literature. All these approaches enable forecasting with varying degrees of constraints, whether in terms of precision or data acquisition.

This chapter presents an approach based on Recurrent Neural Networks (RNNs) to predict tool degradation (Figure 8.1). The case study focuses on the production of steel bars from the dataset described in Chapter 6. In this dataset, producing a single bar takes an average of 2 minutes and 45 seconds, during which machining cannot be interrupted. In order to predict if the cutting tool will be able to produce the next bar, this chapter develops a forecasting model capable of predicting the tool condition up to 3 minutes into the future. It is worth noting that the prediction horizon used in this thesis can be adapted to different machining scenarios. Moreover, the developed model is flexible and can be adjusted to forecast alternative prediction horizons depending on the specific application.

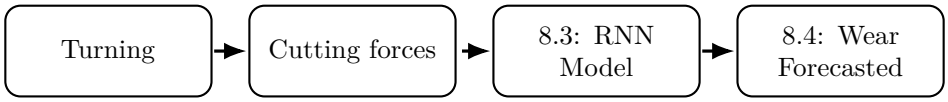


Figure 8.1: Summary of the approach developed in this chapter.

The remainder of this chapter is structured as follows:

- Section 8.2 introduces the problem and the type of model used for forecasting.
- Section 8.3 details the training process and the RNN architectures employed for forecasting.
- Section 8.4 presents the selection of the best model and the overall results on the dataset, highlighting general trends. This section also shows the results for a single tool, illustrating the practical interest of this approach.
- Section 8.5 concludes this chapter.

8.2 Methodology

The case study of this chapter, is the production of steel bars obtained by straight turning, as described in Chapter 6. Given that the cutting process cannot be interrupted during the machining of a bar, it is essential to determine beforehand if the tool will be able to produce the next bar with sufficient quality. Therefore, the objective is to forecast the degradation of the cutting tool to ensure that it is replaced at the most optimal moment. If the wear forecast indicates that the tool will have too much wear and thus will not be able to produce the required quality, it can be replaced before starting the production of the next bar. In this chapter, the assumption is that the cutting tool degradation is a direct indicator of machining quality. Figure 8.2 provides a schematic illustration of the objective of the chapter. At time T, the aim is to be able to predict not only the current degradation points but also the “H” future points (in red in Figure 8.2). To achieve this, it is proposed to use the measurements acquired during the “P” previous measurement points to predict the next degradation points.

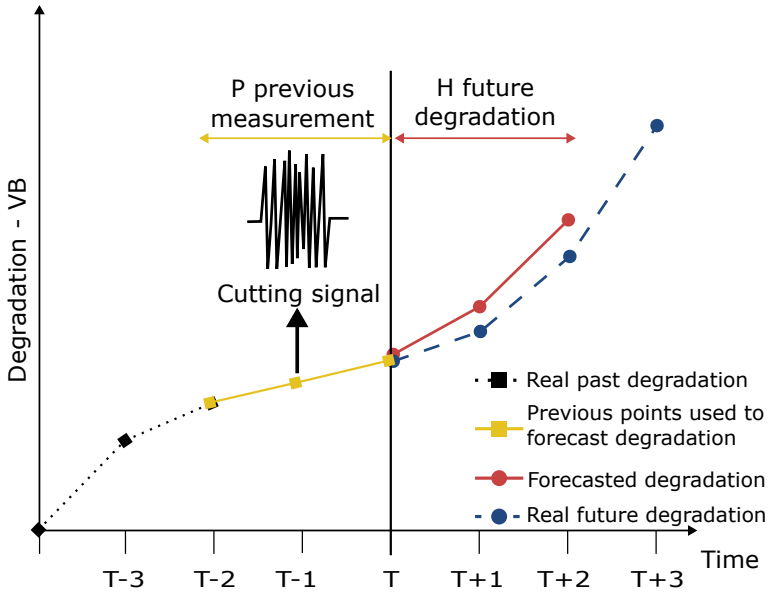


Figure 8.2: Forecasting Principle. Uses the previous points measured in the past to predict the future evolution of the degradation.

A model capable of carrying out this task is therefore a model such as the one shown in Figure 8.3. The input to the model consists of a matrix containing the features X_i acquired at the previous P time points. These features are signals measured at the P past points up to time T. The output of the model is a table that contains the future tool degradation over the next H time points. In this chapter, the model will be a recurrent neural network (RNN), as these are well suited to handle these types of problem [151].

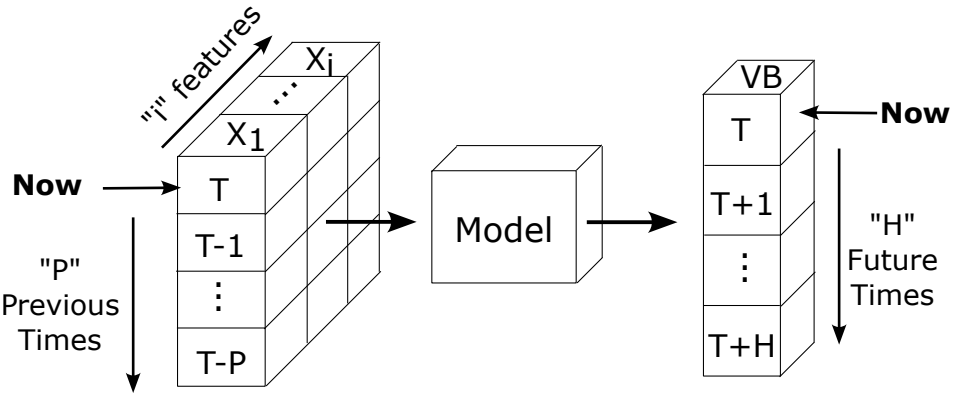


Figure 8.3: Schematic representation of the model used. In entry a matrix containing the timed features is given. In output the model forecast the H previous evolution of the wear of the tool.

8.3 Learning Process

8.3.1 Training parameters

Before training the model, it is necessary to clearly define its objectives, particularly in terms of the prediction horizon (H) and the number of points used to make the prediction (P).

In this case study, producing one bar takes about 2 minutes and 45 seconds. Thus, a prediction horizon (H) of 3 minutes is set to estimate tool wear at the end of each bar's production. This helps determine if the tool will exceed the wear criterion during machining. It is also possible to define the number of intermediate points, which determines the time delta between two predictions. For example, with a 3-minute delta, predictions will be made for T (VB T+0) and T+3 (VB T+3). For simplicity, a time base of 1 minute has been chosen, meaning the time between two outputs is 1 minute.

The parameter P allows the model to use past data for future predictions. If P is 4 and the time delta is 1 minute, the model needs 4 minutes of data before making a prediction. Through trial and error, it was found that values of P greater than 2 did not improve performance.

All the models are trained with the same set of learning hyperparameters. The batch size is set to 8, the learning rate to 1e-4 and the maximum number of epoch is set to 1500.

All these parameters are synthesized in Table 8.1.

Table 8.1: Model and learning parameters of the forecasting models.

	Parameter	**Value**
Model parameters	Horizon (H)	3
	Past values (P)	2
	Delta time [min]	1
Learning parameters	Batch size	8
	Learning rate	1e-4
	Epoch (max)	1500

8.3.2 Censored Learning

During the learning phase, the network may need to predict results that extend beyond the time frame of the collected data. In such cases, it is not possible to calculate an error for these predictions, as their real values are unknown. This inability to calculate an error makes it impossible to optimize the network based on these points. To address this issue, a technique known as censored training is used [152]. This technique excludes all future predictions that extend beyond the range of the measured data from the network optimization process. This approach allows the network to learn in situations where the prediction overlaps the boundary between the known and the unknown. This concept is illustrated in Figure 8.4, where the prediction extends outside the measured time frame. In this case, only

the points within the measured time frame are evaluated during the network optimization process. The points outside the measured time frame are predicted but not used in the optimization process. This technique allows the network to be trained in a greater variety of scenarios, especially the ones that are incomplete at the end of the life of the tool. By applying censored learning, the dataset is utilized more effectively, allowing the model to learn from partially observed cases too. Ultimately, this approach aims to improved forecasting performance by leveraging all available information.

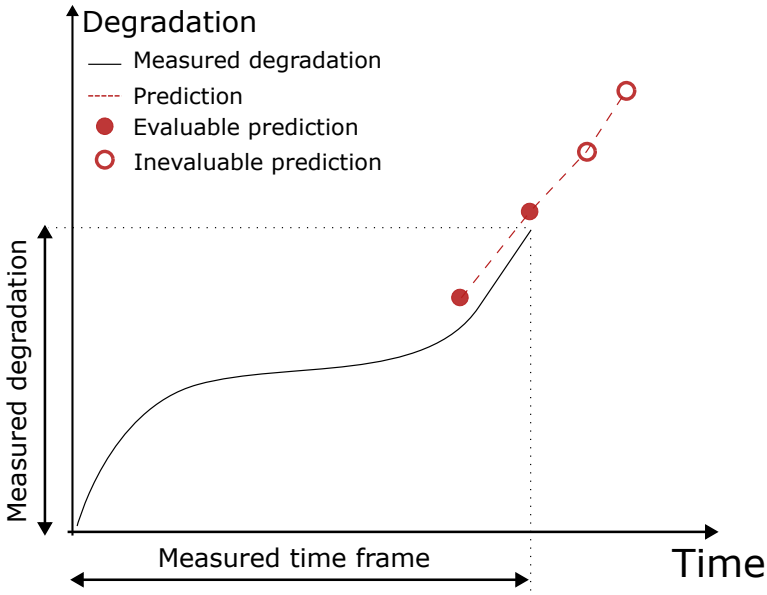


Figure 8.4: Concept of censored learning. It is not possible to evaluate the error outside the range of measurement. Nonetheless, some prediction overlap evaluable and inevaluable part of the dataset.

8.3.3 Network Architecture

Various architectures were evaluated to identify the most effective one. All models are based on recurrent neural networks and incorporate one of three types of recurrent units: the standard Recurrent Neural Network units (RNN), Gated Recurrent Units (GRU), or Long Short-Term Memory units (LSTM). Note that throughout this chapter, the term "RNN" is used interchangeably to refer both to recurrent neural networks in general and to the specific type of recurrent unit. The different variations of recurrent units are described hereafter [153]:

- **Recurrent Neural Network** (Figure 8.5 left): These are a type of neural network particularly suited for processing time series. They contain a hidden state capable of "retaining" information from the previous element. At each step, an RNN takes an input and its hidden state to produce an output and a new hidden state.

- **Long Short Term Memory** (Figure 8.5 middle): These are recurrent networks with an architecture that allows them to better "retain" long-term relationships. Their architecture consists of an input gate, output gate, and forget gate. They are more complex than RNNs and can therefore be more difficult to train [153, 154].
- **Gated Recurrent Units** (Figure 8.5 right): These are also a type of recurrent network. Their architecture lies between LSTM and RNN. They are simpler than LSTMs but can retain more information than an RNN. They are often a good compromise between LSTMs and RNNs.

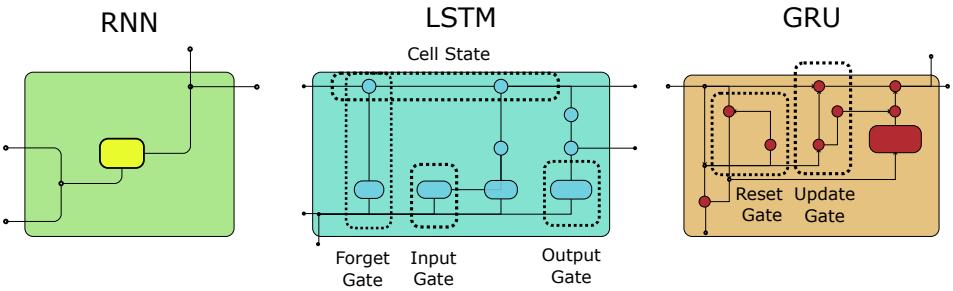


Figure 8.5: Schematic representation of the architecture of the RNN, LSTM and GRU neurons (or units).

Each architecture has multiple variations: Standard, Bidirectional, Stacked, and Bidirectional Stacked. The different variations of the architecture are described below:

- **Standard**. This designation indicates that only the recurrent layers are used. No layers are added between the recurrent layers and the outputs.
- **Bidirectional**. A bidirectional recurrent network is a network that processes data from the past to the future and from the future to the past [155]. In the context of this chapter, this corresponds to looking at input values from T-P to T+0 and from T+0 to T-P.
- **Stacked** (optional stacked layer in Figure 8.6). A stacked recurrent network indicates that after the recurrent layers, fully connected layers are implemented.
- **Both**. A combination of stacked and bidirectional architecture.

Altogether, this results in 12 distinct architectures (3 types of units in 4 different variations). Figure 8.6 show the global architecture of the tested networks. For all architectures, the recurrent layers contain 2 layers of 20 units (40 for bidirectional). Between these two recurrent layers, a dropout layer is added with a drop probability of 10 %. If the architecture contains stacked layers (optional layers on Figure 8.6), it consists of 2 layers each containing 10 neuron with the ReLU activation function. Finally the output layers consists of the different states of the tool for the different timesteps (T+0 to T+3).

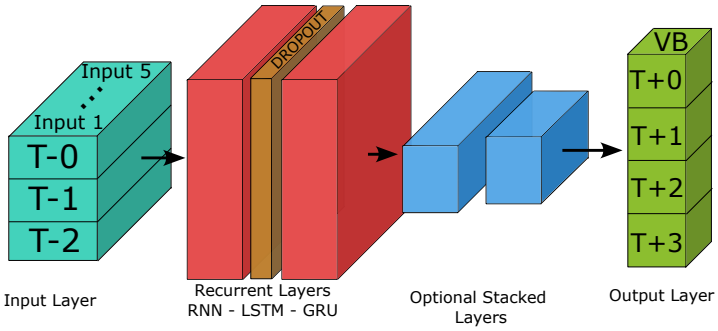


Figure 8.6: Global network architecture layout. The recurrent component consists of multiple layers of RNN, LSTM, or GRU units, optionally followed by stacked fully connected layers.

8.4 Results

8.4.1 Best model selection

To determine the most suitable architecture for monitoring and predicting the degradation of cutting tools, a 5-fold cross-validation method is used. This involves training each network on 4/5 of the dataset trajectories and testing the model on the remaining 1/5. By repeating this process five times, the entire database is evaluated. This procedure is repeated for the 12 tested models. This training approach differs from the one used in the previous chapter, where only test trajectories were considered. In the context of recurrent networks, it was preferable to adopt methods that evaluate performance across the entire dataset to ensure that the presented results are as representative as possible. This choice is motivated by the fact that RNNs tend to exhibit greater variability in their outputs compared to the previously developed methods. For this reason, the following section first presents the overall results to confirm that general trends are well captured, before focusing on the performance observed on a testing degradation trajectory.

Figure 8.7 compares the MAPE of the different architectures across various prediction horizons (T+0 to T+3), with lower MAPE values indicating better performance. The figure shows that, in general, the results are quite similar; however, two groups stand out: the stacked architectures and the “both” group.

The analysis of MAPE is an interesting indicator, but it only provides an average value and does not account for the dispersion around this value. The goal of this chapter is not only to monitor degradation but also to detect when the tool exceeds the end-of-life criterion. It is possible to calculate the Remaining Useful Life (RUL) using the model’s predictions. The RUL is calculated when the prediction exceeds the threshold value of 200 µm. This threshold differs from the end-of-life criterion typically used in monitoring applications, which is set at 300 µm. The decision to lower the threshold to 200 µm in this chapter is due to the fact that the dataset contains very few tools with wear levels exceeding 300 µm. Attempting to compute and compare RUL values using a 300 µm threshold would result in too few usable cases. Therefore, the threshold was reduced to 200 µm to ensure that performance

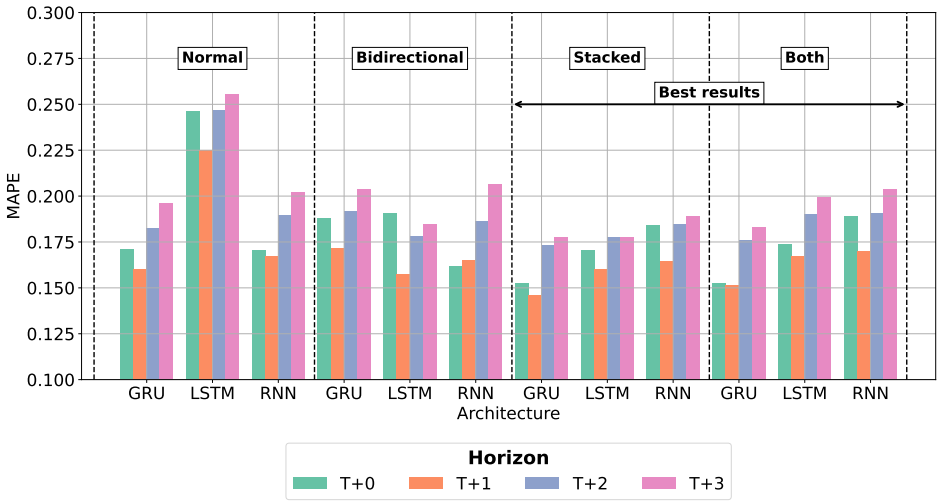


Figure 8.7: Comparison of performance for different architectures. The lower MAPE values indicates more accurate results.

could be evaluated across the entire dataset, as all tools in the database reached at least this level of wear.

Figure 8.8 shows the error distribution between real and estimated RULs. A positive error means the model predicted a shorter RUL, which is safer, while a negative error means it overestimated the RUL, which is less safe. The figure uses a violin plot to show this distribution, with the white dot and its line representing the mean, and the lower and upper lines indicating the 1st and 3rd quartiles, respectively.

The analysis of the RUL dispersion shows that the 'Stacked GRU' and the 'Both RNN' have similar dispersions. However, the MAPE of the 'Stacked GRU' approach is better than that of the 'Both RNN'. Therefore, the GRU stacked architecture is used for all results in this chapter

8.4.2 Overview of the Results

With the best architecture identified in the Section 8.4.1, a more in depth analysis of the trends in the results on the entire database is presented. To observe the quality of the prediction across the entire database (split with a 5 K-Folds), Figure 8.9 shows the distribution of errors between the predictions and the actual values for the 4 output states (T+0 to T+3). On this figure, a positive error means that the model overestimate the actual wear. This side of error is therefore safer as it leads to an earlier tool replacement. The MAPE for each horizon is also recalled. This figure indicates that the distribution of residuals for all predictions is more or less centered around 3 µm, suggesting that, on average, the error between the predicted value and the actual wear value is approximately 3 µm for all prediction horizons. It is important to highlight that such an average error is nearly negligible in this

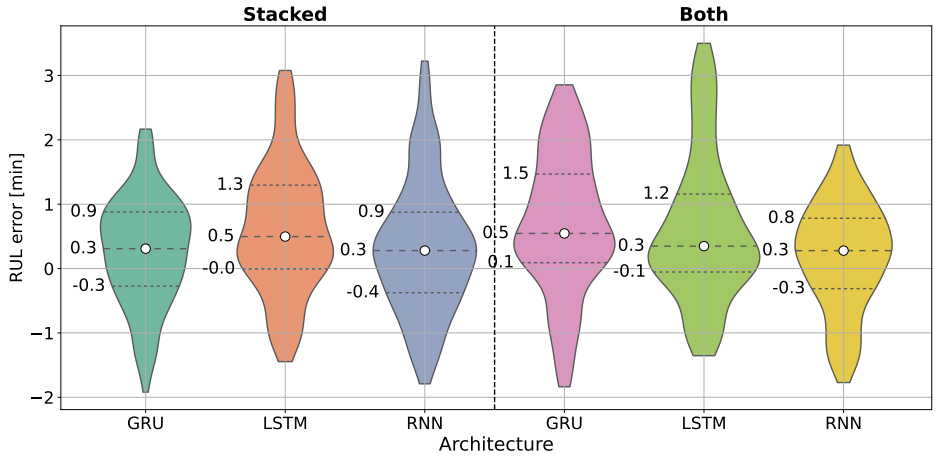


Figure 8.8: Violin plot showing the distribution of error between the real RUL and the predicted one. A mean value (represented by the white dot) closer to 0 indicates better results. The shape of the distribution also indicates how it is spread around the mean value The upper line indicates the third quartile, the middle line with the white dot shows the mean value, and the lower line represents the first quartile.

context, as it represents roughly 1 % of the total measured range. This level of error is even smaller than the variability typically introduced by manual tool wear measurements.

All distributions have a fairly high positive lobe due to certain trajectories in the database that were not correctly followed by the model. However, as the distribution shows, this only represents a few percent of the database. These errors may be due to experimental conditions. For T+0 and T+1, the distributions are similar, with 50 % of the points falling between -10 and 10 µm of error. The T+2 distribution shows that the model overestimates the degradation at T+2 more than previously. For the T+3 distribution, it is slightly more spread out than for the other predictions. This increase in uncertainty as we move away from T+0 is consistent with the uncertainty of the manufacturing process, which is itself highly dispersive. These errors can be compared to classical monitoring methods that attempt to estimate the tool's state at T+0. The Chapter 7 describes the optimization process of a neural network that allows tool monitoring on the same database. In this chapter, the optimal performance for estimating the wear at T+0 is 15 %, which is the same values found here (Figure 8.9). This shows that for the database used in this study, the forecasting model is able to track the real-time degradation of the tool as accurately as a dedicated monitoring network.

Figure 8.10 shows the distribution of the RUL estimation when the end-of-life criterion is set at 200 µm, as in the previous examples. Most predictions allow the RUL to be estimated within one minute. Given the application case of this chapter, having a predicted RUL with an error of less than 2 minutes and 45 seconds indicates that we will be able to detect tool wear in advance in most cases. For this figure, a positive error indicates that the actual RUL is greater than the predicted

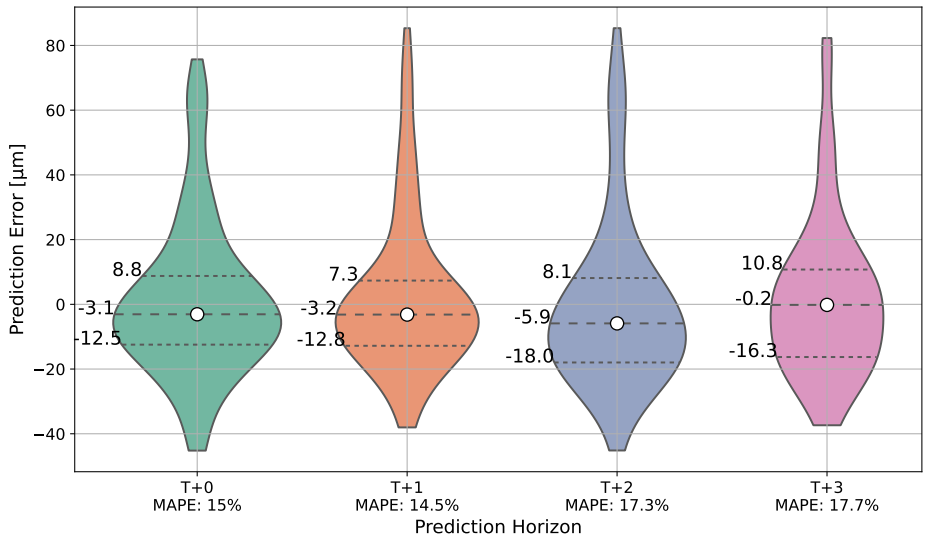


Figure 8.9: Violin plot of error for the different time horizon of the best model.

RUL, which shows that the model errors on the side of safety in estimating the tool's end of life.

In the case study focused on steel bar production, the model accurately predicts the tool's state at the end of the next bar's production. From an industrial perspective, this approach helps in determining the optimal time for tool replacement. If the predicted degradation exceeds the end-of-life threshold, it is advisable to replace the tool before starting the next bar's production. This is also observed when an end-of-life value is set to compute the effective RUL of the tool. In this case, an error of less than 1 minute is generally observed. This helps prevent potential poor-quality production.

8.4.3 Detailed Results – Example on one tool degradation trajectory

The trajectory chosen to illustrate the model's performance is trajectory 28 from the database (Figure 8.11). This trajectory is selected because it exhibits an interesting degradation pattern. The degradation was measured for just over 20 minutes. Notably, from 17 minutes onward, the degradation increased significantly, rising from 230 µm to 400 µm of wear in just 2 minutes and 30 seconds. This test trajectory is an effective way of determining whether the model can detect variations in the rate of degradation.

To test the proposed approach, the network is tasked to forecast the tool's degradation for the next 3 minutes at the end of each machining bar. Figure 8.12 summarizes the different stages of these predictions at the start of each bar's production. A detailed description of each prediction step is provided below:

- Figure 8.12a shows the model's prediction at the beginning of the production

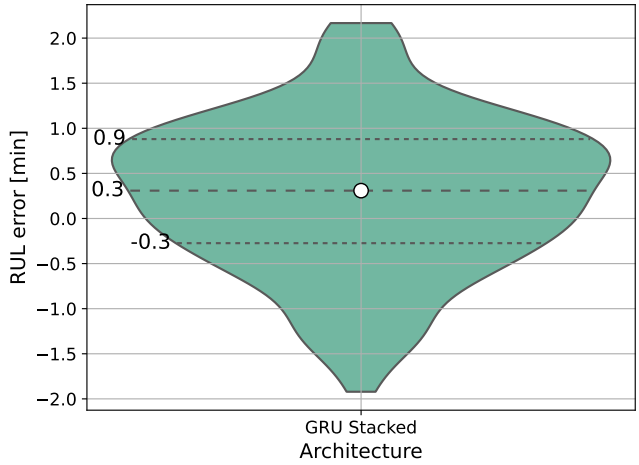


Figure 8.10: Violin plot of the error of the estimated RUL with the best model.

of the third bar. The prediction, in blue, shows the 4 points predicted by the model (T+0 to T+3). The prediction estimates that 3 minutes later, the tool's degradation will be below 100 µm.

- Figure 8.12b shows the model's prediction at the beginning of the production of the next bar. As before, the model correctly predicts the next 3 minutes of production.
- Figure 8.12c shows that the model can observe the slight change in the degradation rate. It slightly overestimates the actual degradation compared to previous predictions. Nonetheless, the error is limited to a few µm which is acceptable.
- Figure 8.12d is interesting because it shows the moment when the degradation starts to increase drastically. At this stage, the network predicts an evolution quite consistent with the previously observed prediction. The model slightly overestimates the degradation at the end of the new bar's production.
- Figure 8.12e shows the moment when the tool's degradation increases drastically. It appears that the model was able to detect that the degradation would increase and estimates that the degradation is around 400 µm. For the sake of experiments, the tool was used until 400 µm in this example. Industrially, that amount of wear is often considered as excessive and the tool would have been replaced before arriving to this state. Therefore, industrially, this prediction should warn the machinist to change the tool to avoid producing poor quality.
- Figure 8.12f illustrates the potential degradation of the tool if machining had continued. The tests were halted, making it impossible to verify the accuracy of the prediction. Nevertheless, based on the previous degradation rate, it is reasonable to assume that the predicted degradation would be plausible.

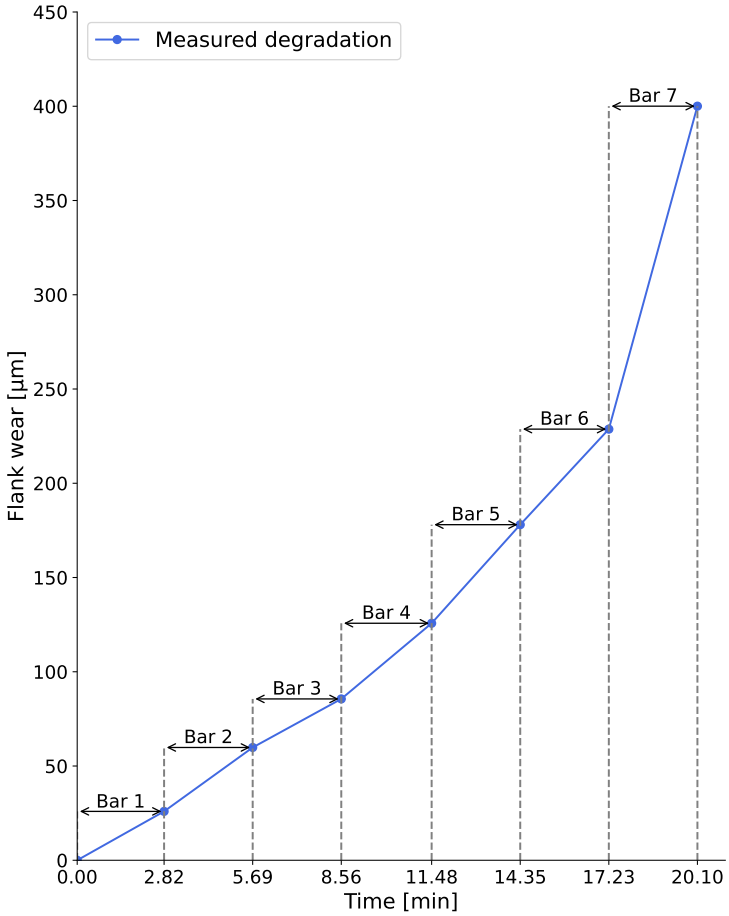
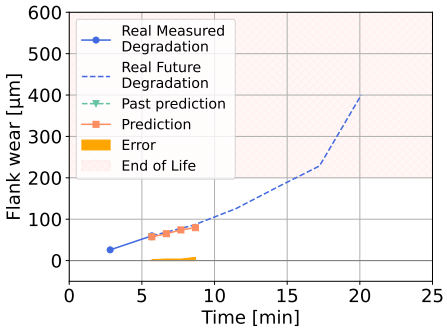


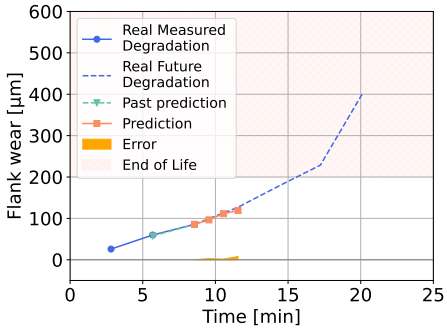
Figure 8.11: Degradation trajectory of Tool 28 with produced bars.

Figures 8.13a to 8.13d show a summary of all predictions, presenting each forecast individually. It is observed that the predictions are very accurate and that each forecast horizon correctly follows the tool's degradation.

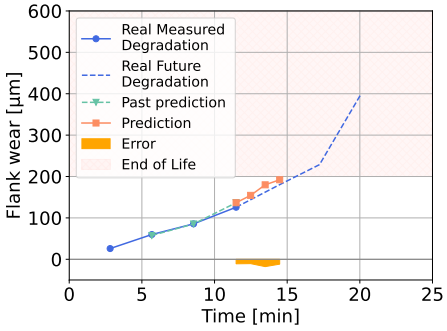
The model's predictions are used to calculate the Remaining Useful Life (RUL) with an end-of-life criterion set at 200 µm (Figure 8.14). The predicted RUL is calculated when the model's VB prediction exceeds the imposed limit of 200 µm. This is why the RUL only appears 3 minutes before the end of the tool's life, as the prediction horizon crosses the usage limit at that point. Given that the forecast closely follows the actual degradation, the predicted Remaining Useful Life (RUL) here almost perfectly matches the observed RUL. At the end of machining bar 5 (Figure 8.14), the predicted RUL is 62 seconds, while the actual RUL is 75 seconds, resulting in an error of 10 seconds on the safe side. Since machining one bar takes more than 62 seconds, this forecast should trigger a tool replacement.



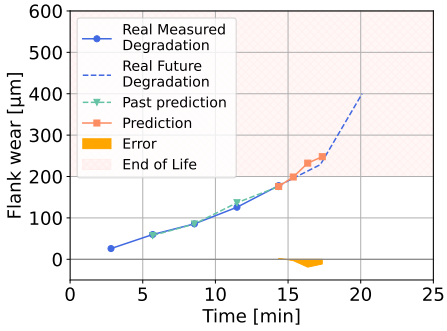
(a) 1st wear forecast. Before producing bar n°3



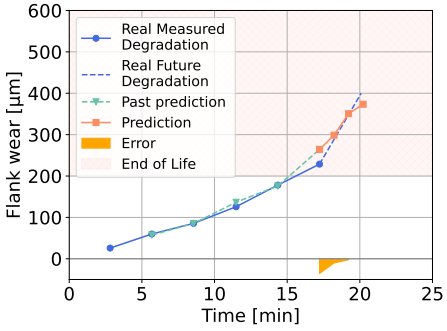
(b) 2nd wear forecast. Before producing bar n°4



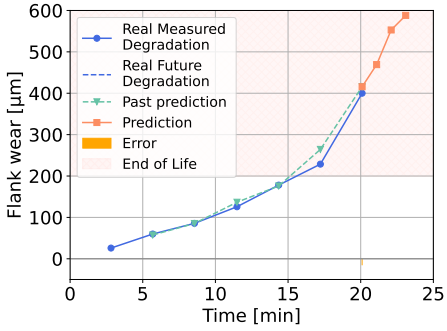
(c) 3rd wear forecast. Before producing bar n°5



(d) 4th wear forecast. Before producing bar n°6

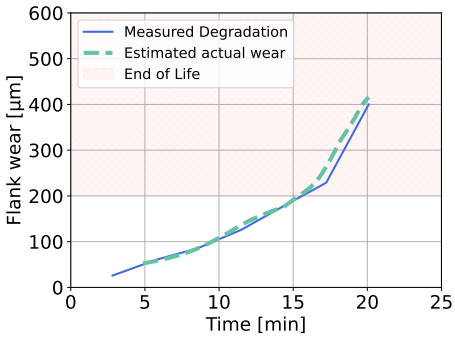


(e) 5th wear forecast. Before producing bar n°7

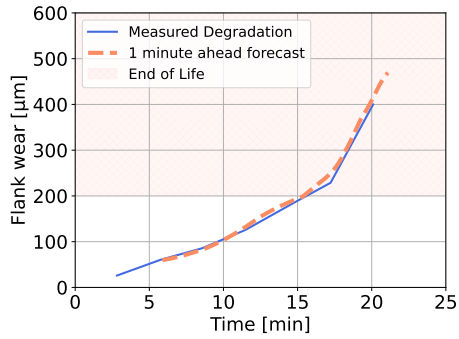


(f) 6th wear forecast.

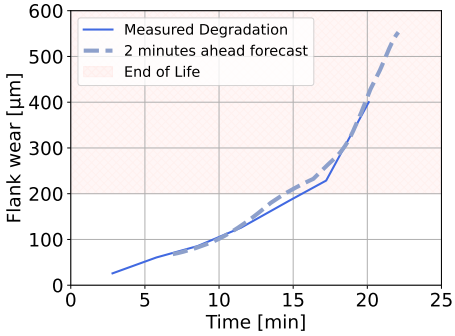
Figure 8.12: Different stages of wear forecast during the life of the tool 28. The blue line represents the actual degradation, the light-green line shows the previously estimated wear by the model, and the dashed line indicates the future evolution of the tool. The orange line illustrates the network's prediction for the next 3 minutes of machining. Finally, the orange area represents the error between the prediction and the forecasted degradation.



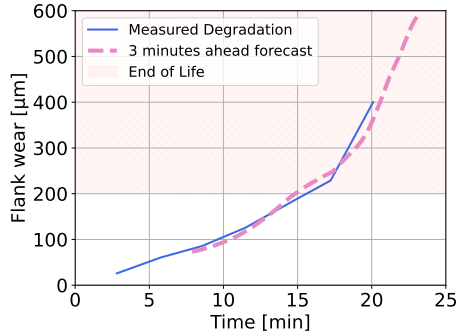
(a) Prediction T+0 – Actual wear



(b) Prediction T+1



(c) Prediction T+2



(d) Prediction T+3

Figure 8.13: Summary of the prediction of the wear divided by horizon of prediction.

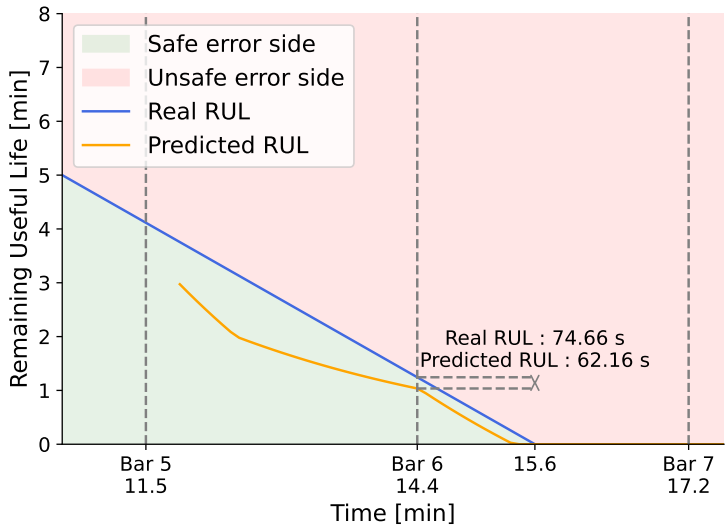


Figure 8.14: Zoom on the Remaining Useful Life predicted by the model when its forecasting crosses the limit of 200 µm for VB.

8.5 Conclusions

This chapter presents the use of recurrent neural networks to monitor and predict the degradation of cutting tools. The case study focuses on the production of C45 steel bars using a straight turning process. Each bar takes 2 minutes and 45 seconds to produce. The developed model can predict the tool degradation up to 3 minutes into the future. This allows for estimating the tool's condition at the end of the next bar's production, enabling optimal tool replacement timing. Several conclusions can be drawn:

- Several recurrent networks were tested: RNN, LSTM, and GRU, in different configurations: standard, bidirectional, stacked and bidrectionnal-stacked ("both" configuration). All networks achieved similar performance, with MAPE between 15 % and 25 % on predictions. The two best models are the stacked LSTM and the stacked GRU. These models have very similar performance, but a more in-depth analysis shows that the error distribution is smaller for the GRU than for the LSTM.
- The performance of the best model is evaluated across the entire database. In over 50 % of cases, the model can identify the tool's condition at time T+0 with an accuracy of ±10 µm. For future predictions, the accuracy slightly decreases, but an average error of less than 20 µm is observed. The estimation of the Remaining Useful Life (RUL) identifies the tool's end-of-life with an average precision of less than one minute. These findings demonstrate that the method is sufficiently accurate to support concrete decision-making regarding whether or not to replace the tool.
- The model's performance in predicting the current state of the tool (T+0) can be compared to a monitoring model (Chapter 7). The monitoring accuracy is equivalent, indicating that the forecasting model provides monitoring as accurate as a dedicated model while also enabling future predictions of the tool's condition.
- The model's performance is also presented for trajectory 28 to detail the results obtained. The results, tracked throughout the tool's lifespan, demonstrate excellent degradation monitoring with an error of less than 10 µm on this trajectory. Additionally, the Remaining Useful Life (RUL) is accurately estimated with an overall error of less than 10 seconds.

The methodology presented in this chapter is broadly applicable, enabling industries to anticipate tool degradation and take preventive action before defective parts are produced.

Chapter 9

Classifying the Cutting Tool State from Tool Images in Turning

Direct Monitoring with Classification

The results presented in this chapter have been published in a scientific article [156], from which most of the chapter's content is derived.

9.1 Introduction

The turning dataset includes images of tool wear, which enables the development and evaluation of models based on direct monitoring approaches. The direct approach consists of taking the images of the cutting tool to assess if the tool is still able to perform the cutting operation. This method is often more accurate than indirect ones, but if conducted by humans, it can yield more uncertain results due to individual interpretations and definitions of wear. Consequently, several image analysis techniques and tools are available [157]. However, these images, frequently taken under varying lighting and masked by cutting fluid or chips, make wear detection challenging for traditional image analysis. Nevertheless, artificial intelligence methods overcome these issues, by their ability to reliably identify wear even in such conditions [157]. Numerous AI methods have been employed to identify or categorize the region of tool wear using images. For example, Pagani et al. utilized the colour of the chips to infer the tool's condition [158]. In 2019, Wu et al. compared their own model with VGG16 [159] to distinguish between different types of wear. Their model was quicker, but the performance was comparable between the two models [160].

In the field of image classification, it is generally not recommended to build models from scratch. Indeed, deep learning models generally require large amounts of data and meticulous tuning to guarantee robust and reliable performance [161]. This is why transfer learning is often the preferred approach in this type of application [162]. It offers a significant advantage by providing models that are

inherently more robust than those built from scratch. Despite the existence of numerous approaches for tool condition monitoring, the significance of transfer learning and the applicability of methods developed for image classification tasks remain unclear in the literature (Chapter 4).

In this chapter, it is proposed to use and compare 3 AI methods for image classification using transfer learning. The 3 baseline architectures are: VGG19 [159], EfficientNetV2 [163] and Vision Transformers [44] (Figure 9.1). These techniques are evaluated on modified images that reflect those acquired in an industrial environment, with modifications including rotation and adjustments to brightness and contrast. This chapter is structured as follows:

- Section 9.2 explains how the dataset is prepared for classification tasks, including the preprocessing steps and modifications applied to the images.
- Section 9.3 describes the three models used and how they are adapted for the classification of tool wear images.
- Section 9.4 presents the results of the three models in relation to the image modifications, highlighting the most robust approaches. This section also includes a discussion on the impact of classification errors.

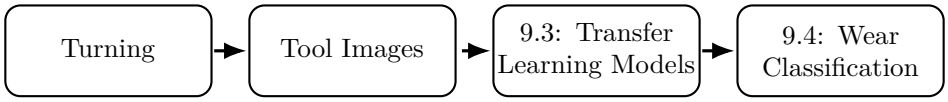


Figure 9.1: Summary of the approach developed in this chapter.

9.2 Classification and Modification of Images

A total of 30 tools were used during the straight turning testing campaign (see Chapter 6). On average, each tool was inspected 6 to 7 times during its life. The inspection consists of taking a picture of the insert. The database therefore contains 192 images of the flank face of the tool. The tool was inspected every 2 minutes 45 seconds using a Bysameyee EU-1000X 3 digital portable microscope. The images obtained are colour images with a resolution of 460 by 640 pixels from which the tool wear is measured. An example of image acquired during the experimental campaign is presented Figure 9.2.

In this chapter, AI is used to classify the state of the tool. It has been chosen to classify the state of the tool into 3 different classes, each of them representing a part of the tool's life (Figure 9.3). A tool with wear between 0 and 150 µm is deemed new, one with wear between 150 and 300 µm is considered with moderate wear, and a tool with wear exceeding 300 µm is classified as worn. This classification system is structured to take account of the different stages in the life of a cutting tool. Initially, in class 1, a new tool undergoes rapid degradation until it moves into class 2, where it remains for most of its service life. During this class 2 phase, the rate of degradation of the tool slows down compared with class 1. Finally, after a few minutes, the tool suffers excessive wear, leading to rapid deterioration

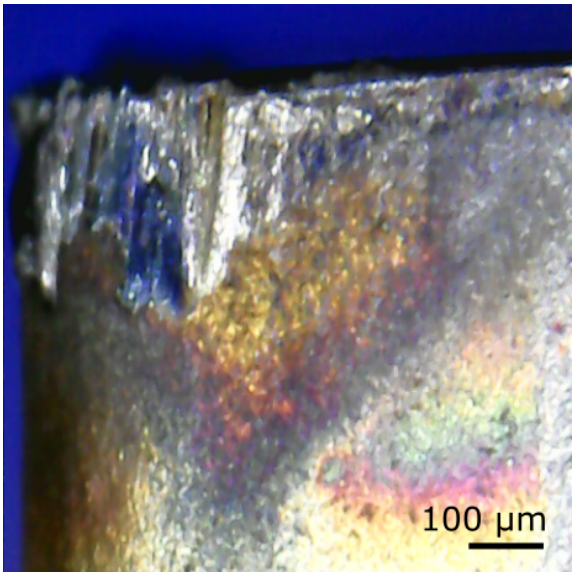


Figure 9.2: Image of an insert (VB: 280 µm).

that brings it into class 3. In this last class, a tool is considered to be "worn" if it has a flank wear of more than 300 µm. This threshold is established by the ISO 3685 standard [126]. This degradation pattern is observable in the database. However, the dataset lacks homogeneity: there are more images of new tools than worn tools, which can be a limitation when training AI. Specifically, there are 141 images in class 1, 38 in class 2, and only 13 in class 3. To counteract this imbalance, the training dataset is oversampled. This technique duplicates the number of images in a class so that there are as many images in each class. This method's drawback is the potential repetition of the same image multiple times in the training database. To mitigate this issue, data augmentation is used, so that even 2 identical images are not augmented twice in the same way, thereby expanding the diversity of data for training. Data augmentation is frequently implemented in databases to compensate for a limited quantity of images and to generalize images under conditions that are unevenly represented in the database. Data augmentation is used here to generate new, unique images from the images in the database [164]. Two types of augmentation are taken into account: image manipulation and lighting modification. During training, each image is randomly augmented using Pytorch [46] with a combination of the following modifications (Figure 9.4):

- Image manipulation:
  - **Horizontal flip**. There's a 50 % probability that the image will be horizontally flipped, resulting in a mirrored version of the original. This modification applied on Figure 9.2 is shown in Figure 9.4a
  - **Rotation**. This technique rotates the image. The rotation angle is randomly chosen between -20 to 20°. This modification applied on

Figure 9.2 is shown in Figure 9.4b

- Lighting modification:
  - **Contrast modification**. This technique modifies the image contrast by a factor randomly chosen in the range -0.2 to 0.2. This modification applied on Figure 9.2 is shown in Figure 9.4c
  - **Brightness modification**. This technique randomly change the brightness of the image by a factor randomly chosen in the range -0.2 to 0.2. This modification applied on Figure 9.2 is shown in Figure 9.4d

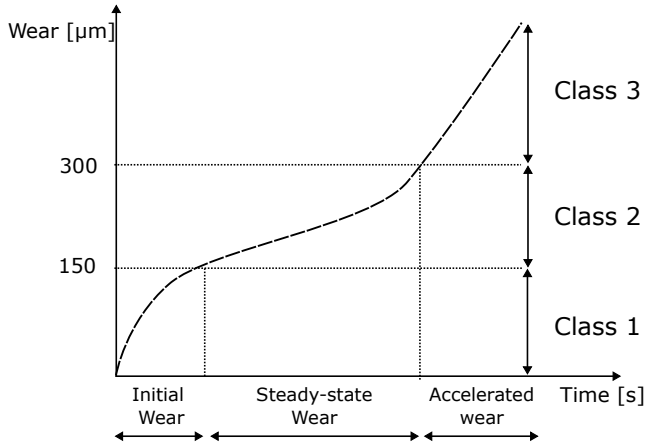
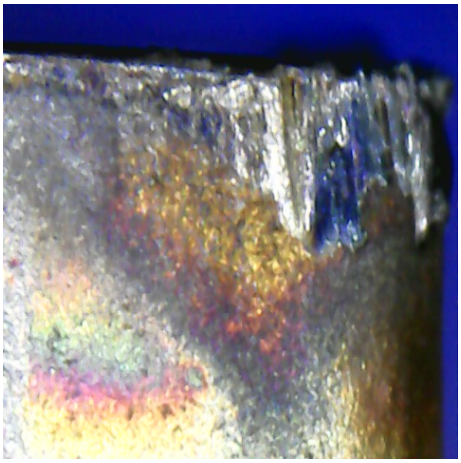


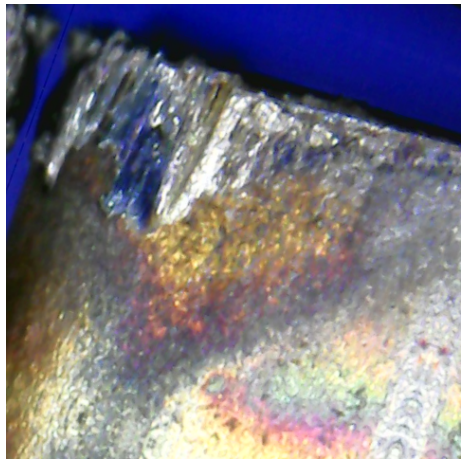
Figure 9.3: Classification of the tool degradation into 3 classes.

The selection of these modifications is driven by the objective to reproduce any distortion on image acquired industrially. Adjusting the image orientation is intended to replicate the inconsistencies and imperfections of real-world image capture. Modifying the image by altering its contrast or brightness mimics the disruptive elements of the industrial environment which prevent measurements being taken under constant and controlled lighting conditions. The training domain of the networks therefore consists of images of a single type of tool machining at different cutting speed and acquired under different lighting conditions.

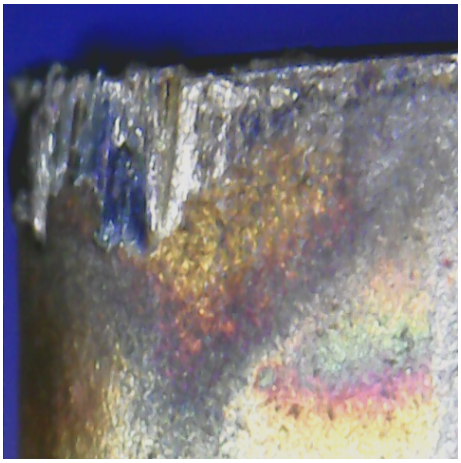
In order to test the ability of an AI approach to learn this training domain, a testing database needs to be defined. This database must represent all existing cases and it consist of 5 images of each class. Each of this image is tested six times under distinct configurations: once in its original form, once flipped, once rotated, once after adjusting its contrast, once after altering its brightness and finally once with all the previous modifications (Figure 9.4). This process ensures that each test image is thoroughly evaluated under various conditions, enhancing the robustness of the testing phase.



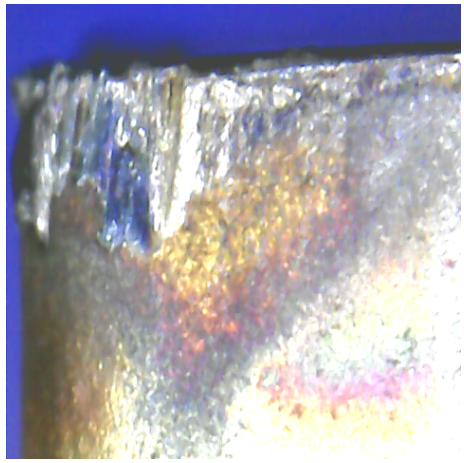
(a) Horizontal flip



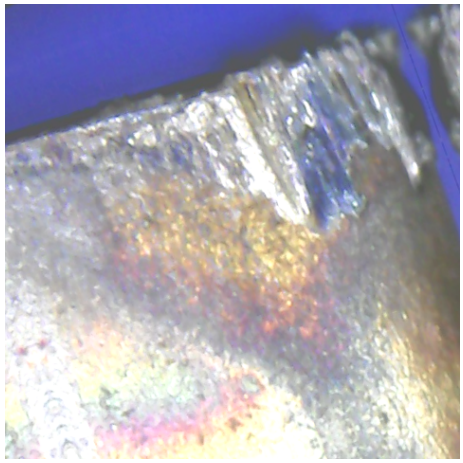
(b) Rotation



(c) Contrast modification



(d) Brightness modification



(e) All transformations

Figure 9.4: Image transformations applied to the image from Figure 9.2.

9.3 Models for Transfer Learning

In the following, different pre-existing models will be used. These models are selected based on their performance in different classification tasks as well as the different approaches behind them. These models are:

- **VGG19** [159]. Developed in 2014, VGG19 is a deep convolutional neural network that aims at simplicity. It is composed of 16 3×3 convolutional layers and 3 fully connected layers. It was trained on the ImageNet dataset (14 millions images) to classify a thousand different classes. Its architecture is often used as a reference in image classification.

- **EfficientNetV2-M** [163]. Introduced in 2019, it is an extension of the initial EfficientNet. The particularity of this model lies in its balance between model size, performance and computational efficiency. It was also trained on ImageNet. The EfficientNetV2 model family includes several variants, in this case the M stands for medium and is selected for its compromise between complexity and accuracy.

- **Vision Transformers (ViTs)** [44]. The Transformer model, originally designed for natural language processing, was adapted in 2020 for image data, leading to the creation of ViTs. ViTs work by dividing images into patches, each of which is converted into a vector and processed by a transformer encoder (see Chapter 2). Unlike Convolutional Neural Networks (CNNs) that mainly capture local features, ViTs excel in identifying both local and global features, including long-range dependencies between patches. This makes them especially useful for tasks requiring a comprehensive understanding of an image. Since 2020, ViTs have generated significant interest in the field of computer vision.

The choice of these models is driven by the intent to contrast three distinct approaches: an initial "classic" model, an improved version of this type of model, and finally an innovative approach to vision through transformers. This provides a comparative perspective on the different image processing methods. As these models have all been pre-trained on ImageNet, the entries for these networks are colour images of 224 by 224 pixels. The images in the database are therefore resized to be compatible with this shape. The number of parameters for each model is listed in Table 9.1, along with their respective GFLOPS values. GFLOPS (Giga Floating Point Operations Per Second) is a unit of measurement for computational performance, representing one billion (10^9) floating-point operations per second. This table illustrates the complexity of the models in terms of the number of parameters, but not necessarily in terms of computational operations per second. The values presented correspond to the pre-trained portions of the models.

To adapt these models to classify the state of the tool, the last layers are modified to classify the tool state (Figure 9.5). The layers that are added consist of a Maxpooling2D layer, a layer containing 1024 neurons with ReLU activation function, a dropout layer and finally a classification layer representing the 3 possibles states of the tool.

Table 9.1: Number of parameters in the pre-trained components of the selected models. The GFLOPS values indicate their required computational resources.

Model	Num parameters	GFLOPS
VGG19	143,667,240	19.63
EfficientNetV2-M	54,139,356	24.58
ViT	86,567,656	17.56

All the approaches are trained with a maximum of 300 epochs with an AMD Ryzen 9 7950 X3D CPU and RTX 4080 GPU. The input for each model is a colour image with dimensions of 224 by 224 pixels. The loss function "Categorical crossentropy" is used. This loss function is commonly used in multi-class classification problems in AI. It measures the dissimilarity between the predicted probability distribution and the true distribution of classes [165]. The metrics for evaluation are:

- Accuracy. It measures the overall correctness of the model by calculating the ratio of correctly predicted instances (both positive and negative) to the total number of predictions. It is computed as:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{9.1}$$

  With:

  - TP (True Positives): instances correctly predicted as positive,
  - FP (False Positives): instances incorrectly predicted as positive,
  - FN (False Negatives): instances incorrectly predicted as negative,
  - TN (True Negatives): instances correctly predicted as negative.

- Precision. It measures the proportion of correctly predicted positive instances out of all predicted positives, and is defined as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{9.2}$$

- Recall. It evaluates the proportion of actual positives that were correctly identified, given by:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9.3}$$

- F1 score. It is a metric that combines precision and recall into a single value, providing a balanced measure of a model's performance. It is computed as;:

$$\text{F1 score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9.4}$$

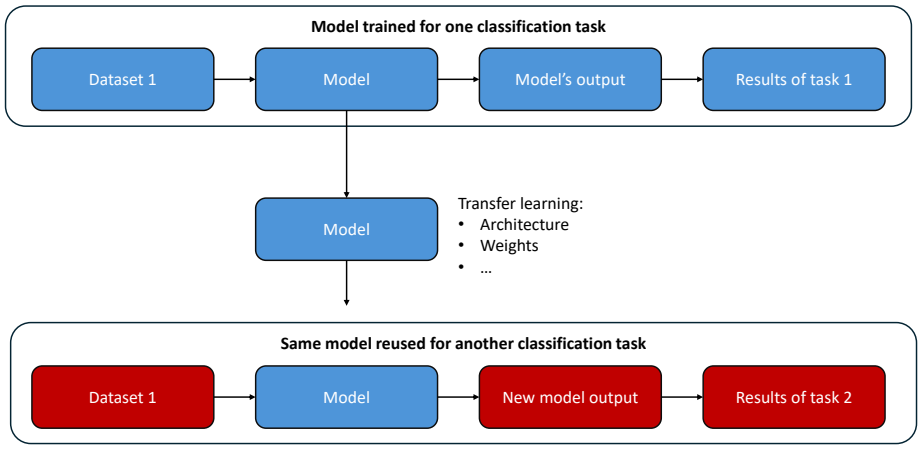


Figure 9.5: Transfer learning principle: A given architecture is reused for a different task. The model's architecture and weights are transferred to the new task, where the output layer is adapted to suit the new objective.

9.4 Results

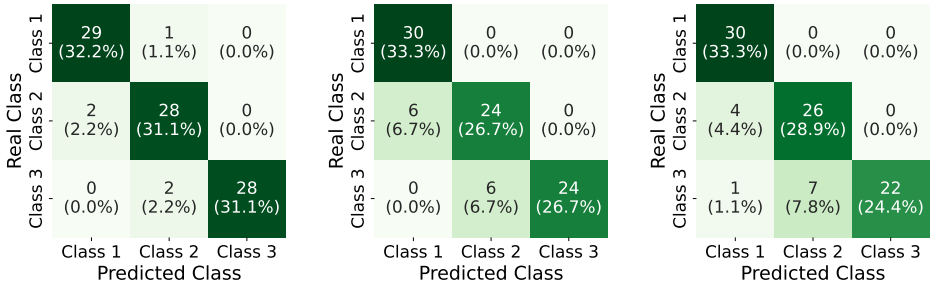
Table 9.2 presents a comparative analysis of the performance across the three network architectures previously discussed. It provides detailed results, segmented by class and the type of image modification applied. Each type of modification is composed of 5 images per class. Therefore there are 15 test images per modification. The following observations are drawn from this table:

- VGG19 is the architecture that obtained the best results in this chapter. It obtained an overall accuracy of 94 % and F1 scores of 0.95, 0.92 and 0.97 for classes 1 to 3 respectively. The model reached convergence after 120 epochs.
- The EfficientNetV2-M model achieved an overall accuracy of 87 %, making it almost as good as VGG19. This model can perfectly identify class 1, but it seems to be less accurate for classes 2 and 3. With the original image (referred to as 'Initial' in Table 9.2), it makes mistakes in identifying class 2 and 3. This model is quicker to train than VGG19 with a convergence in just 65 epochs. To put it in perspective, the EarlyStopping function waits for 60 cycles without improvement before stopping, which means the network converges in just 5 cycles.
- The ViT model achieved the same overall accuracy as EfficientNetV2. However, it tended to classify class 3 less well than the other architectures. This architecture is also the longest to train, with 300 epochs.

Figure 9.6 shows the three confusion matrices associated with Table 9.2. These highlight that VGG19 achieves the highest accuracy, particularly in classifying

Table 9.2: Comparison of accuracy for the different methods segmented by class and the type of image modification applied. C1 corresponds to Class 1, C2 corresponds to Class 2 and C3 corresponds to Class 3.

	VGG19			EfficientNetV2-M			ViT		
Accuracy per Modification									
	C1	C2	C3	C1	C2	C3	C1	C2	C3
Initial	100 %	100 %	100 %	100 %	80 %	80 %	100 %	100 %	80 %
Flipped	80 %	100 %	100 %	100 %	80 %	80 %	100 %	80 %	80 %
Rotated	100 %	80 %	80 %	100 %	60 %	80 %	100 %	60 %	60 %
Contrasted	100 %	80 %	100 %	100 %	80 %	80 %	100 %	100 %	80 %
Brightness	100 %	100 %	100 %	100 %	80 %	80 %	100 %	80 %	60 %
All	100 %	100 %	80 %	100 %	100 %	80 %	100 %	100 %	80 %
Overall metrics									
Precision	94 %	90 %	100 %	83 %	80 %	100 %	86 %	79 %	100 %
Recall	97 %	93 %	93 %	100 %	80 %	80 %	100 %	87 %	73 %
F1 Score	0.95	0.92	0.97	0.91	0.80	0.89	0.92	0.83	0.85
Accuracy		94 %			87 %			87 %	



(a) VGG19 **(b)** EfficientNetV2 **(c)** ViT

Figure 9.6: Confusion matrices for the three tested models. The values are computed based on the results presented in Table 9.2

classes 2 and 3. It is also observed that misclassifications occur mainly between neighboring classes, with no instance of class 1 being predicted as class 3, and vice versa. A straightforward comparison of results serves as a useful performance indicator, but it does not fully determine whether one method outperforms another. This is particularly true when the accuracies achieved by the architectures are quite similar, hence, more detailed analysis are needed. The goal of classifying the tool's condition means that detecting class 3, or a worn tool, is the only classification that can influence the tool's replacement. The wear limit is set at 300 µm, so in a strict case, a tool with 299 µm of wear would still be considered as moderate wear being only 1 µm away from being worn. In practice this is not the case, the boundary between a usable tool and a worn one is not always clear and needs to be taken into account. To illustrate this fact, Table 9.3 shows the position of errors

Table 9.3: Comparison of the position of errors for the different architectures. A transition between classes is added.

	Position of errors in the classification				
Approach	**Class 1**	**Transition 1–2**	**Class 2**	**Transition 2–3**	**Class 3**
VGG19	1	0	2	1	1
EfficientNetV2-M	0	0	6	6	0
ViT	0	1	3	1	7

made by the different architectures in different classes. A buffer zone of 50 µm, 25 for each class, is added between the classes, creating a transition zone between classes 1 and 2, and between classes 2 and 3 (Figure 9.7). The first zone covers wear from 125 to 175 µm (transition from class 1 to 2), and the second covers wear from 275 to 325 µm. Analysis of the results in this zone reveals where the boundary between a usable tool and a worn one becomes unclear. Table 9.3 indicates that the EfficientNetV2 architecture made 6 errors in the transition zone between class 2 and 3. This is higher than the other approaches. However, these errors are of little consequence in practical applications. Indeed, this error is less than 25 µm which have almost negligible impact on the quality of production. Therefore, although EfficientNetV2 has the same accuracy as ViT, its errors have less impact in practice.

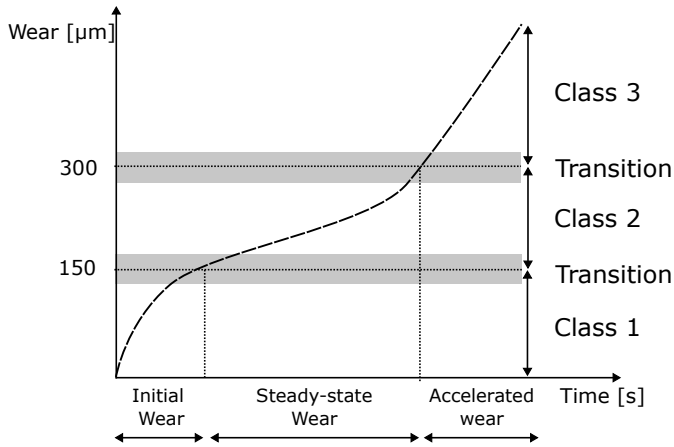
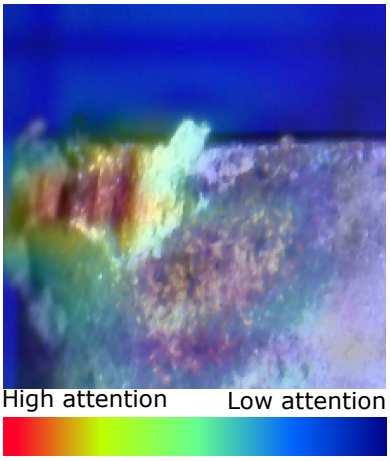


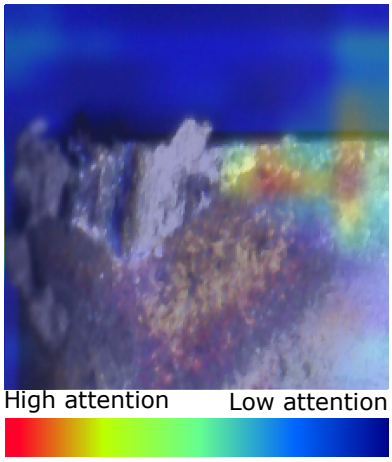
Figure 9.7: Classification of the tool degradation into 3 classes with transition zone between the classes.

In addition to the position of errors, it is also necessary to understand the reason for a correct or incorrect classification. In order to explain and visualise the cause of the classification made by the architectures, a Grad-CAM (Gradient-weighted Class Activation Mapping) method is used [166]. Grad-CAM is a technique utilized for understanding and explaining the decisions made by a CNN in image classification. By analyzing the gradients in the last convolutional layer of the CNN, this technique determines the importance of each region of the image. In other words, it indicates

the areas of focus of the network during its classification process. Grad-CAM serves as a crucial tool in explaining the workings of the CNN and verifies that the network has successfully learned the desired patterns. Figure 9.8b shows an example of an attention map obtained using Grad-CAM on the VGG19 architecture. This attention map highlights the region of the image used to predict the condition of the tool. In particular, the focused area is located on the tool's wear zone, enabling the network to correctly categorise this image. The area of focus can differ based on the architecture employed. In this study, all accurately classified images have a concentrated attention located on the area of wear. This proves that the method have identified that the distinction between a new tool and a worn tool is attributed to the amount of flank wear.



(a) Attention map of a correctly classified image. The highlighted area corresponds to the tool wear.



(b) Attention map of an incorrectly classified image. A contrast variation shifts the attention away from the wear zone.

Figure 9.8: Grad-CAM attention maps from VGG19 classification. The left image shows correct focus on the wear zone, while the right image illustrates misclassification due to altered contrast.

Figure 9.8a shows the VGG19 attention map for the classification of the same image as Figure 9.8b but this image has undergone a change in contrast. In this case, the network wrongly classified the image as representing a new tool. This change in contrast appears to have misled the network, causing it to focus on an incorrect part of the tool for its prediction. The highlighted area clearly indicates that the network has focused on a part of the tool that shows minimal signs of wear. The same conclusions can be drawn for all the errors made by the different approaches. In general, misclassification is due to an inability to detect the area of wear on the image. This is either because the image has been modified or because the original image contains features that make it difficult for the AI to detect.

9.5 Conclusions

This chapter explores the use of transfer learning to classify the state of cutting tools based on their images. Three architectures pre-trained on the ImageNet database are utilized for this purpose: VGG19, EfficientNetV2, and Vision Transformers. These models are employed to categorize the state of the tools into three classes: new, moderate, or worn. The images used for training the models are derived from the experimental turning tests (see Chapter 6). The original dataset is augmented and balanced through data augmentation and oversampling. This is done to assess the robustness of the different architectures against variations in brightness, contrast, and image orientation that may occur in an industrial environment.

Among the approaches, VGG19 yielded the best results with an accuracy of 94 %. It was closely followed by EfficientNetV2 and Vision Transformers (ViT), both achieving an accuracy of 87 %. All models demonstrated robustness against image modifications, showcasing the strength of transfer learning.

A detailed analysis of the errors made by the networks revealed that even though EfficientNetV2 and ViT have the same overall accuracy, the errors committed by EfficientNetV2 occur at the transition between classes and it makes fewer errors in the last class. Consequently, the errors made by EfficientNetV2 have less negative impact on tool replacement compared to those made by ViT.

In addition to the accuracy of each technique, the Grad-CAM method provides additional information on the ability of the networks to detect the area of flank wear. The analysis reveals that the networks successfully located the region of interest in the images, which corresponds to the flank wear region. In addition, this examination highlights the reasons for the misclassification of certain images, illustrating the challenges faced by the networks in recognising the area of wear.

Chapter 10

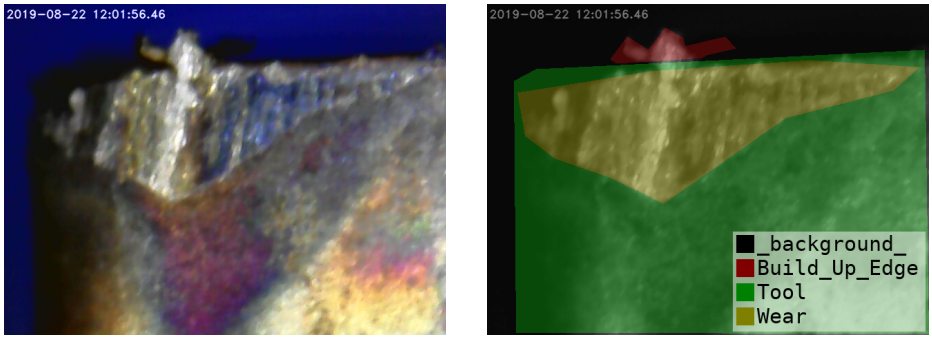
Monitoring the Cutting Tool Degradation from Tool Images in Turning

Direct Monitoring with Regression

10.1 Introduction

In Chapter 9, the images from the database were used to classify the tool's condition into three distinct categories. In the current chapter, the focus is on direct monitoring with regression. For this approach, the aims is to directly identify the wear zone on the image in order to measure it and obtain the value of VB. The detection of the wear is called "image segmentation". Figure 10.1 illustrates the segmentation concept. The objective of segmentation is to start with an image of the tool as the one on Figure 10.1a and the objective of the model is to provide its segmentation map as on Figure 10.1b.

In the context, of this chapter, the segmentation task of the model will focus on identifying: the cutting tool, the background, the wear and any potential build up edge (see Figure 10.1b). The task of segmenting the wear zone has been addressed in the literature. For instance, Bergs et al. developed segmentation methods to detect wear on various types of tools, either by using multiple U-Net models, each tailored to a specific tool, or a single model capable of identifying wear across different tool types [167]. Similarly, Li et al. proposed a segmentation approach that targets both tool wear and the machined surface [168]. While these methods focus on image segmentation, the originality of the proposed approach lies in its ability to perform multi-class segmentation of the different parts of the tool wear. In this chapter, the models used to perform the segmentation are U-Net models (Figure 10.2). In order to build this U-Net model adapted to this multi-class segmentation task, this chapter is structured as follows:



(a) Image of the tool given as input to the model **(b)** Corresponding segmented image (manually obtained)

Figure 10.1: Segmentation : Tool wear and its degradation detected manually.

- Section 10.2 explains the preprocessing steps applied to the images in the database.
- Section 10.3 motivates and describes the use of U-Net models to perform the segmentation task.
- Section 10.4 describes the results of the models and its technical limitations.
- Section 10.5 concludes this chapter.

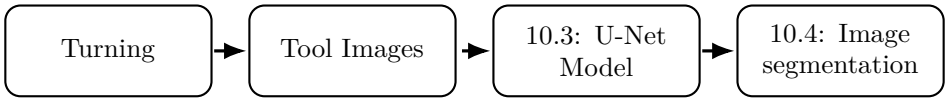


Figure 10.2: Summary of the approach developed in this chapter.

10.2 Data Preparation for Image Segmentation

The data preparation process used in this chapter follows the same approach as described in Chapter 9, specifically in Section 9.2. The turning dataset consists of only 192 images, which is relatively limited given the segmentation objectives. To address this limitation, just as in Chapter 9, image augmentation techniques is applied to artificially expand the dataset and simulate industrial alteration of the images.

Image segmentation is essentially a classification task where each pixel in an image is assigned to a specific class. In Figure 10.1, the classes include: background, build-up edge, tool, and wear. These classes were chosen to classify everything that can be seen in the images from the database. The background class corresponds to the background and everything that is not the tool. The build-up edge class shows a part of the tool where the cutting material has remained stuck to it. The tool class encompasses the entire tool. And the wear class includes all the worn areas of

the tool. The goal of the model is to determine, for each pixel in the input image, which class it belongs to. To train such models, each input image must be paired with its corresponding segmented image, often referred to as a segmentation map (Figure 10.3). These segmentation maps are actually matrices. While a standard image is represented as a matrix of shape [x-pixels, y-pixels, 3], where the three channels correspond to red, green, and blue colour intensities, a segmentation map follows a similar structure but with a different interpretation. In a segmentation map, each channel represents a class rather than a colour. Thus, the shape of a segmentation map is [x-pixels, y-pixels, number of classes]. A pixel is set to 1 in a given channel if it belongs to the class represented by that channel. This type of class encoding is known as one-hot encoding (Figure 10.3). Alternatively, it is possible to create a segmentation map with the shape [x-pixels, y-pixels, 1], where each pixel value is no longer binary (0 or 1), but instead corresponds to the class index. For example, a pixel with a value of 2 indicates that it belongs to class 2. In this chapter, the one-hot encoding approach has been selected. Therefore segmented images have 4 channels : background, build up edge, tool and wear. These segmentation maps are created using dedicated annotation software. Several tools are available, but for the purposes of this thesis, LabelMe has been fully integrated into the development workflow [169]. This open-source software allows for the creation of segmentation maps from images.

The image size of the cutting tool is 460 by 640 pixels. To evaluate the models' performance while keeping their size and memory usage low, making them suitable for use on a standard computer, the input images are resized to 256×256 pixels. To train the model, 75 % of the dataset is allocated to the training set. From the remaining 25 %, 70 % is used for testing and 30 % for validation. These proportions differ from those used in previous chapters, where the dataset was split based on complete degradation trajectories (Chapters 7 and 8) or balanced class distributions (Chapter 9). In the context of image segmentation, no such constraints apply, allowing for a standard split of 75 % for training and 25 % for testing.

10.3 Model for Image Segmentation

Multiple methods to perform image segmentation exist but most of them are based on an encoder-decoder architecture [170] (Figure 10.4). The encoder is composed of convolutional and pooling layers responsible for extracting meaningful features from the image, much like the convolution and pooling part of a traditional classification network (such as VGG19 for example). It progressively reduces the spatial dimensions while capturing high-level semantic information. The resulting vector or map is called the latent space. The decoder, on the other hand, takes the encoded features from the latent space and apply deconvolution (opposite of convolution) and unpooling layer (opposite of pooling) aiming to classify each pixel in the image, thereby generating a segmentation map [170].

Image segmentation is a topic particularly important for medical image segmentation. In this domain, numerous encoder-decoder architectures have been proposed. One of the most widely recognized models is U-Net [171], named for its distinctive U-shaped design. The U-Net architecture consists of two main components (Figure 10.4): a contracting path that captures contextual information

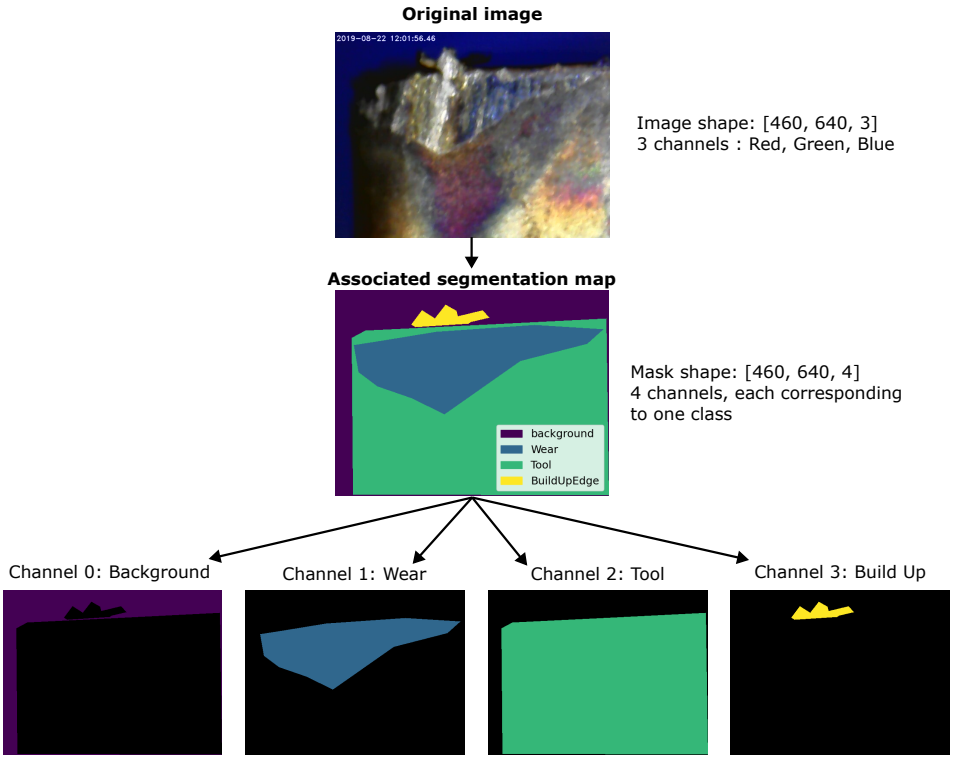


Figure 10.3: Image segmentation with one hot encoding principle. The segmentation map is divided into multiple channels, each corresponding to a given class. In the channel decomposition of the segmentation map, colours correspond to value of 1 while black correspond to value of 0.

(encoder), and a symmetric expanding path that allows for precise localization (decoder). The contracting path resembles a Convolutional Neural Network (CNN), using 3×3 convolutions to extract features. The expanding path performs up-sampling through deconvolutions, which increase spatial resolution while reducing the number of feature maps. To preserve important pattern information, feature maps from the contracting path are copied and merged into the expanding path. Finally, a 1×1 convolution is applied to produce a segmentation map that assigns a category to each pixel in the input image [170, 171]. Various extension of U-Net have been developed. In this chapter, 3 types of U-Net will be compared:

- Classic U-Net, originally designed for biomedical image segmentation [171].
- U-Net++, which introduces nested and dense skip connections for better feature fusion. By redesigning the skip pathways with a series of convolutional layers, U-Net++ reduces the semantic gap between encoder and decoder features, leading to more precise segmentation, especially in complex scenarios [172].
- Double U-Net (DU-Net), which stacks two U-Net architectures to refine

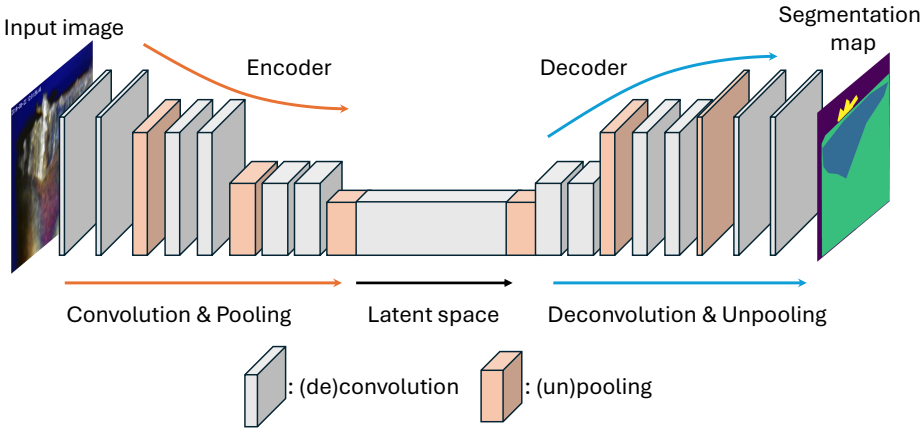


Figure 10.4: Schematic representation of an encoder-decoder architecture. The encoder part use convolution and pooling layer to reduce the dimension into a latent space. The decoder uses this latent space with deconvolution and unpooling layer to reconstruct the output segmentation map.

segmentation results . The first U-Net generates an initial segmentation map, which is then used by the second U-Net—along with the original image—to produce a more accurate and detailed output. This dual-stage approach allows DU-Net to correct errors and improve boundary detection [173].

Table 10.1 shows the number of parameters for each model for an image input size of 256 by 256 pixels. Since the U-Net model is the baseline for the other ones, it is observed that the U-Net model have the lowest amount of parameters.

Table 10.1: Number of parameters of the different models.

	U-Net	**U-Net++**	**DU-Net**
Nb of parameters	7,961,634	9,279,330	18,836,771

All models are trained on 100 epochs with a batch size of 16. They are all trained in the same condition and with the same training dataset but the image modification are randomly generated.

10.4 Results

To evaluate the model performance, the indicator Intersection over Union (IoU) is used. This is a common metric used in image segmentation. It measures the overlap between the predicted segmentation and the real segmentation. Specifically, it is computed as the ratio of the area of overlap between the predicted and true

regions to the area of their union. A higher IoU indicates better model performance, with a value of 100 % representing perfect overlap. This indicator is computed as (Eq. (10.1)):

$$\text{IoU} = \frac{|\text{Prediction} \cap \text{True}|}{|\text{Prediction} \cup \text{True}|} \tag{10.1}$$

To compare the performance of different models in image segmentation tasks on the turning database, Table 10.2 presents the IoU scores obtained under various image modifications. The standard deviation reported in the table is computed over 10 independent runs for each model. Results are grouped by type of modification. Across all models, image rotation consistently yields the lowest scores, indicating that rotation poses a greater challenge for accurate wear detection. For other types of modifications, the performance remains relatively stable, with models maintaining similar accuracy levels within the bounds of the standard deviation. Overall, the three models demonstrate comparable performance, with an average IoU around 75 %. Despite these similarities, it is worth noting that U-Net++ achieves the highest average score and exhibits the lowest variability in performance.

Table 10.2: Comparison of IoU accuracy across different architectures under various image modifications. Each value represents the mean IoU over 10 independent runs of the same model.

Image Modification	**U-Net**	**U-Net ++**	**DU-Net**
IoU per Modification			
No modification	$76.9 \pm 2.7\,\%$	$78.0 \pm 1.6\,\%$	$78.1 \pm 3.0\%$
Flipped	$76.7 \pm 2.9\,\%$	$78.1 \pm 1.6\%$	$78.1 \pm 2.7\,\%$
Rotated	$62.2 \pm 3.5\,\%$	$64.7 \pm 2.9\%$	$63.4 \pm 3.4\,\%$
Brightness +50 %	$75.4 \pm 3.1\,\%$	$77.2 \pm 1.9\%$	$75.6 \pm 3.2\,\%$
Brightness -50 %	$75.8 \pm 4.2\,\%$	$77.3 \pm 1.4\,\%$	$77.5 \pm 2.4\%$
Contrast +50 %	$74.7 \pm 0.7\,\%$	$76.9 \pm 1.7\%$	$75.8 \pm 3.0\,\%$
Contrast -50 %	$75.8 \pm 2.3\%$	$75.5 \pm 2.7\,\%$	$74.9 \pm 2.9\,\%$
Overall Score			
IoU	$73.9 \pm 5.9\,\%$	$75.4 \pm 4.9\%$	$74.8 \pm 5.6\,\%$

To visualize the results obtained, Figure 10.5 shows an example of segmentation map obtained by the U-Net ++ model. The Figure illustrates the input image provided to the model, the expected segmentation map (unknown to the model), and the predicted map. This example showcase that even though the dataset is small, the model is able to reliably detect the wear zone, the tool, and any potential built-up edge on the tool. Interestingly, the models demonstrate a high level of robustness, sometimes producing segmentations of higher quality than the manual annotations used for training. This is illustrated in the figure, where the wear zone is detected more precisely than in the manually annotated map. A similar observation can be made for the detection of the built-up edge, which is also more accurately identified by the model than by manual labelling. It is important to note that manual annotation is a complex and often repetitive task, which can lead to

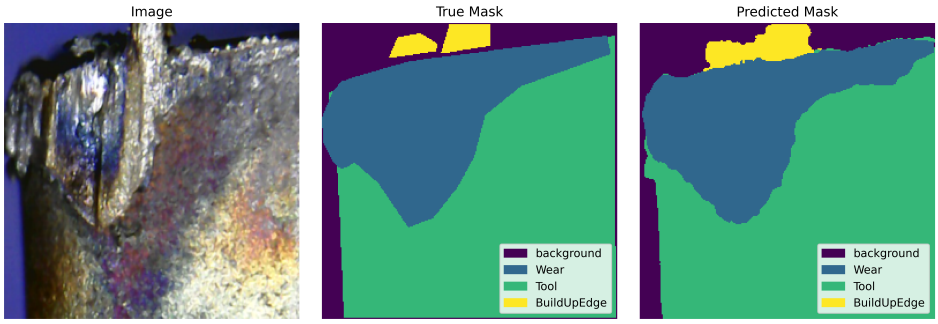


Figure 10.5: Example of prediction obtained by the U-Net++ model. From left to right: Image given to the model, expected segmentation map and map produced by the model.

simplifications or inconsistencies in the segmentation maps used for training. This highlights a key strength of AI-based approaches over human annotation: factors such as subjectivity, fatigue, and the repetitive nature of the task can negatively impact the reliability of manual labelling. In contrast, AI models can maintain consistent and precise segmentation performance, even in challenging scenarios. This is especially relevant in the context of Industry 5.0, where machines are not only designed to automate repetitive tasks but also to enhance worker well-being [174].

Using segmentation maps, it is possible to isolate the area of tool wear. From this area, it would theoretically be possible to measure the degree of wear if the aspect ratio between the image and the actual dimensions is known. (i.e., how many microns correspond to one pixel). Unfortunately, the dataset lacks uniformity in this regard. Another limitation concerns the identification of the evaluation zone used to measure tool wear. As mentioned in Chapter 2, wear should be assessed beyond the corner radius. However, most images do not allow for a clear identification of the tool edge, making it difficult to accurately locate the corner radius. While manual wear measurement remains feasible, it must be performed on a case-by-case basis. As a result, this final measurement step cannot be consistently applied across the entire dataset, preventing fair comparison of results. This represents a limitation of the current dataset. A complete description of this measurement process will be provided in the corresponding chapter, where it is applied to the milling dataset. This milling dataset addresses the limitations mentioned above and enables a more reliable and consistent evaluation of tool wear (see Chapter 18).

10.5 Conclusions

This chapter presents an approach for identifying the tool, wear zones, built-up edge, and background in images of turning cutting tools. The concept of image segmentation is introduced to describe the preprocessing required for building the training dataset. This preprocessing requires manual segmentation steps that can be time-consuming and challenging, but it enables the creation of segmentation maps on which encoder-decoder models can be trained to distinguish the different

regions of the tool. Three encoder-decoder architectures are compared: U-Net, U-Net++, and DU-Net, evolutions of the original U-Net model.

The results assess the robustness of these models against image alterations representative of industrial conditions. All three models achieve similar performance, with U-Net++ showing the best overall results obtaining an overall IoU score of 75.4±4.9%. The findings demonstrate that these models are highly effective in identifying wear zones. A discussion on their performance compared to manual annotation highlights the superior robustness of AI models in handling this complex and repetitive task.

Finally, due to the limited size and heterogeneity of the dataset, the conclusions remain partial, as a complete evaluation of wear measurement is not feasible. A more thorough analysis will be presented in the chapter dedicated to direct regression-based monitoring in milling, which addresses the limitations of the current turning dataset (see Chapter 18).

Chapter 11

General Conclusions on the Development of Tool Condition Monitoring Strategies in Turning

This Part II presented the development of TCM applications built throughout the thesis, based on a dataset applied to turning operations. Based on the observations from the state of the art, several key points were addressed in the corresponding chapters:

- Chapter 7 introduced the methods used to monitor tool wear degradation from measured cutting force signals. A neural network-based approach was described and complemented with techniques to estimate confidence intervals around the network's predictions, as well as an explainability method. The bootstrap method was used to estimate reliable confidence intervals around the monitored points. To enhance explainability, an approach based on explainable boosting machines was developed. This method achieved performance comparable to neural networks, with an R^2 of 90 % versus 92 % for the neural network-based method with the advantage of explainability
- Chapter 8 presented a novel approach for forecasting the future state of the tool based on past cutting force signals. This method was developed to predict the tool condition up to 3 minutes ahead, corresponding to the production time of a part in the experimental campaign. The recurrent neural network-based approach showed excellent results, with precise monitoring down to a few tens of microns, even for future state estimation.
- Chapter 9 demonstrated how AI models obtained through transfer learning can be used to classify tool condition from images. The architectures used included VGG19, EfficientNetV2, and a Vision Transformer. Results showed that these models could detect the wear zone and classify the tool condition with an

average accuracy close to 90 % with VGG19 showing greater robustness to image alterations. These findings highlight the relevance of such methods, particularly the use of transfer learning to address the challenge of limited image datasets.

- Chapter 10 explored how images can be used to estimate tool condition through automatic identification of the wear zone (segmentation). The models employed were based on an encoder-decoder approach, which proved highly effective in identifying the wear zone with segmentation more accurate than manual annotation. Due to limitations in dataset size and homogeneity, no definitive conclusions could be drawn regarding the accuracy of VB measurement.

This part of the document has therefore applied many TCM approaches to the case of turning, while also proposing novel methods in the literature. It is worth emphasizing that applying the full range of TCM techniques to a single dataset represents a unique contribution, as such a comprehensive comparison has not previously been reported in the literature. However, given the relatively small and turning-specific dataset, the next part of the document will focus on building a larger milling dataset and addressing the limitations observed in the turning dataset. All the methods developed for turning will be adapted to milling.

Part III

Application of Tool Condition Monitoring Methods in Milling Operation

Chapter 12

Introduction to the Development of Tool Condition Monitoring Methods in Milling

In Part II of this thesis, various Tool Condition Monitoring (TCM) strategies were applied to the case of turning. In Part III, similar strategies are now applied to the case of milling. While a turning dataset was already available within the lab, no such dataset existed for milling. Therefore, the initial chapters of this part focus on building a dedicated milling dataset, while addressing the limitations observed in the turning dataset. As a reminder, these limitations include: a small dataset size, the use of only one sensor, varied cutting conditions but with limited variation, and tool images captured under suboptimal conditions. The chapters are organized as follows:

- Chapter 13 presents the experimental framework used to collect milling data. It details the experimental setup and the data acquisition procedure, with the objective of creating a more complete and homogeneous dataset than the one used for turning.

- Chapter 14 presents the data collected during the experimental campaign. It analyses how the tool and sensor signals evolve over time and with tool wear. This chapter also defines the indicators used to characterize the signals in both time and frequency domains, and explains how the data can be processed and segmented to train the various methods used throughout this part. In particular, feature selection is described to compare the performance of individual sensors and to evaluate the benefits of data fusion. This categorization will be applied across all indirect approaches, enabling a comparison of model performance based on the sensors used either individually or in combination.

With the dataset developed in these two chapters, the TCM methods previously applied to turning are adapted and applied to milling, as follows:

- Chapter 15 applies the indirect monitoring regression methods introduced in Chapter 7. It evaluates the performance of these methods and explores the use of confidence intervals and explainability techniques.
- Chapter 16 focuses on forecasting tool wear using sensor signals. As in Chapter 8, the goal is to use recurrent neural networks to predict the evolution of tool wear over time.
- Chapter 17 applies classification methods based on tool images. The same transfer learning approaches developed in Chapter 9 are used to classify the tool state on 3 states.
- Chapter 18 implements image segmentation techniques to detect wear zones in tool images from the milling dataset. This chapter complements Chapter 10, which could not fully assess model performance due to limitations in the turning dataset.
- Chapter 19 concludes this part dedicated to the application of TCM methods to the milling dataset.

Chapter 13

Framework for Creating a Milling Database for Tool Condition Monitoring

13.1 Introduction

In Part II, the developed methods were applied to turning operations. In Part III, these approaches are now extended to milling. This chapter therefore describes the conditions under which this new database was acquired.

The database presented in this chapter is designed to be a continuation of the turning dataset. However, the turning database had several limitations that will be addressed and corrected in the milling dataset (see Chapter 12). Although these limitations did not prevent the use of AI on this dataset, they highlighted several areas for improvement. The objectives of the milling experimental campaign are therefore multiple:

- Collect sufficient data to effectively train artificial intelligence techniques for monitoring cutting tool degradation. One limitation of the turning dataset was that cutting signals were recorded only during the last 20 seconds of each machining operation. This prevented continuous tracking of tool wear throughout its lifespan and limited the number of observations to 190. For the milling dataset, the entire tool life will be recorded, allowing signal tracking from beginning to end. This will significantly increase the amount of data compared to the turning approach. Since the turning dataset still provided enough data to train models, it is expected that capturing the full degradation process will retain or improve this effectiveness.

- Compare the correlation between multiple features from different sensor types and cutting tool wear during the machining process. In the turning dataset,

only cutting forces were measured. For milling, cutting forces, accelerations, acoustic signals, and machine power will be recorded, thereby increasing the volume and diversity of data and capturing various physical phenomena. It will also allow for a performance assessment of each individual sensor in monitoring and forecasting tool degradation, as well as an evaluation of sensor fusion strategies.

- Evaluate the impact of varying cutting conditions on tool wear and the measured signals. This includes introducing sufficient variation in cutting parameters to observe meaningful differences across conditions. This approach is inspired by the methodology used in the turning experiments.

- Ensure that all images of tools are captured from a consistent angle to facilitate image analysis. This limitation was identified in chapters where tool images were analyzed to determine wear states. Particular attention to image acquisition will be made to ensure that all images are taken under similar conditions, enabling automated image processing.

This chapter describes the equipment and procedure followed to acquire the data for the milling dataset. Therefore, it is divided as follow:

- Section 13.2 describes the experimental framework that was used during the experimental campaign. More specifically:
  - Section 13.2.1 describes the equipment that was used during the tests.
  - Section 13.2.2 describes the sensors used during the tests.
  - Section 13.2.3 describes the machining procedure followed during the tests.
- Section 13.3 describes all the cutting conditions of the database.
- Section 13.4 describes how all the data are organized to be easily integrated in automatic features extraction techniques.
- Section 13.5 presents the conclusions of this chapter.

13.2 Description of the Experimental Framework

A schematic representation of the equipment and sensors used during the experimental milling campaign is presented Figure 13.1. This section provides a detailed description of all the components shown in the figure. Section 13.2.1 describes the equipment used during the tests. Section 13.2.2 presents the sensors employed, distinguishing between on-line sensors (used during machining) and off-line sensors (used after machining). Finally, once all components of the test setup have been described, Section 13.2.3 outlines the procedure followed during the tests.

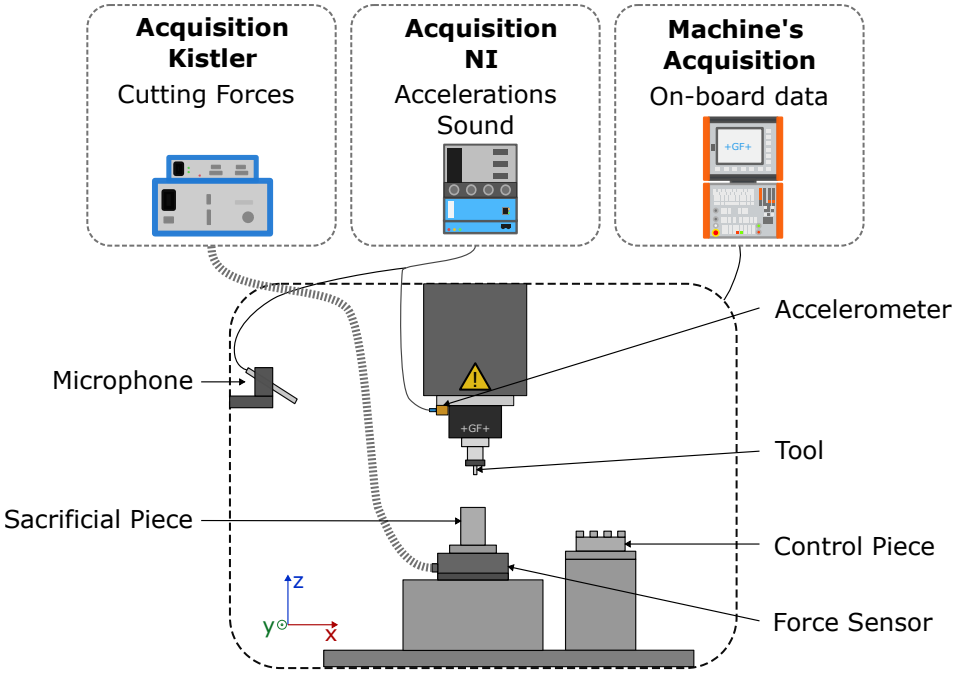


Figure 13.1: Overview of the experimental setup. The three acquisition means are shown along with their respective measured phenomena and the position of the components within the machine.

13.2.1 Equipment Used During the Tests

The most important equipment of the experimental campaign is the CNC machine. It is a GF MIKRON VCE 600 Pro, a 3-axis CNC milling machine installed in the lab (see Figure 13.2). The machine is controlled by a Heidenhain TNC 620 CNC system. The VCE 600 pro features a cross-table design: the x and y axis movements are executed on a table holding the workpiece, while the z axis movements are performed by the tool. The spindle power is 14.75 kW at the max spindle speed which is 16 kRPM.

As illustrated in Figure 13.1, two workpieces are mounted inside the machine: one sacrificial and one control workpiece. Figure 13.3 shows the mounting configuration of these two components on the machine's table. The sacrificial workpiece has a parallelepiped geometry measuring 60 mm × 60 mm × 100 mm (length, width, height). The control workpiece is also parallelepiped-shaped, with dimensions of 35 mm × 90 mm × 60 mm (length, width, height), on which grooves have been machined to create smaller parallelepipeds of 15 mm × 25 mm × 10 mm. Each of these smaller blocks is machined at a specific stage of the tool's life, allowing the surface quality to be recorded at different levels of tool wear (see Section 13.2.3).

Both workpieces are made of C45 steel and undergo a homogenizing heat treatment, similar to the one used for the turning database (see Chapter 6). This treatment ensures uniform hardness throughout the entire volume of each workpiece, eliminating the risk of localized hard spots either internally or on the surface.



Figure 13.2: Machine used during for the experimental milling campaign: GF MIKRON VCE 600 Pro.

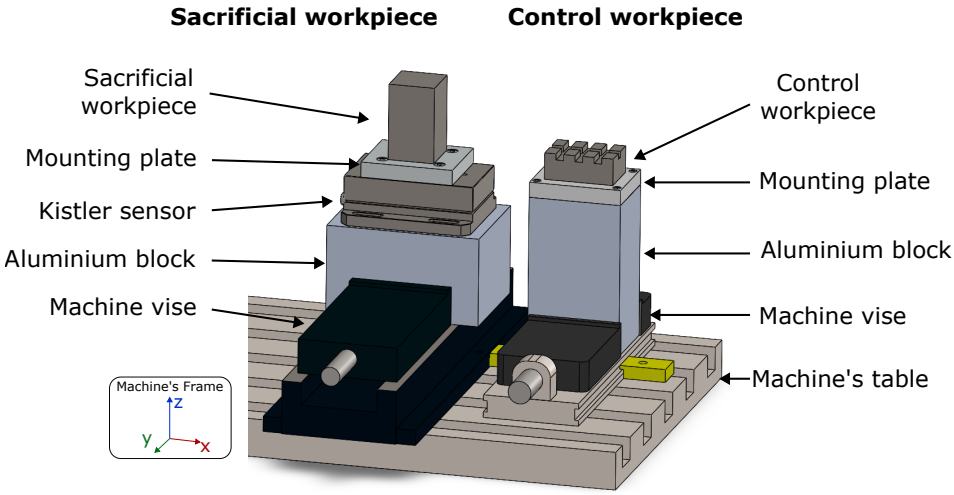


Figure 13.3: Sacrificial and control workpiece mounting assembly.

The objective of this heat treatment is to achieve a hardness between 179 and 203 HV_{30}, consistent with the turning database. Prior to the experimental phase, the Vickers hardness of four pieces (two control pieces and two sacrificial pieces) was measured at several points on each piece. The measured hardness are homogeneous and their range varies from 179 to 187 HV_{30}.

To machine this workpieces, the selected cutting tool is an uncoated solid carbide end mill (Figure 13.4). The tool is mounted in an ER 25 collet chuck, Type ADB, SA 40 A = 70. As with the turning database, the selection of this tool was made to minimize experimental time and reduce material waste. This objective

is reflected in the tool's specifications, which are listed in Table 13.1. The end mill is uncoated to intentionally shorten its lifespan. Additionally, the machining is performed dry, even though the tool is originally recommended for use with cutting fluid. The cutting conditions were also chosen to accelerate tool wear while maintaining stable machining parameters (see Section 13.3).



Figure 13.4: Selected cutting tool to perform the experimental milling campaign [62].

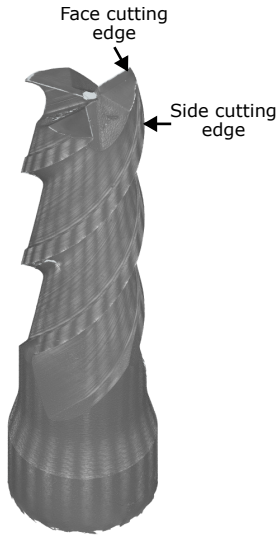
Table 13.1: Specifications of the cutting tool.

Manufacturer	Holex
Manufacturer reference	2023075
Standard for geometry	DIN6527
Nb of teeth	3
Diameter	5 mm
Helix angle	45°
Cutting length	13 mm

To ensure that the cutting tools exhibit consistent geometries and that geometric variability does not significantly affect the experimental results, the edge sharpness radii of new cutting tools were measured. The measurements were performed at the Mondragon Unibertsitatea using an Alicona Infinite Focus G6 equipped with a 150 WD11 objective (50× magnification) (see Figure 13.5a). Figure 13.5b shows the 3D reconstruction of the the cutting tool obtained with the Alicona using the 1900 WD30 lens (4× magnification). Two sharpness radii were characterized on each tool, corresponding to the face cutting edge and the side cutting edge (see Figure 13.6a). Figure 13.6b presents the sharpness radii measurements obtained for four new tools. The dispersion associated with each sharpness radius reflects the variability observed across the three measurements performed on each tool (one measurement per tooth). The mean sharpness radius and its corresponding dispersion are also reported in the figure. These results indicate that the edge sharpness radii exhibits only minor variation between tools, with a dispersion below 1 µm for both the face and side cutting edges. This low variability demonstrates a high level of repeatability in the manufacturing process of the cutting tools, implying that this parameter can be considered constant during the experiments.



(a) Alicona IFG6 - Optical surface roughness measuring device installed at Mondragon Unibertsitatea.



(b) 3D reconstruction of the cutting obtained with the Alicona Infinite Focus G6 showing the face cutting and side cutting edges.

Figure 13.5: Alicona Infinite Focus G6 and the 3D reconstruction of a new cutting tool.

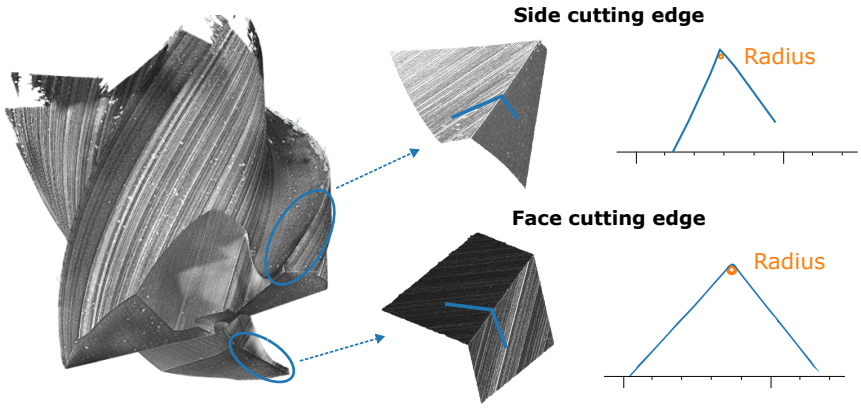
13.2.2 On-Line and Off-Line Sensors

On-Line Sensors

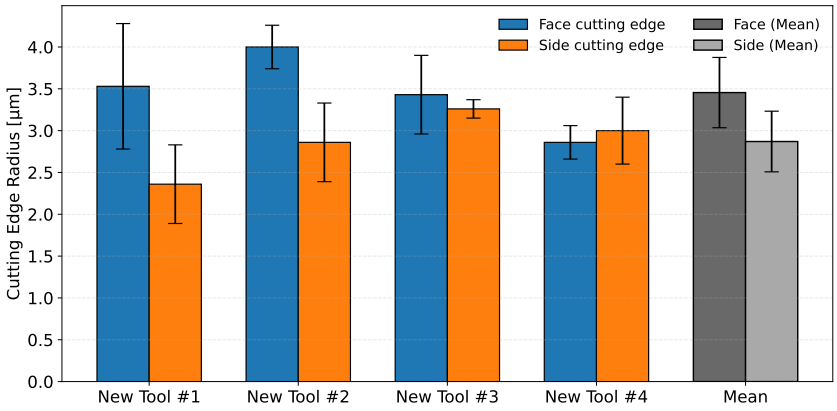
During machining, multiple sensors are measuring the process: a cutting force sensor, an accelerometer, a microphone and the power meter of the machine.

The cutting forces are measured with a Kistler 9257B force sensor. Figure 13.7 shows a photography of this sensor during machining with the sensor and machine's frame. The complete mounting assembly of this sensor is visible in Figure 13.3. The piezoelectric cells of this sensor allow for the measurement of cutting forces and moments in three spatial directions. In the x and y planes of this sensor, cutting forces can be measured up to $\pm$ 5 kN, and in the z direction up to $\pm$ 10 kN at a maximal sampling frequency of $\pm$ 1 MHz (for a single channel). The natural frequency of the sensor is around 2.3 kHz for the x and y planes and 3.5 kHz for the z direction. This sensor has its own dedicated acquisition line. It measures the signals during the machining of the sacrificial workpiece at an acquisition frequency of 20 kHz. The duration of the acquisition is adjusted depending on the cutting conditions but in average the acquisition takes around 2 to 3 minutes.

Mounted on the spindle, the Dytran 3263A2 triaxial accelerometer (Figure 13.8) measures its vibrations. This location was chosen because it is the closest flat surface suitable for sensor placement. The accelerometer has a sensitivity of 100 mV g^{-1} across all three spatial axes, within a frequency range of 2 to 5000 Hz, and can measure accelerations up to $\pm$ 50 g. Its resonance frequency exceeds 40 kHz. Data from this sensor is acquired using National Instruments (NI) acquisition cards, which are shared with the microphone. As a result, both sensors operate at the



(a) Cutting edge radii for the side and face cutting edges.



(b) Cutting edge radius of 4 new tools measured with the Alicona Infinite Focus G6 (see Figure 13.5a).

Figure 13.6: Geometry of the cutting edge radii for the cutting tool.

same sampling rate, set to 51.2 kHz to meet the requirements of the microphone. Similar to the cutting force sensor, the acquisition duration is adjusted based on the cutting conditions and the length of the machining operation.

The sound generated during the cutting process is captured using a microphone with a sensitivity of 8.2 mV Pa^{-1}. In the experimental setup, the microphone is positioned above the left door of the machine, at a safe distance to avoid any risk of chip projection (Figure 13.9). It is oriented toward the centre of the machine to directly capture the sound of the machining operation. The microphone is connected to a NI acquisition card, operating at a sampling rate of 51.2 kHz, which is the maximum supported by the card. This configuration ensures compliance with the Nyquist-Shannon theorem, allowing accurate sound measurement within the audible range of 20 Hz to 20 kHz [175].

During machining, it is also possible to collect data directly from the CNC machine. Among the available signals, the selected ones include the power con-

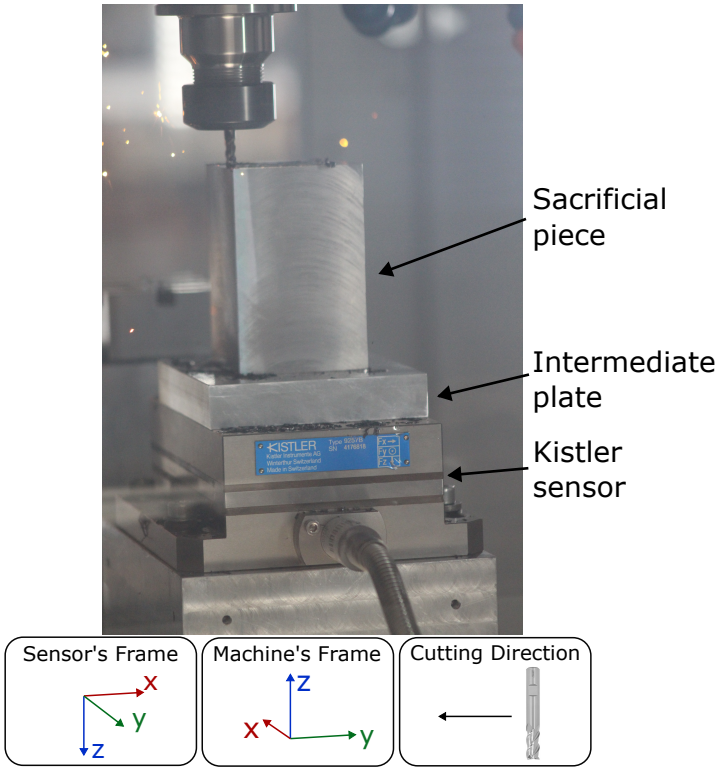


Figure 13.7: Kistler 9257B and its assembly with the sacrificial piece during machining. The sensor frame differs from the one of the machine.

sumption of all machine axes (x, y, z, and spindle), measured accelerations along each axis, nominal (commanded) accelerations, axis velocities in all directions, and the spindle coordinates. Although the full specifications of the machine's internal sensors are not entirely known, data acquisition is performed at a frequency of 10 kHz throughout the machining process.

Off-Line Sensors

Offline sensors are also used to capture images and measure both cutting tool wear and the surface finish of the machined parts.

To photograph the cutting tools, a Dino-Lite AM7013MZT microscope (Figure 13.10) is used in combination with its dedicated software, DinoWare. For each tooth, two images are taken: one showing flank wear and another showing tip wear. These images are captured at different magnifications due to the varying size and location of the wear areas. Flank wear is photographed at 50× magnification (Figure 13.11a), while tip wear is captured at 150× (Figure 13.11b). Since the tool has three teeth, a total of six images are taken during each inspection. To identify the first tooth, a mark is made on the tool using a permanent marker. This mark indicates tooth number 1. The remaining teeth are then numbered by viewing the

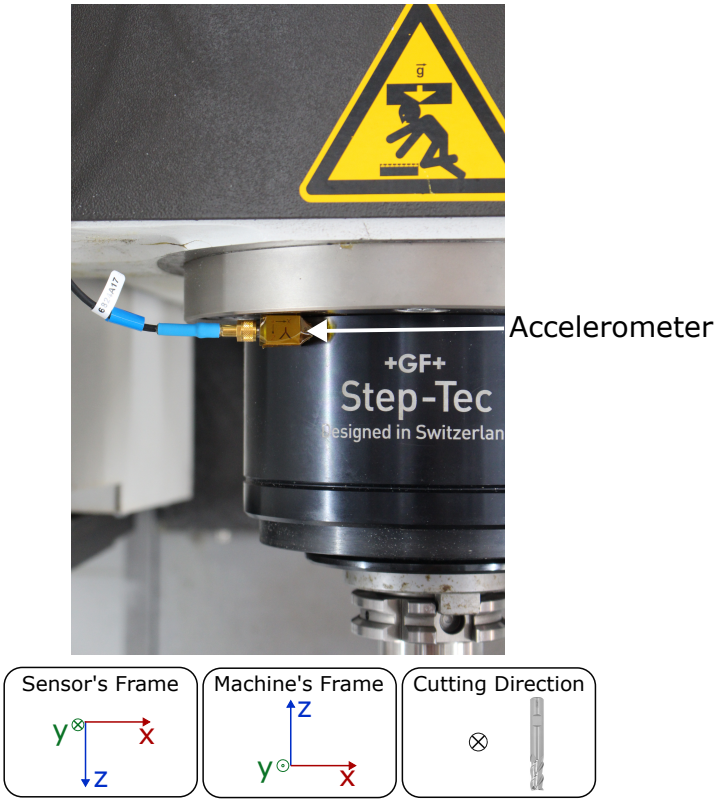


Figure 13.8: Dytran 3263A2 fixed on the closest flat surface on the machine. The sensor frame differs from the one of the machine.



Figure 13.9: Microphone oriented toward the centre of the machine.

tool from the end and counting in a counter-clockwise direction. As mentioned in the introduction to this chapter, particular attention was paid to ensuring that all images in the database were acquired consistently. As shown in Figure 13.10, the tool was placed on a 3D-printed support designed to align it with the microscope.

A black background was also used in all images of flank wear (see Figure 13.11a) to clearly highlight the tool against the background. For tip images, however, using a black background was not possible because the background corresponds to the chuck (see Figure 13.11b). For all images, the exposure histogram was adjusted to avoid overexposed pixels and ensure the image remained usable. Focus was also carefully set to ensure that the sharpest area corresponded to the worn part of the tool. This last point was particularly challenging due to the geometry of the tool and the limited depth of field, which inevitably causes some parts of the image to be slightly out of focus. It should be noted that due to these challenges, capturing images of the tool tip during testing was more difficult. This is mainly because the focus area becomes narrower at a magnification of 150×. As a result, the images of the tool tip are generally of lower quality compared to those of the flank face.

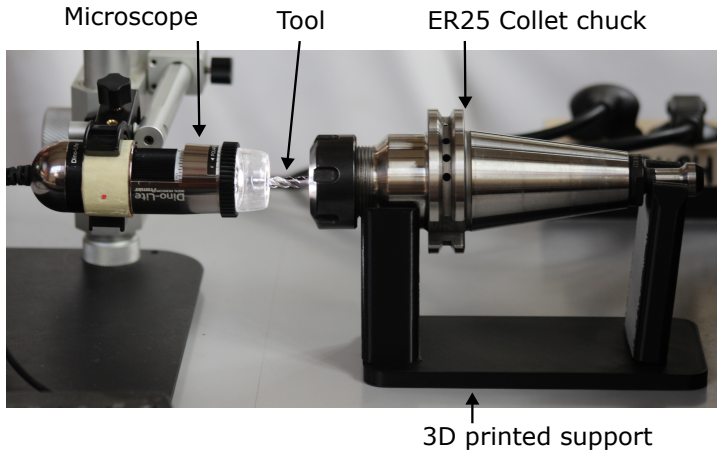


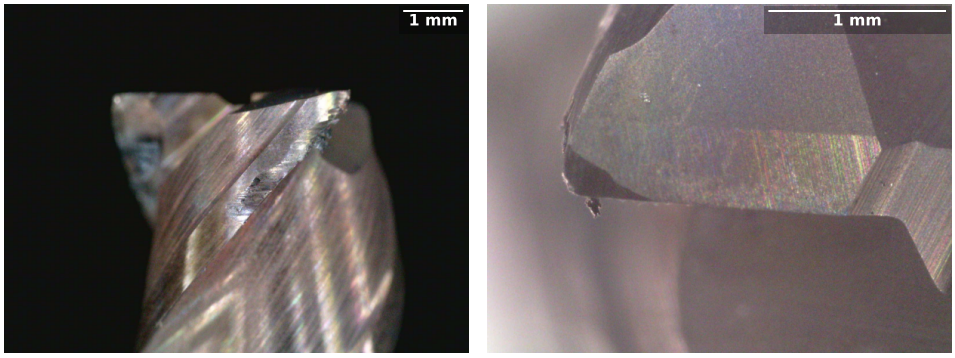
Figure 13.10: Microscope used to take picture of the cutting tool. In this configuration, an image of the tip of the tool is acquired.

At the end of each test, the control piece is retrieved, and the surface roughness of the areas machined by the tool tip is measured using a Keyence VR-6000 optical profilometer (Figure 13.12). This profilometer enables both surface imaging and detailed surface analysis. Among the various analyses possibilities with this device, only the surface roughness of the control piece will be considered in this study.

13.2.3 Machining and Data Acquisition procedure

The machining procedure is divided into three parts. Figure 13.13 presents an overview of the different steps.

The first part begins with the tool already mounted in the machine and ready to cut. During this phase, the tool is used to machine the sacrificial workpiece. Figure 13.14 illustrates the machining path. Since the tool operates in opposite milling, the trajectory starts at the red point shown in the figure. The tool then performs a pass through the material, with the depth, width of cut and feed rate determined by the parameters a_p, a_e and v_f, which are defined by the cutting conditions. The depth and width of cut are set to 2 mm for all the



(a) Image of the flank wear acquired with a 50× magnification

(b) Image of the tip wear acquired with a 150× magnification

Figure 13.11: Example of image acquired with the microscope. For the tool 0020 tooth number 2.

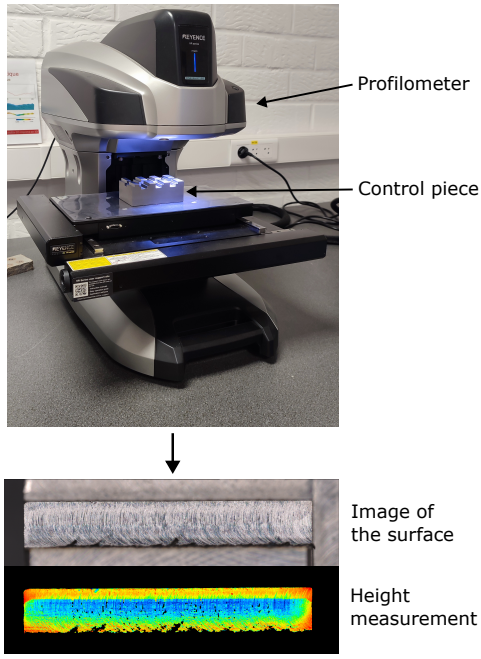


Figure 13.12: Keyence VR-6000 profilometer inspecting the surface roughness of a control piece.

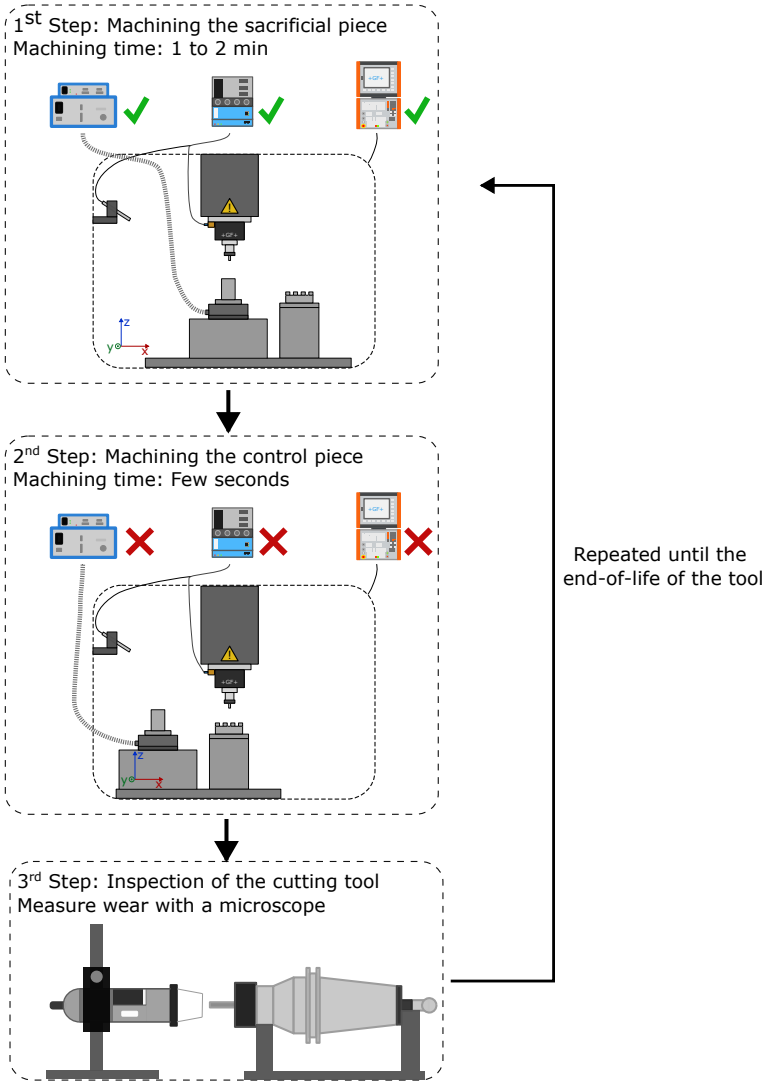


Figure 13.13: Overview of the machining procedure divided into three steps: first, machining the sacrificial workpiece while recording sensor data; second, machining the control workpiece without any data acquisition; and finally, inspecting the cutting tool under a microscope. This procedure is repeated until the tool reaches its end of life.

tests (see Section 13.3). Once the first pass is completed, the tool retracts and repositions itself to begin the second pass. This machining strategy is known as ZIG machining. The operation is repeated 30 times until the entire layer is machined. In practice, however, it was preferred to work in half-layers to allow for more frequent interruptions in the process. Nevertheless, the principle remains the same. During all machining passes, the various online sensors continuously record signals. Therefore, each layer acquisition in the database contains 15 passes.

Once the layer has been machined, the process is paused for one minute to immobilize the machine and allow the acquisition systems to stop recording. For the second part, the tool is then moved to the control workpiece. At this stage, a notch is machined into the control workpiece using the same cutting conditions that were applied to the last layer of the sacrificial workpiece (Figure 13.15). This notch is created in four passes of 2 × 2 mm, resulting in a cut with dimensions of 4 × 4 mm. This size was chosen to allow for subsequent surface roughness measurements. This machining operation takes a few seconds. During machining, no signal are measured. The position of the notch depends on the number of the layer that was just machined, enabling traceability of the surface quality produced after machining each corresponding layer. In the example shown in Figure 13.15, the machined notch corresponds to the reference notch for the fifth machined layer.

Finally, once the sacrificial and control workpieces have been machined, the chuck and tool are removed from the machine to inspect tool wear using a microscope. After the inspection, the chuck is reinstalled on the machine, and the procedure is repeated until the tool reaches the end of its life. The end-of-life criteria applied in this study do not follow the ISO 8688 standard [28]. During the tests, it was necessary to exceed the standard end-of-life threshold in order to investigate the tool's behaviour in a worn state. Consequently, machining was stopped when the tool began producing excessive sparks and the entire flank face was degraded. This criterion approximately corresponds to a VB_{max} value of around 600 µm. Once the tool is fully worn out, the control workpiece is removed from the machine for surface analysis using the profilometer.

13.3 Cutting Conditions

To plan the experimental campaign, a preliminary study was conducted to define the cutting conditions and validate the sensor acquisition setup. The manufacturer's recommended nominal cutting conditions are listed in Table 13.2, along with the test conditions selected for this study. Compared to the recommendations, the cutting speed was almost doubled to $112.5\,\mathrm{m\,min^{-1}}$. This adjustment was made to accelerate tool wear, shorten the duration of the tests and reduce material waste. This adjustment ensures a tool life of 7 to 10 min, which aligns with the standards for cutting tool life tests [28]. The feed per tooth was kept the same. Finally, the radial and axial depth of cut were set to 2 mm.

The test cutting conditions varies the two parameters that have the most influence on tool wear: cutting speed and feed per tooth. These two parameters are divided into three levels, as shown in Table 13.3. An experimental plan testing the nine combinations of these three levels is followed for the machining tests. Each combination is repeated three times (i.e., until the end-of-life of the tool is

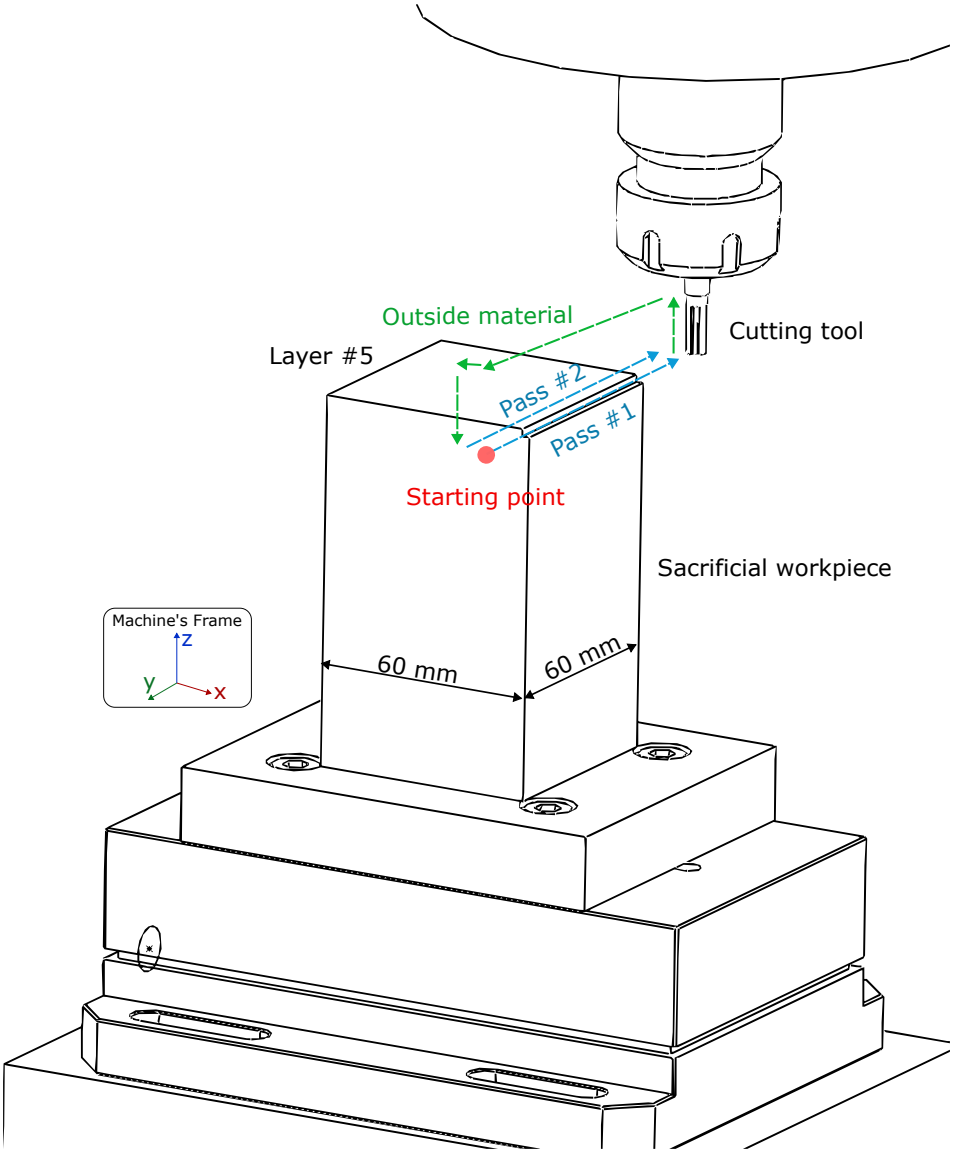


Figure 13.14: First step of the experimental campaign: ZIG machining. In blue, the cutting tool is in the material at the speed of the determined feed rate. In green, the cutting tool move at its maximum velocity as it is outside material. Example with the machining of the layer #5 with only the 2 first pass out of 15 represented.

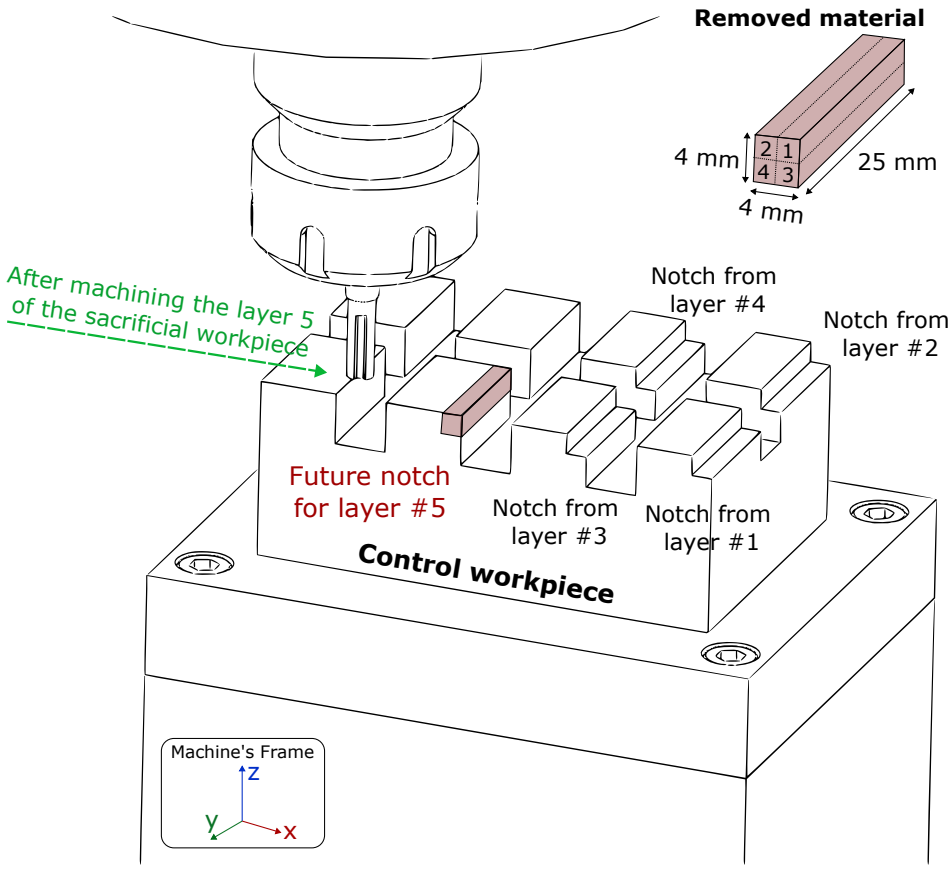


Figure 13.15: Second phase of the experimental campaign. The tool machines the control workpiece at the position associated from the layer it machined in the first phase. Example from the layer number 5 associated with the Figure 13.14. The removed material shows how the material is removed from the 4 differents passes.

Table 13.2: Nominal and determined cutting condition to perform the test.

	v_c [m min^{-1}]	f_z [mm tooth^{-1}]	a_p [mm]	a_e [mm]	N [rot min^{-1}]
Nominal	60	0.0327	2.5	5	3820
Determined	112.5	0.0327	2	2	7162

observed). Therefore, the experimental plan involves the degradation of 27 tools under nine different cutting conditions (Table 13.4). To keep track of each cutting tool, a unique identifier is assigned to every tool. The tools are numbered using the format 000T, where T represents the tool number. Each cutting condition is tested using a group of three tools. For example, test number 1 corresponds to tools 0001, 0002, and 0003.

Table 13.3: Cutting conditions modified during the tests, with three associated levels.

Factor	**Level 1**	**Level 2**	**Level 3**
v_c [m min^{-1}]	100	112.5	125
f_z [mm tooth^{-1}]	0.0327	0.0376	0.0425

Table 13.4: All combination of cutting conditions tested during the experimental test.

Test #	v_c **[m min^{-1}]**	f_z **[mm tooth^{-1}]**	**Repetition**	**Tool #**
1	112.5	0.0327	3	0001, 0002, 0003
2	112.5	0.0376	3	0004, 0005 ,0006
3	112.5	0.0425	3	0007, 0008 ,0009
4	100	0.0327	3	0010, 0011, 0012
5	100	0.0376	3	0013, 0014, 0015
6	100	0.0425	3	0016, 0017, 0018
7	125	0.0327	3	0019, 0020, 0021
8	125	0.0376	3	0022, 0023, 0024
9	125	0.0425	3	0025, 0026, 0027

13.4 Signal Acquisition and Organisation of the results

The data acquisition procedure was fully developed specifically for the experimental campaign. As previously shown (see Figure 13.1), three on-line acquisition chains were used in coordination: the Kistler acquisition chain for cutting forces, the NI acquisition chain for accelerations and sound, and the acquisition chain for retrieving power signals provided by the CNC. During the experimental campaign, each acquisition chain was connected to a dedicated computer. For the NI data acquisition, a complete software solution was developed to capture the signals. For the other acquisition chains, the manufacturers' proprietary software for the sensors and the machine was used. Table 13.5 summarizes the three acquisition chains, including the number of channels and the sampling frequency for each. While not all channels are directly used in the analysis presented in this document, the total corresponds to 684,800 measurement points per second of machining. In addition to these acquisitions, six photos of the tool were taken for each degradation measurement in order to monitor wear on each of the tool's teeth. In total, the raw dataset amounts to approximately 500 GB, collected over 10 days of machining. This corresponds to an average of 18.5 GB of raw data per tool. A dedicated

data management program was fully developed to handle this large volume of data generated during the tests. This program enables the management of various signals, correction of missing values, compensation for acquisition drifts, and resynchronization of the different acquisition chains despite their differing sampling frequencies. It also supports the generation of all analyses presented in the following chapters as well as images analysis.

Table 13.5: Summary of the three acquisition chains, including the number of measured channels and their respective sampling frequencies.

Measured	**Nb of Channels**	**Frequency [Hz]**
Cutting forces	16	20,000
Accelerations and sound	4	51,200
Machine axis power	16	10,000

Due to the large amount of data that are gathered during the machining process, the structure of the data is at least as important as the tests themselves. To ensure efficient analysis, data must be easily accessible and follow a logical and consistent structure. This allows data processing tools to directly target the relevant information without ambiguity. The data are stored according to the directory tree shown Figure 13.16, where T represents the tool number and L the layer number. Since the acquisition board for both sound and acceleration is the same, it was decided not to create a separate "Sound" folder. This choice helps maintaining synchronization between these two types of data.

```
Tool Number format:  000T
├─ Accelerometer
│  └─ files in format:000T_Accelerometer_00L.csv <-with sound
├─ Force
│  └─ Files in format:000T_Force_00L.csv
├─ Power
│  └─ Files in format:000T_Power_00L.csv
├─ Microscope
│  └─ Folder 000L
│     └─ Images of the tool
└─ Roughometer
   └─ 000T.xlsx Excel files containing every measurement
```

Figure 13.16: Architecture of the directory tree for organizing data.

The image files are organized into subfolders indicating whether the image was taken at the tool tip (B) or on the tool flank (R). The number following the letter

(B or R) refers to the tooth number shown in the image. For example, the file name 0010_R1_003.png corresponds to a flank image (R) of tooth number 1, taken after the third layer for tool 10. Figure 13.17 shows an example of acquisition directory for tool number 10

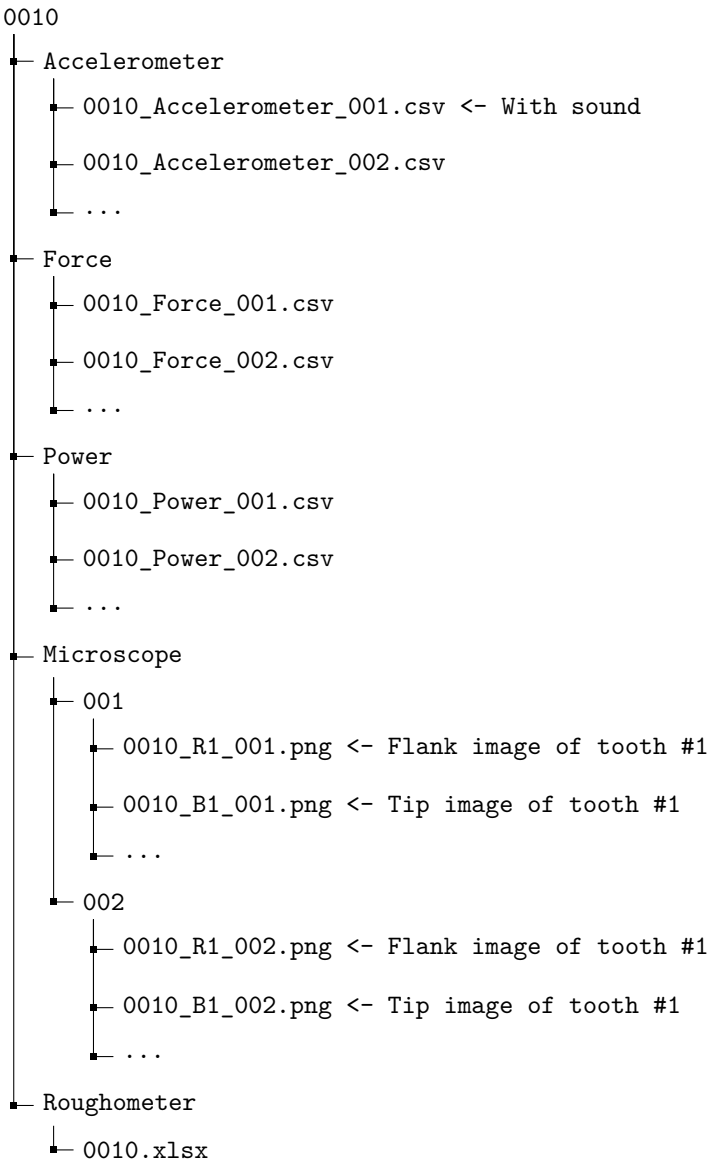


Figure 13.17: Architecture of the directory tree for organizing data of tool 0010.

Thanks to this file naming format, it is possible to directly locate the corresponding file by knowing the tool number, the desired layer, and the type of sensor.

This is particularly useful for data analysis and for synchronizing the acquisitions with cutting conditions, for example.

13.5 Conclusions

This chapter presents the experimental setup, equipment, sensors, machining procedure and cutting conditions used to create the milling database. This database is designed to complement the turning database while addressing the weaknesses previously identified in turning.

Table 13.6 compares the features of the turning and milling databases. All limitations identified in the introduction of this chapter were taken into account in the development of the milling database:

- Collect sufficient data. Compared to the turning database, four times more sensors were used, and 3688 passes were recorded versus 192 in turning. Since artificial intelligence methods were successfully applied to the turning dataset, the increased data volume provides confidence in applying AI techniques to this new database.
- Compare the correlation between multiple sensors. This objective is also addressed, as more than one sensor is used during the tests.
- Evaluate the impact of cutting conditions. The cutting conditions were selected to ensure observable variations in tool degradation.
- Ensure consistency in image acquisition. The procedure for analyzing cutting tool wear is fully defined, ensuring that all images are captured under consistent conditions.

Table 13.6: Comparison of the turning and milling database.

	Turning (see Chapter 6)	Milling
On-Line Sensors	Cutting forces	Cutting forces Accelerometer Microphone Machine's power
Off-Line sensor	Microscope Roughometer	Microscope Profilometer
Number of tools	30	27
Cutting conditions	5	9
Number of pass measured	192	3688

In the following chapter, the trends observed in the new dataset will be described.

Chapter 14

Analysis of the Milling Database

14.1 Introduction

This chapter analyses the trends observed in the database described in Chapter 13. Its objective is to present the key observations made during the experiments, focusing both on tool degradation and the evolution of various indicators in relation to the tool's condition. The chapter is structured as follows (Figure 14.1):

- Section 14.2 describes how the tools degraded during the experiments. It also outlines the methodology used to measure tool wear.
- Section 14.3 explains how tool degradation affects the surface quality produced by the tool.

With knowledge of the tool's degradation, it becomes possible to correlate the behaviour of various sensors with the wear progression.

- Section 14.4 presents how time-domain and frequency-domain indicators are calculated for the cutting forces. All the indicators used are detailed in this section and referenced in the following sections without being reintroduced. The correlation of the computed indicators in relation with the wear is also presented.
- Section 14.5 presents how time-domain and frequency-domain indicators are calculated for the accelerometer. The correlation of the computed indicators in relation with the wear is also presented.
- Section 14.6 presents how time-domain and frequency-domain indicators are calculated for the sound. The correlation of the computed indicators in relation with the wear is also presented.

- Section 14.7 presents how time-domain and frequency-domain indicators are calculated for the power of the axes of the machine. The correlation of the computed indicators in relation with the wear is also presented.
- Section 14.8 presents the global view of the correlation of all indicator extracted from all the measured signals.

Based on the previously described feature extraction methods, the structure of how the milling database is used throughout the remainder of this section can now be outlined:

- Section 14.9 details the feature selection process and the categorization of the extracted indicators.
- Section 14.10 explains how the dataset is partitioned for training and evaluating the different models.

Finally, Section 14.11 concludes this chapter.

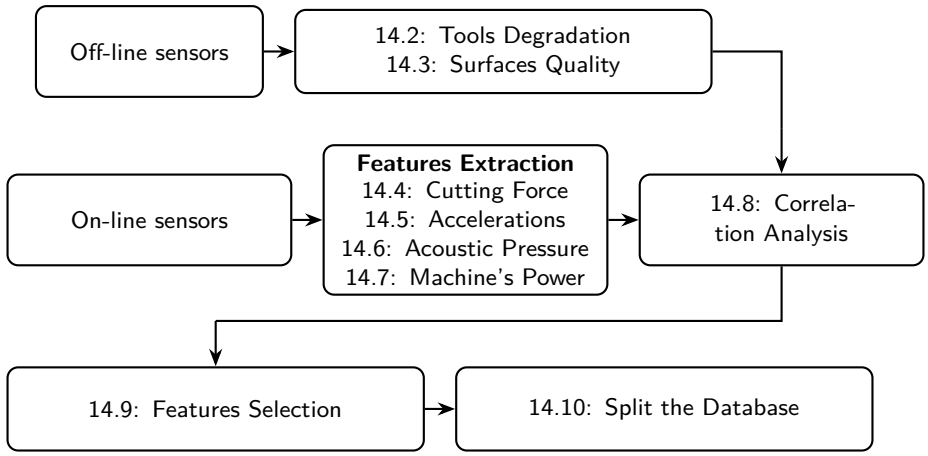


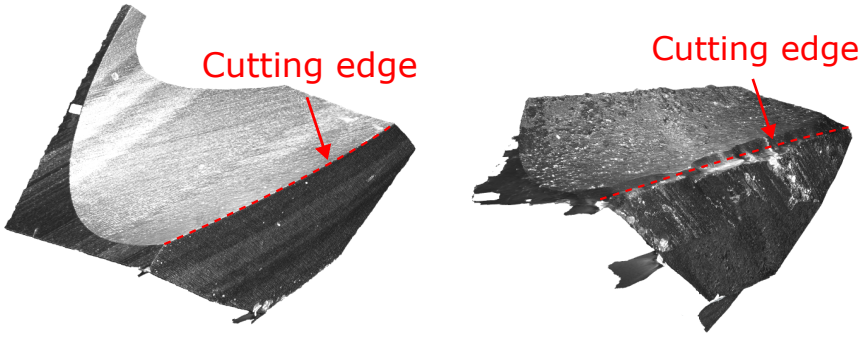
Figure 14.1: Flowchart illustrating the approach developed in this chapter, structured according to the data preparation logic outlined in Chapter 2.

14.2 Cutting tool's Degradation

During machining, tool wear occurs primarily in two areas: the flank and the tip. Throughout the experiments, wear was monitored using a microscope that captured images of both regions (see Chapter 13, Section 13.2.2). The effect of tool wear is clearly visible in Figure 14.2, which compares scans of a new tool and a worn one. Visually, the cutting edge appears damaged, indicating that the worn tool can no longer machine the material properly.

This section describes the evolution of flank wear in Section 14.2.1, and the progression of tip wear in Section 14.2.2. For flank wear, the ISO 8688 standard [28] introduces the VB indicator, which is used to monitor tool degradation.

Section 14.2.3 explains how this standard is applied to measure wear from the captured images. Once the wear measurement procedure is established, it becomes possible to analyse different wear trajectories under varying cutting conditions. This analysis is presented in Section 14.2.4. Additionally, an analysis of how the roughness of the different cutting-tool faces changes with wear is provided in Appendix B.



(a) Side edge of a new cutting tool. The cutting edge is intact.

(b) Side edge of a worn cutting tool. The cutting edge is damaged.

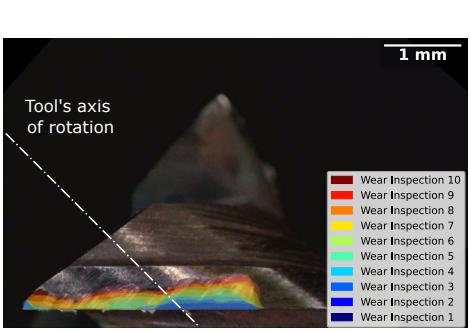
Figure 14.2: Image of the side cutting edges of a new and a worn cutting tool acquired using an Alicona Infinite Focus G6 (see Figure 13.5a).

14.2.1 Flank Wear Degradation Pattern

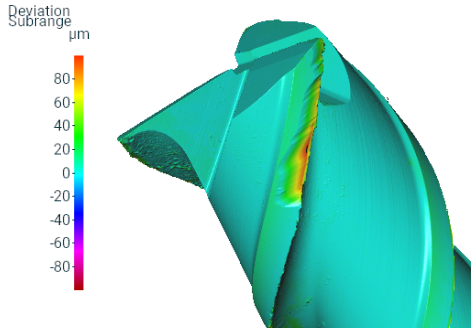
The flank wear of the tool is observed with the microscope presented in Section 13.2.2 with a magnification of 50×. During the test, it was observed that all tool exhibits similar wear progression pattern. For example, Figure 14.3a shows the evolution of flank wear for tool 0020 using a single superimposed image. Figure 14.3b shows a 3D deviation map obtained from the subtraction between the geometry of a new tool and the worn cutting tool. As with the other tools, the tooth exhibits flank wear extending just under 3 mm, which is consistent with a depth of cut of 2 mm and a tool helix angle of 45°. The wear progression follows this pattern: it begins approximately 2 mm along the tooth and gradually expands. At the same time, the extremity of the tooth also begins to degrade. Over time, these two wear zones converge, eventually resulting in wear across the entire tooth surface. Inside the tooth, crater wear is also visible (see Figure 14.4). This type of wear was not measured systematically.

14.2.2 Tip Wear Degradation Pattern

Similarly to the flank wear, the tip wear also exhibits similar characteristics for all tools. Figure 14.5a shows the degradation of tip wear for tool 0020. Figure 14.5c presents the deviation map between a new and a worn tool from the tip view. The wear is located at the extremity of the teeth. Unfortunately, tracking this



(a) Example of the flank degradation of the cutting tool 0020 superimposed on one image. The machining time between two inspections is 1.1 min.



(b) Flank view of the deviation map between a new and worn tool obtained with the Alicona Infinite Focus G6 (see Figure 13.5a).

Figure 14.3: Flank wear of the cutting tool

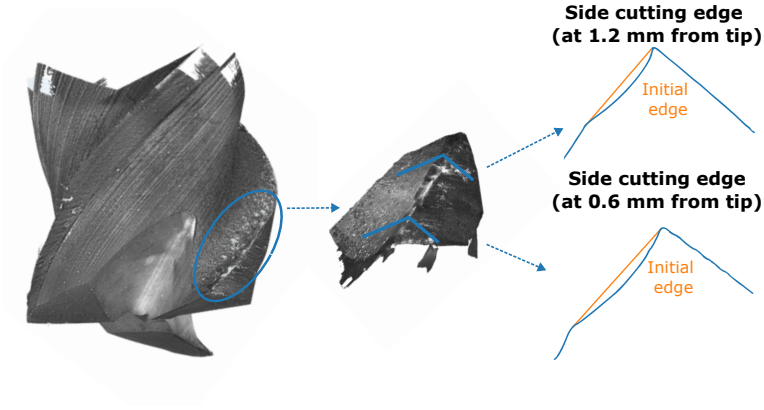


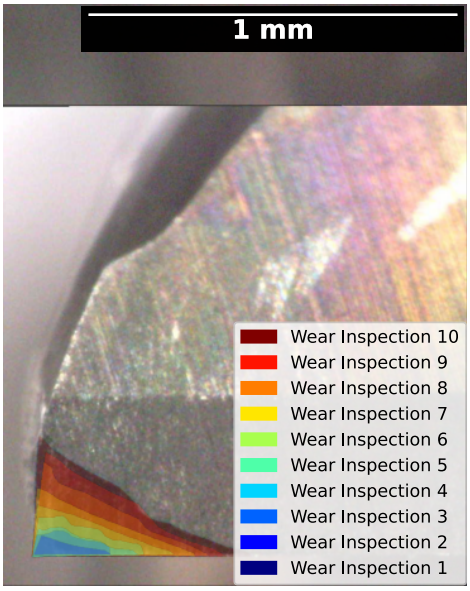
Figure 14.4: Crater wear of the worn cutting tool 0020

degradation is more complex due to the experimental setup (lighting conditions, alignment of the tool with the microscope, etc.), making the exact identification of the wear zone more challenging (See Chapter 13 Section 13.2.2). Values that could not be precisely measured are interpolated based on those that could be. The degradation pattern consistently starts at the tip of the tooth and extends in a triangular shape over time. In some cases, it has been observed that a part of the tooth would also break (Figure 14.5b).

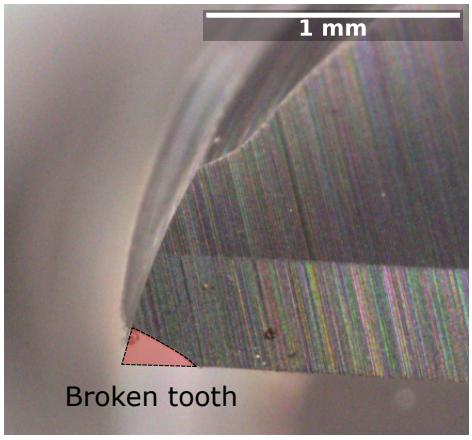
14.2.3 Wear Measurement

ISO 8688 addresses the procedure for measuring the degradation of milling cutters [28]. For flank wear, two indicators are extracted: maximum wear (VB_{max}) and average wear (VB_{mean}).

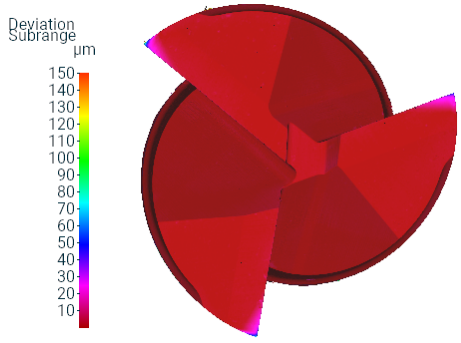
In the standard, the degradation pattern is supposed to be uniform. Due to this hypothesis, the wear is evaluated along the degradation. However, this is not what



(a) Example of the tip degradation of the cutting tool 0020 superimposed on one image. The machining time between two inspections is 1.1 minutes.



(b) Example of a broken tip after the first pass.



(c) Tip view of the deviation map between a new and worn tool obtained with the Alicona Infinite Focus G6 (see Figure 13.5a).

Figure 14.5: Example of tip images acquired during the experimental campaign.

was observed during the tests. Therefore, the standard has been interpreted to measure the two indicators: wear is evaluated from the tip of the tool up to 3 mm from it (Figure 14.6). This length of 3 mm corresponds to the contact between the tool and the workpiece. This value is confirmed by observing the wear patterns as mentioned in Section 14.2.1. This differs from the standard, which does not impose an evaluation length. This does not impact the measurement of VB_{max}, which is evaluated at the maximum wear point of the tool over this evaluation length. The average wear (VB_{mean}) is then calculated following the method recommended by

the standard, by taking the mean wear value over the evaluation zone length, even in cases where certain parts of the tooth show no visible wear (particularly in the middle of the evaluation zone).

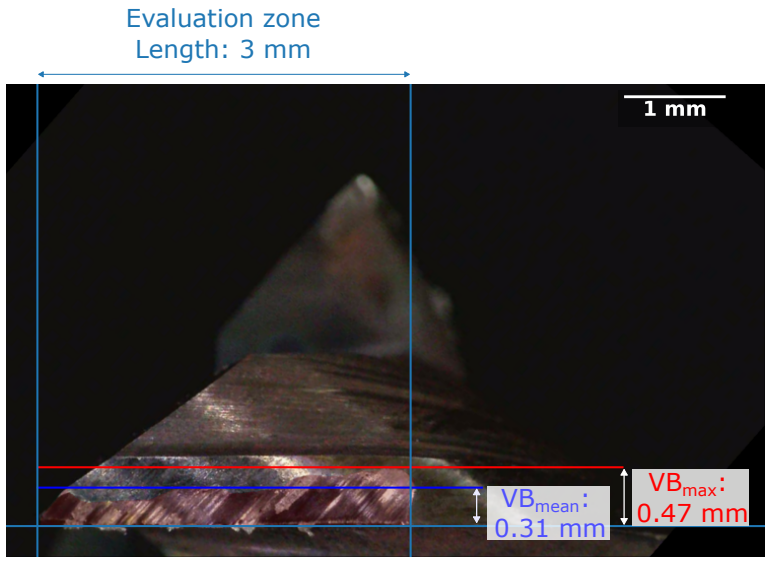


Figure 14.6: Wear measurement for tool 0020 layer 010. Evaluation zone length is 3 mm.

14.2.4 Degradation for Different Cutting Conditions

Thanks to image-based degradation measurements, it is possible to monitor tool wear over time. Figure 14.7 illustrates the degradation of all cutting tools, grouped by cutting condition. This visualization highlights the overall trend in tool wear. A key observation is that the degradation trajectories do not follow the typical S-shaped curve commonly associated with tool wear progression (see Chapter 2 Section 2.2). This deviation is primarily attributed to the fact that the cutting operations were not performed under the recommended machining conditions. Figure 14.7 also shows that some tools did not reach an average degradation of 300 µm. This is because, during the tests, due to the non-uniform wear pattern, the VB_{max} criterion of 600 µm was sometimes reached before the VB_{mean} value of 300 µm. Due to this excessive wear, proper cutting was no longer possible leading to the end-of-life of the tool. Figure 14.8 shows tool degradation grouped by feed per tooth. For each feed value, the figure presents the different degradation trajectories corresponding to various cutting speeds. These results highlight that, for a given feed, the cutting speed significantly influences the slope of the degradation curve. This result is similar to the observation of the turning database (see Chapter 6) on the same material. It also aligns with similar findings reported in the literature [134]. An important conclusion from this observation is that the selected cutting conditions clearly affect the tools degradation behaviour.

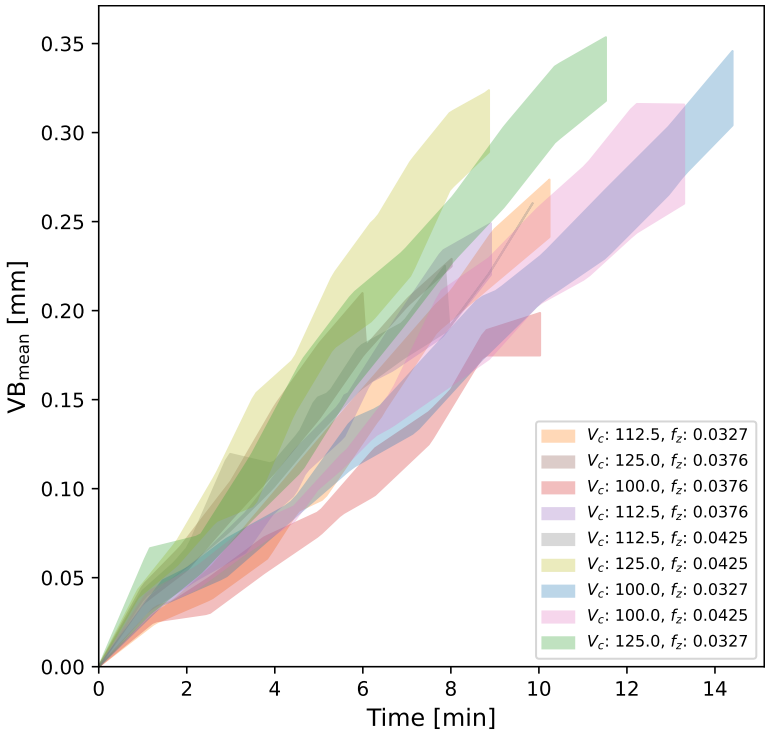
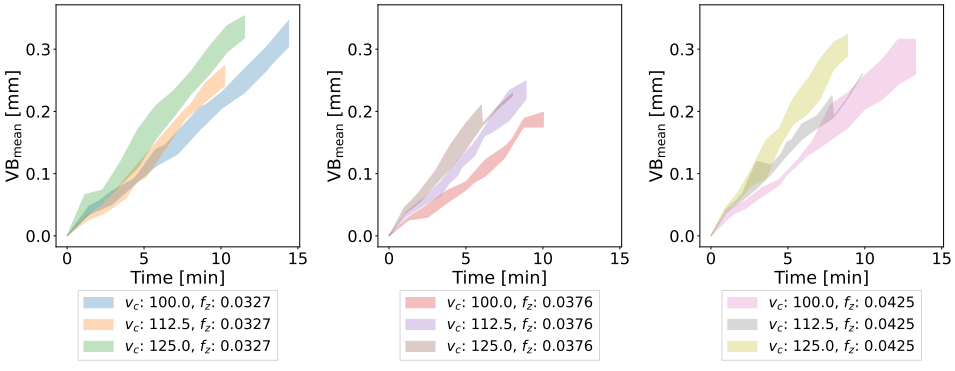


Figure 14.7: Superimposed degradation of the cutting tools for different cutting conditions.

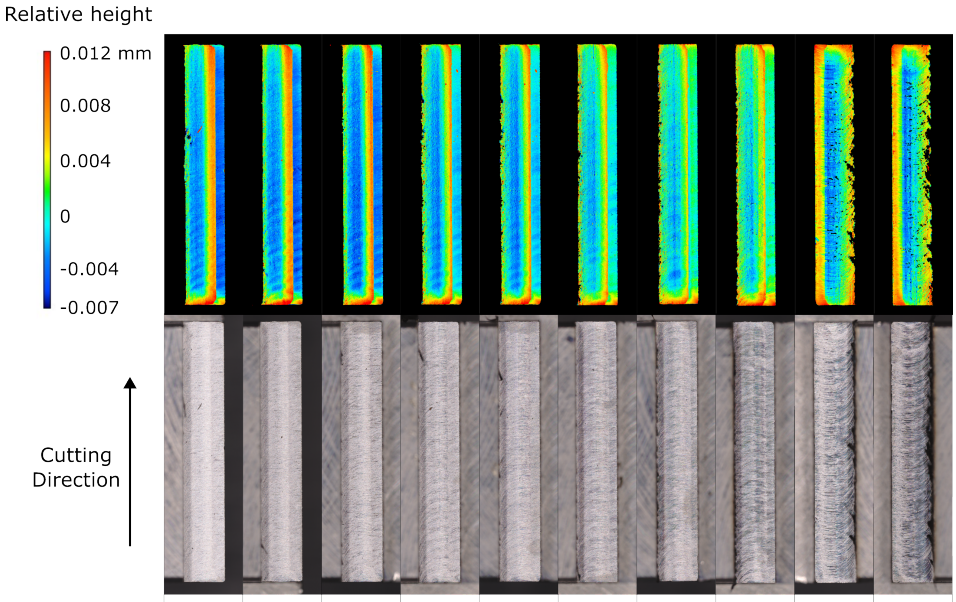


(a) f_z: 0.0327 mm tooth^{-1} **(b)** f_z: 0.0376 mm tooth^{-1} **(c)** f_z: 0.0425 mm tooth^{-1}

Figure 14.8: Degradation of cutting tools for fixed feed per tooth.

14.3 Roughness and Quality of Surfaces Produced

As described in Chapter 13, the control workpiece is used to preserve the surface finish produced by the tool at a specific point in its lifetime. This allows for the assessment of how tool wear affects the quality of the machined surfaces. Figure 14.9 was obtained using the profilometer described in Section 13.2.2. The figure is organized from left to right, with the leftmost image showing the surface produced by a new tool, and the rightmost image showing the surface produced by a worn tool. The top part of each image displays the height profile of the machined surface, while the bottom part shows a photograph of the surface itself. For each surface, the corresponding tool wear, measured surface roughness, and its roughness class (as defined by ISO 21920 see Table 14.1 [176]) are indicated. The figure illustrates that as the cutting tool wears, the surface roughness of the machined surface deteriorates. Specifically, when VB_{mean} reached between 170 and 205 µm, a change in roughness class is observed. In the middle of each height profile, a line can be seen appearing at the intersection between two cumulative passes (see Chapter 13, Section 13.2.3).



VB_{mean} [µm]	34	50	81	111	131	170	205	229	265	311
R_a [µm]	0.96	0.99	1.01	1.16	1.29	1.41	1.69	1.89	2.18	2.66
Roughness Class	N6	N6	N6	N6	N6	N6	N7	N7	N7	N7

Figure 14.9: Example of surfaces produced by tool 0020 throughout its lifetime. The tool's lifespan is represented from left to right, with the left side corresponding to a new tool and the right side to a worn tool.

Figure 14.10 illustrates the evolution of arithmetic surface roughness over time

Table 14.1: Surface roughness classes according to ISO 21920 [176].

Ra [µm]	Roughness Class
]6.3, 12.5]	N9
]3.2, 6.3]	N8
]1.6, 3.2]	N7
]0.8, 1.6]	N6
]0.4, 0.8]	N5
]0.2, 0.4]	N4

under different cutting conditions. Across all conditions and tools, surface quality consistently deteriorates, as evidenced by an increase in arithmetic roughness. This increase causes most tools to shift from roughness class N6 to N7. However, no specific trend related to the cutting conditions can be identified.

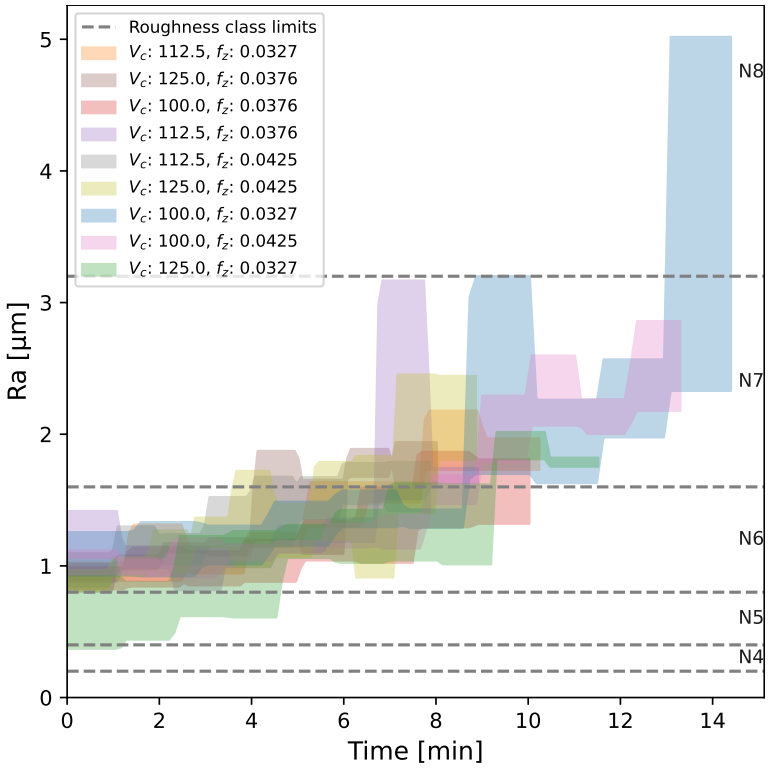


Figure 14.10: Arithmetic roughness in function of machining time for all cutting conditions.

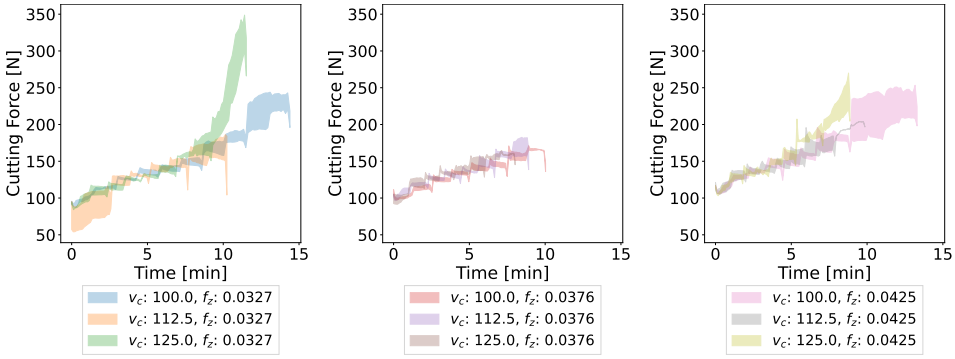
14.4 Evolution of the Cutting Forces Indicators with Tool Wear

The cutting force measurements are carried out using the Kistler sensor described in Chapter 13 Section 13.2.2. This setup enables the acquisition of forces in all three spatial directions. However, in this section, only the cutting forces (i.e., the forces along the machining direction) are considered. These forces are analyzed from both a temporal perspective (Section 14.4.1) and a frequency-domain perspective (Section 14.4.2). Finally, the correlation between cutting forces and tool wear is presented (Section 14.4.3).

14.4.1 Cutting Forces – Statistical Indicators from the Time Domain

As for the turning database (Chapter 6), multiple temporal indicators are calculated, namely: the mean value, the root mean square (RMS), the standard deviation, the skewness and the kurtosis. In the following, only the evolution of the mean value will be presented as it is the most correlated with the wear (see Section 14.4.3).

Figure 14.11 presents the mean cutting forces evolution for a fixed feed per tooth. The literature indicates that tool degradation is associated with an increase in cutting force. Mechanically, this observation is consistent: as the cutting edge deteriorates, the tool becomes duller, requiring greater force to shear the material and form the chip [134]. This trend is also observed in the dataset. We can also observe small, fairly regular drops in cutting force. This phenomenon occurs because, during the tool's final pass through the material, its thickness tended to differ from the previous passes. As a result, the tool did not take a full 2 mm radial cut, thereby reducing the cutting force.



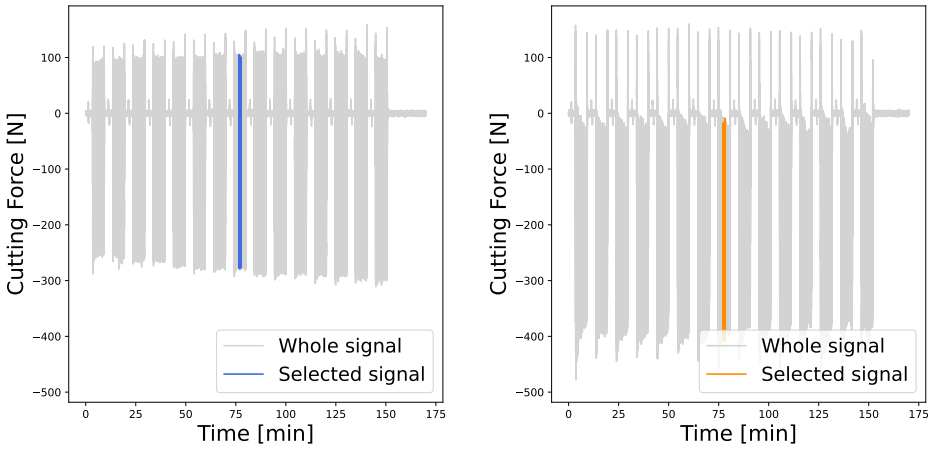
(a) f_z: 0.0327 mm tooth^{-1} **(b)** f_z: 0.0376 mm tooth^{-1} **(c)** f_z: 0.0425 mm tooth^{-1}

Figure 14.11: Mean cutting force evolution in function of the flank wear for different feed conditions.

14.4.2 Cutting Forces – Statistical Indicators from the Frequency Domain

The cutting force signal will be used to demonstrate the frequency analysis performed on all the other signals of the database.

Figure 14.12 presents the cutting force signals used for comparison in this demonstration. Figure 14.12a displays the selected signal when the cutting tool (tool 0020) is new. For consistency, the middle pass of the layer was chosen, and within this pass, a 1 second segment of machining was extracted. This segment represent around 10 mm of machined length. This selection excludes transient phenomena associated with the tool entering and exiting the sacrificial workpiece. The same selection method is applied for the worn tool in Figure 14.12b.



(a) Cutting force of a new cutting tool (Tool: 0020)

(b) Cutting force of a worn cutting tool (Tool: 0020)

Figure 14.12: Cutting force evolution for a new and worn tool (Tool: 0020). The highlighted area show the selected signal on which the frequency analysis will be presented.

During machining, a particularly important frequency is the tooth passing frequency, which corresponds to how often the tool's teeth engage with the material. This frequency, denoted as f_t, depends on the cutting conditions and can be calculated using the following equation:

$$f_t = \frac{N}{60} \cdot N_t \tag{14.1}$$

where:

- N is the spindle speed in revolutions per minute (RPM),
- N_t is the number of teeth on the cutting tool.

Under the cutting conditions used in the database, this frequency ranges between 318 Hz and 398 Hz. For the tool selected to present the frequency analysis (tool

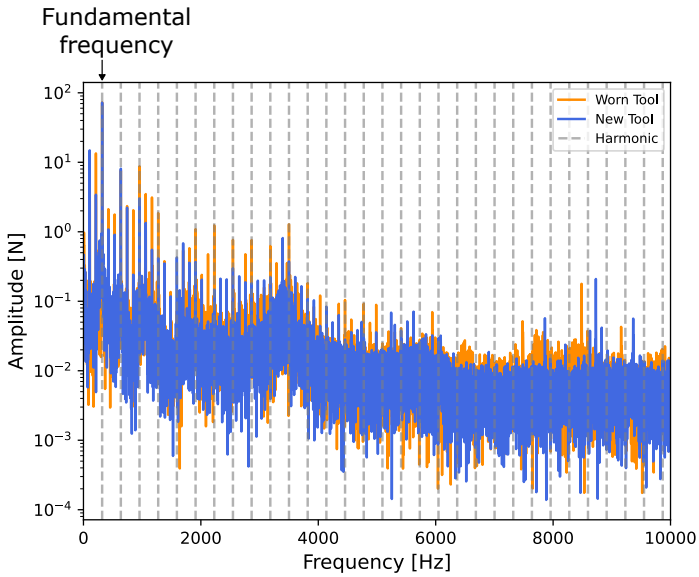


Figure 14.13: Discrete Fourier Transform of the cutting force of tool 0020. Comparison at the beginning of its life and at the end.

0020), this f_t corresponds to 318 Hz. Figure 14.13 shows the Discrete Fourier Transform (computed with the Fast Fourier Transform algorithm) of the signal corresponding to the new tool and the signal corresponding to the worn tool. It highlights that when a tool is worn, it exhibits higher frequency amplitudes compared to when it is new. The emergence of new frequency components and the increase in their amplitudes occur in a range above the fundamental tooth-passing frequency, f_t, indicating that tool wear affects the higher-frequency spectrum.

Figure 14.14 presents the Power Spectral Density (PSD) computed using Welch's method for both the new and worn tools [130]. This figure confirms the observations made with the Fourier transform: the power level of a worn tool is higher than that of a new one, particularly with increased peak amplitudes at frequencies above the fundamental frequency f_t.

This spectral analysis reveals the underlying mechanism of cutting tool degradation. At low wear levels, the tool's frequency characteristics are primarily associated with the periodic entry of the cutting edges into the material. As wear progresses, additional high-frequency components emerge in the spectrum, reflecting more complex dynamic interactions between the tool and the workpiece, along with an amplification of existing frequency components. Notably, no shift in the fundamental frequency f_t is observed, indicating that the basic cutting frequency remains unchanged, while the spectral content becomes richer due to wear-induced phenomena.

Tool degradation, therefore, has an impact on the frequency domain. To be used in certain models, these observations must be transformed into frequency-domain indicators, similar to how RMS values are used in time-domain analysis. In the existing literature, time signals are typically used directly or with basic indicators

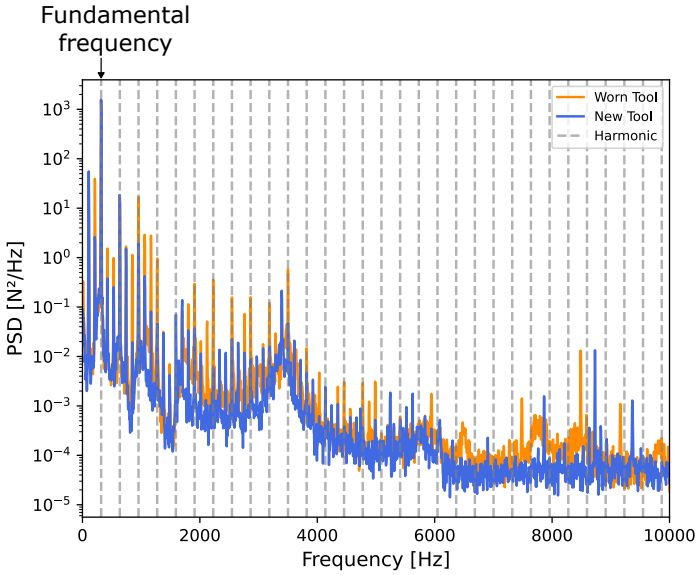


Figure 14.14: Power Spectral Density comparison between tool 0020. Comparison at the beginning of its life and at the end.

such as FFT components, or through highly specific frequency analyses tailored to particular cutting scenarios [177]. This significantly limits the applicability of frequency-based methods to other contexts. For the milling database, simple approaches were tested but proved unsuccessful (FFT frequencies, ...). More specialized frequency analyses were not considered due to the variability in cutting conditions and the aim to characterize signals using indicators that could be reused across different datasets. Therefore, this chapter explores alternative indicators to better capture frequency variations in the signals. This work remains exploratory and focuses more on a educated signal analysis than on established literature related to cutting tool monitoring. The selected frequency-domain indicators are therefore those that highlight changes in frequency, the various ratios within the signals, and the appearance of high-frequency elements that could be considered noise. The frequency-domain indicators used hereafter are:

- **Spectral Energy**. This energy is calculated by taking the integral of the Power Spectral Density (PSD) over a defined frequency band. It highlights the increase in the total power of the signal. It is therefore expected that the appearance of higher frequencies due to wear phenomena will increase the spectral energy. If $S_{xx}(f)$ is the PSD of the signal $x(t)$, the energy E in the frequency band $[f_1, f_2]$ is given by:

$$E = \int_{f_1}^{f_2} S_{xx}(f)\, df \tag{14.2}$$

  For a discrete spectrum obtained by a Fast Fourier Transform (FFT), where

$X[k]$ is the k-th FFT coefficient, the equation becomes a summation:

$$E \propto \sum_{k=k_1}^{k_2} |X[k]|^2 \tag{14.3}$$

- **Harmonic to Noise Ratio (HNR)**. This ratio shows the relationship between the energy contained in the periodic components (harmonics) and the energy of the background noise. In the context of this analysis, the harmonic energy typically corresponds to the energy around the tooth passing frequency (f_t) and its multiples. As previously observed, this indicator is expected to detect the appearance of wear due to the presence of higher frequencies in the spectrum associated with tool wear. The HNR value should therefore drop as the wear will increase the noise on the signal. The HNR, expressed in decibels (dB), can be defined as:

$$\mathrm{HNR}_{dB} = 10 \log_{10} \left(\frac{E_{harmonic}}{E_{noise}} \right) \tag{14.4}$$

  Where $E_{harmonic}$ is the summed energy in narrow windows around the expected harmonic peaks, and E_{noise} is the energy in the rest of the spectrum, considered as noise. As previously mentioned, the HNR is expected to decrease with tool wear. To obtain an indicator that increases, thus visually aligning with the progression of wear, the HNR is inverted. In the following analysis, the inverted HNR is used without being explicitly stated.

- **Spectral Entropy**. This value functions similarly to the HNR indicator but measures the dispersion or uncertainty of the energy in the spectrum, without referencing a specific harmonic frequency. It quantifies the uniformity of the PSD's distribution. A healthy tool is expected to exhibit lower entropy compared to a worn one, due to a more distinct frequency peak and fewer high-frequency components in the signal. To compute the spectral entropy, first, the PSD is normalized to represent a probability distribution p_k:

$$p_k = \frac{S_{xx}(f_k)}{\sum_i S_{xx}(f_i)} \tag{14.5}$$

  The spectral entropy H_s is then calculated as

$$H_s = -\sum_k p_k \log_2(p_k) \tag{14.6}$$

- **Ratio of Cyclostationary Components (RCC)**. This indicator quantifies the proportion of the signal's energy that is due to cyclic (periodic) phenomena. It is therefore sensitive to the degradation of the cutting signal's periodic character. A perfectly periodic signal will have a high RCC. It is therefore expected that the appearance of noise and randomness due to wear decreases this ratio. It can be computed as the ratio of the energy contained in the harmonics to the total signal energy.

$$\mathrm{RCC} = \frac{\sum_{h=1}^{H} E(h \cdot f_t)}{E_{total}} \tag{14.7}$$

Where $E(h \cdot f_t)$ is the energy in a narrow band around the h-th harmonic of the tooth passing frequency (F_{teeth}) and E_{total} is the total energy of the spectrum.

- **Spectral Centroid and its Standard Deviation**. The spectral centroid acts as the "centre of gravity" of the spectrum, indicating where the majority of the energy is concentrated. A change in this value indicates that the energy is shifting, for example, towards higher frequencies, which can be a sign of wear. The spectral standard deviation (or spectral spread) measures the bandwidth of the spectrum around its centroid. An increase indicates that the energy is spreading out over a wider range of frequencies. The spectral centroid C_s is calculated as the weighted average of the frequencies, where the weights are the amplitudes of the spectrum:

$$C_s = \frac{\sum_k f_k \cdot |X[k]|}{\sum_k |X[k]|} \tag{14.8}$$

The spectral standard deviation σ_s, which measures the spread around the centroid, is given by:

$$\sigma_s = \sqrt{\frac{\sum_k (f_k - C_s)^2 \cdot |X[k]|}{\sum_k |X[k]|}} \tag{14.9}$$

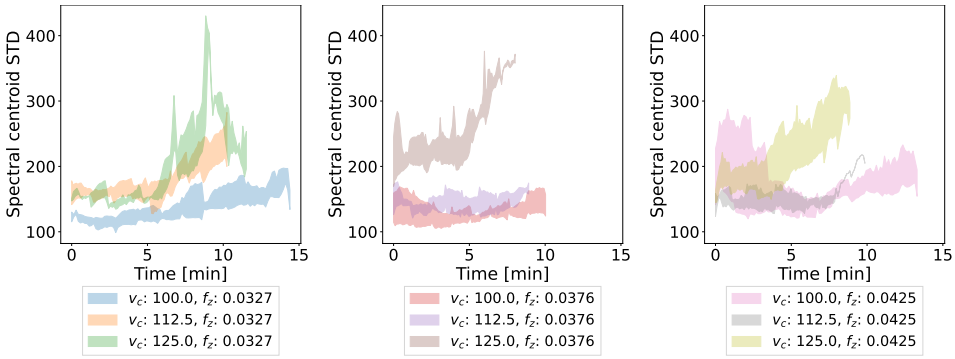
Where f_k is the frequency corresponding to the k-th FFT coefficient, $|X[k]|$.

These indicators have been selected to capture the main phenomena associated with tool wear. Spectral energy provides a global view of the system's power distribution. HNR and RCC offer insights into the signal's harmonic purity and the presence of cyclic components, respectively. Spectral entropy quantifies the emergence of random or irregular phenomena, while the spectral centroid indicates where the energy is concentrated within the frequency spectrum. Together, these indicators provide a comprehensive characterization of the frequency-domain evolution of the signals. For consistency, the same set of indicators will be applied to all other signals throughout the database.

To illustrate the behaviour of a specific indicator, the evolution of the spectral centroid standard deviation at a fixed feed per tooth is shown in Figure 14.15. This frequency indicator is selected as it is the most correlated with the wear (see Section 14.4.3). As with the temporal analysis, different cutting conditions are presented separately. However, in this case, identifying a clear trend is more challenging, suggesting that this indicator may be less sensitive to tool wear under the given conditions.

14.4.3 Cutting Forces – Indicators Correlation with Tool Wear

Tables 14.2 and 14.3 present the Spearman correlation coefficients between various temporal and frequency indicators and cutting tool wear, calculated across the entire dataset. As in Chapter 6, a coefficient of 1 indicates a perfect correlation, while a value of 0 signifies no correlation. Note that interactions between indicators are not considered in this analysis.



(a) f_z: 0.0327 mm tooth^{-1} **(b)** f_z: 0.0376 mm tooth^{-1} **(c)** f_z: 0.0425 mm tooth^{-1}

Figure 14.15: Spectral centroid of the cutting force evolution in function of the flank wear for different feed conditions.

From the results, it is evident that temporal indicators, particularly the mean and RMS values, show the strongest correlation with tool wear, at 0.91 and 0.77 respectively. In contrast, frequency-based indicators exhibit weaker correlations, with the highest being approximately 0.42 for the centroid standard deviation. These findings suggest that, for this case, temporal indicators are more effective in capturing wear-related patterns than frequency-based ones. This observation is consistent with the literature, which tends to show that for cutting efforts, the useful extracted features are mainly from the time domain [178].

Table 14.2: Cutting force temporal indicators correlation with the wear.

Indicator	Correlation
Mean	0.91
RMS	0.77
Kurt	0.28
Skew	0.15

Table 14.3: Cutting force frequency indicators correlation with the wear.

Indicator	Correlation
Centroid STD	0.42
Entropy	0.40
HNR	0.29
RCC	0.27
Centroid	0.20
Spectral energy	0.08

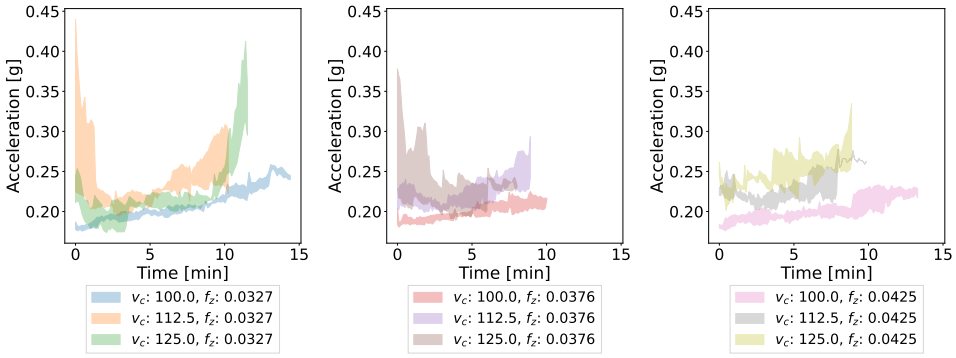
14.5 Evolution of the Accelerations Indicators with Tool Wear

The method for acquiring vibrations was described in Chapter 13 Section 13.2.2. As for the cutting force, only the vibrations in the machining direction will be presented in this section. First in the time domain (Section 14.5.1), and then in the frequency domain (Section 14.5.2). Finally, Section 14.5.3 shows the correlation of the extracted indicators with respect to the wear.

14.5.1 Accelerations – Statistical Indicators from the Time Domain

The time-domain features extracted from the accelerometer signals are the same as those previously defined for the cutting forces in Section 14.4. As it will be shown in Section 14.5.3, the extracted RMS indicator is the most correlated with the cutting tool wear.

Figure 14.16 shows the RMS acceleration in the cutting direction in function of time and for fixed Fz. This figure highlights that, over time (and consequently with tool wear) the vibration levels increase. Some trajectories exhibit significant vibrations during the initial minutes of machining; these correspond to the first five tools in the database. No clear explanation has been identified for the initially high acceleration levels, particularly given that they tend to decrease before rising again. Nonetheless, it is possible to identify that the cutting speed has an effect on acceleration levels for a fixed feed per tooth as a clear distinction between the cutting conditions can be distinguished.

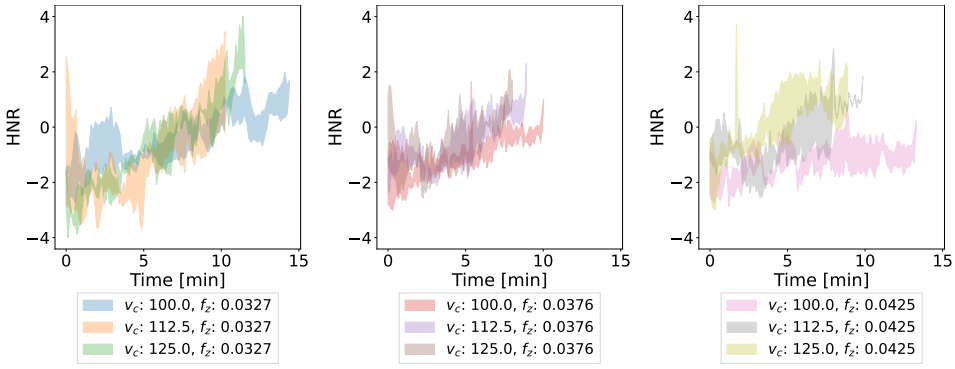


(a) f_z: 0.0327 mm tooth^{-1} **(b)** f_z: 0.0376 mm tooth^{-1} **(c)** f_z: 0.0425 mm tooth^{-1}

Figure 14.16: Acceleration in the cutting direction evolution in function of the flank wear for different feed conditions.

14.5.2 Accelerations – Statistical Indicators from the Frequency Domain

The extracted frequency indicator have been completely presented for the cutting forces in Section 14.4.2. In the context of the accelerometer, the most correlated feature from the frequency domain is the Harmonics-to-Noise Ration (HNR) (see Section 14.5.3). Figure 14.17 illustrates the evolution of the Harmonics-to-Noise Ratio (HNR) for acceleration in the cutting direction. An increase in this indicator is observed across all cutting conditions. However, the influence of these conditions is not clearly distinguishable, as all curves tend to overlap.



(a) f_z: 0.0327 mm tooth^{-1} **(b)** f_z: 0.0376 mm tooth^{-1} **(c)** f_z: 0.0425 mm tooth^{-1}

Figure 14.17: HNR indicator of the acceleration in the cutting direction evolution in function of the flank wear for different feed conditions.

14.5.3 Accelerations – Indicators Correlation with Tool Wear

Tables 14.4 and 14.5 present the Spearman correlation between tool wear and acceleration in the cutting direction, for both time-domain and frequency-domain indicators.

In the temporal domain (Table 14.4), the RMS value shows the highest correlation with wear, reaching 0.49. In contrast, the frequency domain (Table 14.5) reveals that the Harmonics-to-Noise Ratio (HNR) is the most correlated indicator, with a value of 0.65. This higher correlation suggests that, for this case, frequency-based indicators are more sensitive to tool wear than temporal ones. This observation is consistent with the literature [178].

Table 14.4: Acceleration in the cutting direction temporal indicators correlation with the wear.

Indicator	Correlation
RMS	0.49
Skew	0.43
Mean	0.25
Kurt	0.01

Table 14.5: Acceleration in the cutting direction frequency indicators correlation with the wear.

Indicator	Correlation
HNR	0.65
Centroid	0.62
Entropy	0.52
Spectral energy	0.48
RCC	0.44
Centroid STD	0.20

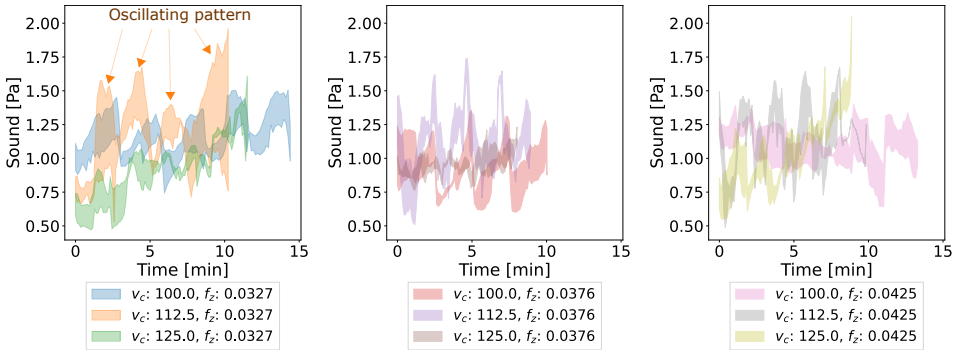
14.6 Evolution of the Acoustic Pressure Indicators with Tool Wear

The sound acquisition method is described in Chapter 13 Section 13.2.2. A time-domain analysis of the sound evolution is provided in Section 14.6.1, while the

frequency-domain analysis can be found in Section 14.6.2. The correlation between the sound and the extracted indicator is described in Section 14.6.3

14.6.1 Acoustic Pressure – Statistical Indicators from the Time Domain

As with the other signals, various time-domain indicators were extracted. Among these, the RMS value showed the strongest correlation with tool wear (see Section 14.6.3). Figure 14.18 shows the temporal evolution of the RMS value of the sound level over time. This figure clearly indicates that no consistent trend is observed. All signals display an oscillatory component, which may be attributed to the movement of the experimental setup within the machine during machining. Although the tool remains stationary relative to the microphone, the perceived sound level appears to be influenced by the varying positions of the workpieces. These positional changes likely alter the geometry of the machine's internal enclosure, affecting its acoustic resonance. Therefore, no clear correlation between tool wear and the sound level measured by the microphone can be established.

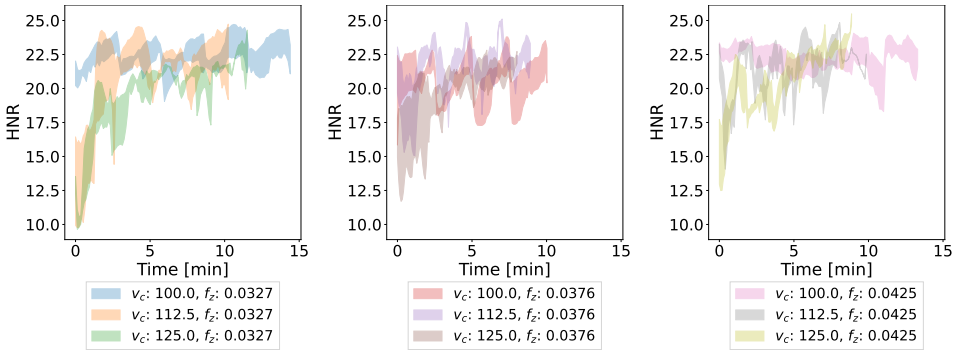


(a) f_z: 0.0327 mm tooth^{-1} **(b)** f_z: 0.0376 mm tooth^{-1} **(c)** f_z: 0.0425 mm tooth^{-1}

Figure 14.18: Sound level in the machine in function of the flank wear for different feed conditions. The oscillating pattern, illustrated for one cutting condition (Figure 14.18a), is representative and occurs across all tested conditions.

14.6.2 Acoustic Pressure – Statistical Indicators from the Frequency Domain

Among the time-domain indicators extracted from the signal, the HNR shows the strongest correlation with tool wear (see Section 14.6.3). Figure 14.19 shows the evolution of the Harmonics-to-Noise Ratio (HNR) under different cutting conditions. Similar to the temporal analysis, this signal is influenced by the spindle's position within the machine, which affects the reliability of the frequency-based analysis.



(a) f_z: 0.0327 mm tooth^{-1} **(b)** f_z: 0.0376 mm tooth^{-1} **(c)** f_z: 0.0425 mm tooth^{-1}

Figure 14.19: HNR of sound in the machine in function of the flank wear for different feed conditions.

14.6.3 Acoustic Pressure – Indicators Correlation with Tool Wear

Tables 14.6 and 14.7 show the correlation between sound-based indicators and cutting tool wear in the time and frequency domains, respectively.

In the temporal domain (Section 14.6.1), the RMS value of the sound signal shows the highest correlation, at 0.28. Although this is a relatively low value, it can be attributed to the fact that sound levels are more influenced by the tool's position within the machine than by wear itself.

In the frequency domain (Section 14.6.2), the Harmonics-to-Noise Ratio (HNR) is the most correlated indicator, with a value of 0.33. Again, this modest correlation suggests that the microphone may not be a particularly sensitive sensor for detecting tool wear.

In the literature, acoustic signals are considered particularly challenging to process due to the numerous disturbances that can occur in industrial environments. As a result, this type of information is less commonly used compared to other approaches for estimating tool condition [178]. Although some methods have successfully detected tool wear using sound signals [179], the milling database reveals that the audio signal is heavily affected by the spindle position. While frequency analyses tailored to the experimental campaign and incorporating spindle position corrections could be performed, such approaches were not considered within the scope of this work.

14.7 Evolution of the Machine's Power Indicators with Tool Wear

The machine power for each axis is extracted directly from the CNC machine, as described in Chapter 13 Section 13.2.2. In the following analysis, only the spindle power is considered. Its temporal evolution is discussed in Section 14.7.1, while the frequency-domain analysis is presented in Section 14.7.2. Section 14.7.3 examines the correlation between the extracted indicators and tool wear

Table 14.6: Sound temporal indicators correlation with the wear.

Indicator	Correlation
RMS	0.28
Mean	0.10
Skew	0.03
Kurt	0.03

Table 14.7: Sound frequency indicators correlation with the wear.

Indicator	Correlation
HNR	0.33
Centroid STD	0.32
Spectral energy	0.27
Centroid	0.19
Entropy	0.15
RCC	0.04

14.7.1 Machine's Power – Statistical Indicators from the Time Domain

As previously, various temporal indicators were extracted from the signal. The signal most correlated with wear is the RMS value (see Section 14.7.3).

Figure 14.20 illustrates the temporal evolution of the spindle's RMS power under various cutting conditions. Across all conditions, spindle power increases over time, reflecting the progression of tool wear. This trend is consistent with observations made for cutting forces (see Section 14.4.1). Since spindle power is directly related to cutting force, this consistency is expected. The relationship between spindle power and cutting force is defined by the following equation:

$$P_{\text{spindle}} = F \times v_c \tag{14.10}$$

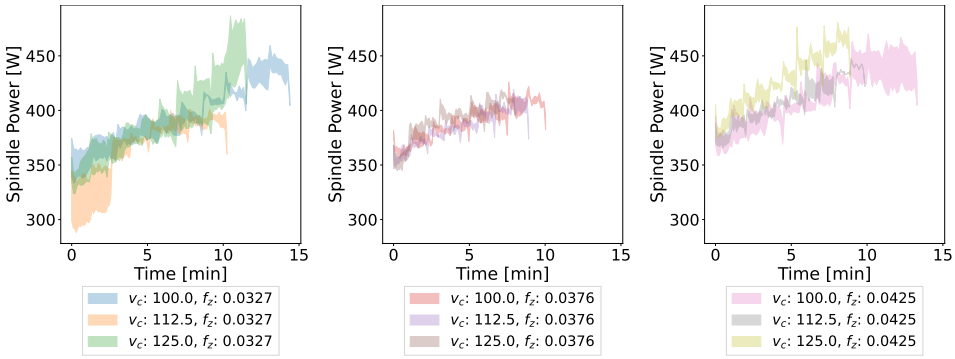
Where P_{spindle} represents the spindle power, F is the measured resultant cutting force, and V_c is the cutting speed in $\mathrm{m\,s^{-1}}$. In the dataset, the cutting speed ranges from $100\,\mathrm{m\,min^{-1}}$ to $125\,\mathrm{m\,min^{-1}}$, with an average of $112.5\,\mathrm{m\,min^{-1}}$, equivalent to $1.875\,\mathrm{m\,s^{-1}}$. The RMS cutting force varies between 125 N and 380 N, with an average of 201 N. Using these average values, the mean spindle power ($P_{\text{spindle mean}}$) is:

$$P_{\text{spindle mean}} = 201\,\mathrm{N} \times 1.875\,\mathrm{m\,s^{-1}} = 376.875\,\mathrm{W} \tag{14.11}$$

This estimation aligns well with the order of magnitude observed in Figure 14.20.

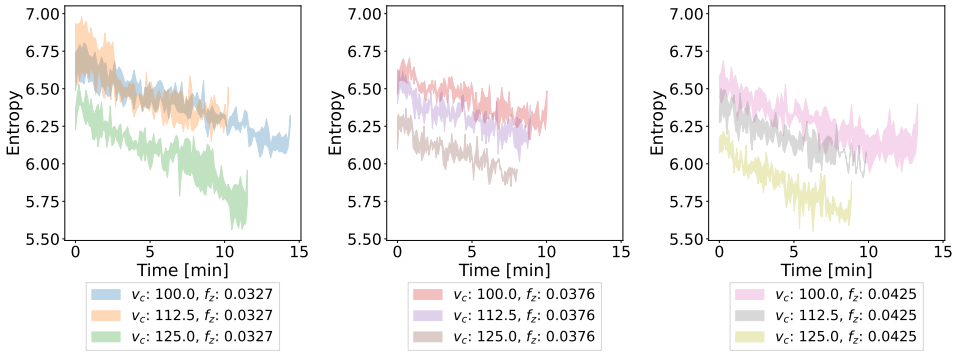
14.7.2 Machine's Power – Statistical Indicators from the Frequency Domain

From the extracted frequency indicator on the spindle power, the spectral entropy is the most correlated with the wear (see Section 14.7.3) Figure 14.21 illustrates the evolution of spectral entropy in the spindle power signal across different cutting conditions. A clear distinction can be observed between the various conditions. Additionally, spectral entropy consistently decreases over time, suggesting that tool wear has a measurable impact on this indicator's behaviour.



(a) f_z: 0.0327 mm tooth^{-1} **(b)** f_z: 0.0376 mm tooth^{-1} **(c)** f_z: 0.0425 mm tooth^{-1}

Figure 14.20: Spindle power in function of the flank wear for different feed conditions.



(a) f_z: 0.0327 mm tooth^{-1} **(b)** f_z: 0.0376 mm tooth^{-1} **(c)** f_z: 0.0425 mm tooth^{-1}

Figure 14.21: Spectral entropy for the spindle power in function of the flank wear for different feed conditions.

14.7.3 Machine's Power – Indicators Correlation with Tool Wear

Tables 14.8 and 14.9 present the correlation between spindle power indicators and tool wear in the time and frequency domains, respectively.

In the temporal domain (Table 14.8), the mean value shows the highest correlation, with a coefficient of approximately 0.77. In the Statistical Indicators from the Frequency Domain (Table 14.9), the most correlated indicator is the spectral centroid, with a correlation of 0.60. The results suggest that, for this particular sensor, time-domain features exhibit a stronger association with tool wear than frequency-based indicators. This observation aligns with the fact that spindle power reflects cutting force, which itself shows a higher correlation with wear through its time-domain features.

Table 14.8: Spinde power temporal indicators correlation with the wear.

Indicator	Correlation
Mean	0.77
RMS	0.75
Skew	0.57
Kurt	0.05

Table 14.9: Spinde power frequency indicators correlation with the wear.

Indicator	Correlation
Entropy	0.60
HNR	0.58
Centroid STD	0.58
Spectral energy	0.58
Centroid	0.56
RCC	0.51

14.8 Correlation of All Extracted Features with the Tool Wear

In the previous sections, the computation of various indicators was presented. Only those measured along the cutting direction were included in that analysis. Figure 14.22 display the overall correlation for all signals recorded during the tests. The directions (x, y, and z) are expressed relative to the CNC machine's frame, where the cutting direction corresponds to the y-axis.

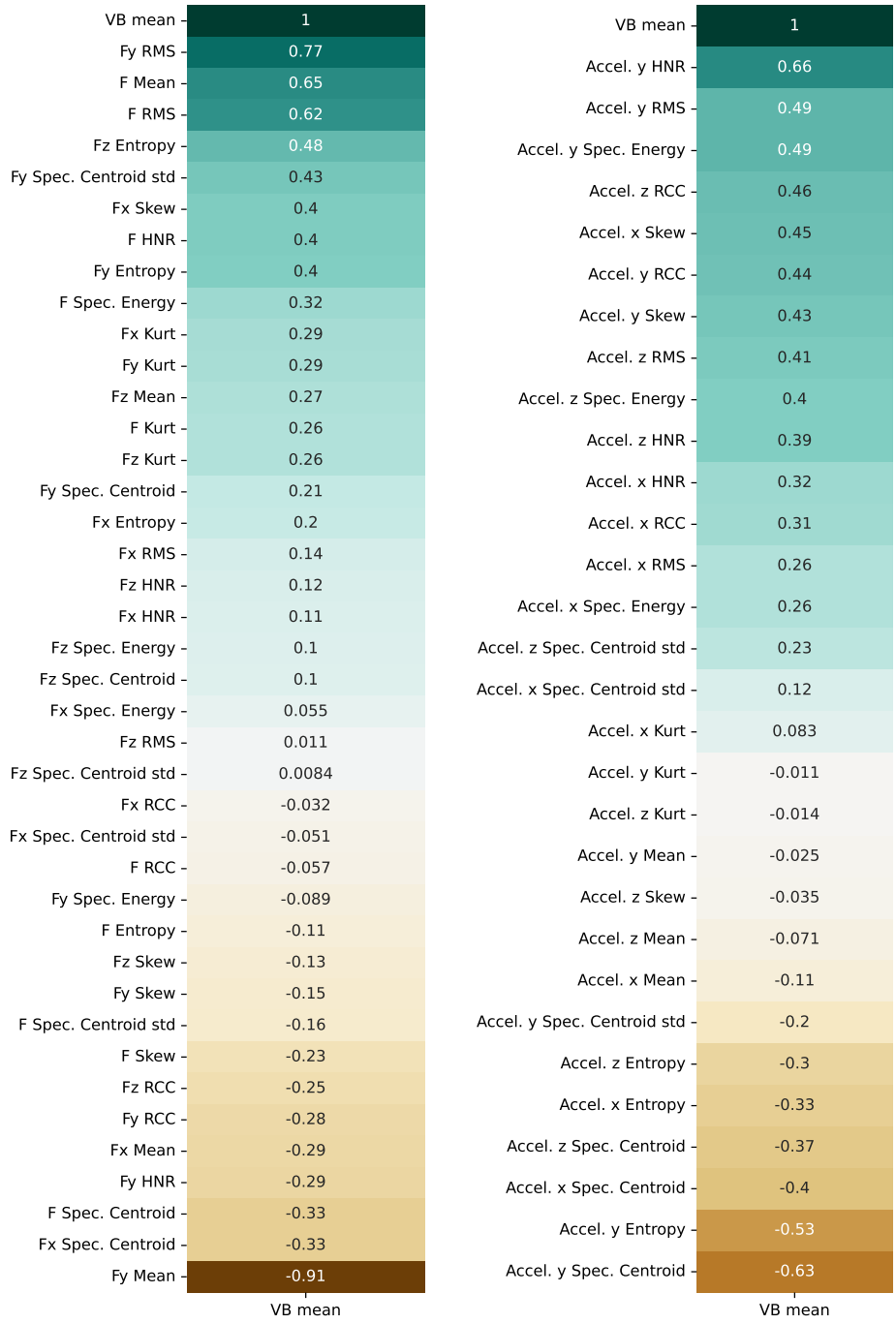
More specifically, Figure 14.22a presents the correlation of cutting forces along the three spatial directions, as well as the resultant force F. Figure 14.22b follows the same approach for acceleration signals. Figure 14.22c shows the evolution of sound-based indicators. For this sensor, no directional component is considered, as sound is captured omnidirectionally. Finally, Figure 14.22d illustrates the correlation of machine axis power, including directions x, y, z, and S, where S corresponds to the spindle axis.

Across all figures, the indicators most strongly correlated with tool wear consistently appear along the cutting direction or the spindle axis.

In addition, the cumulative machining time is correlated at 0.91 % with the wear.

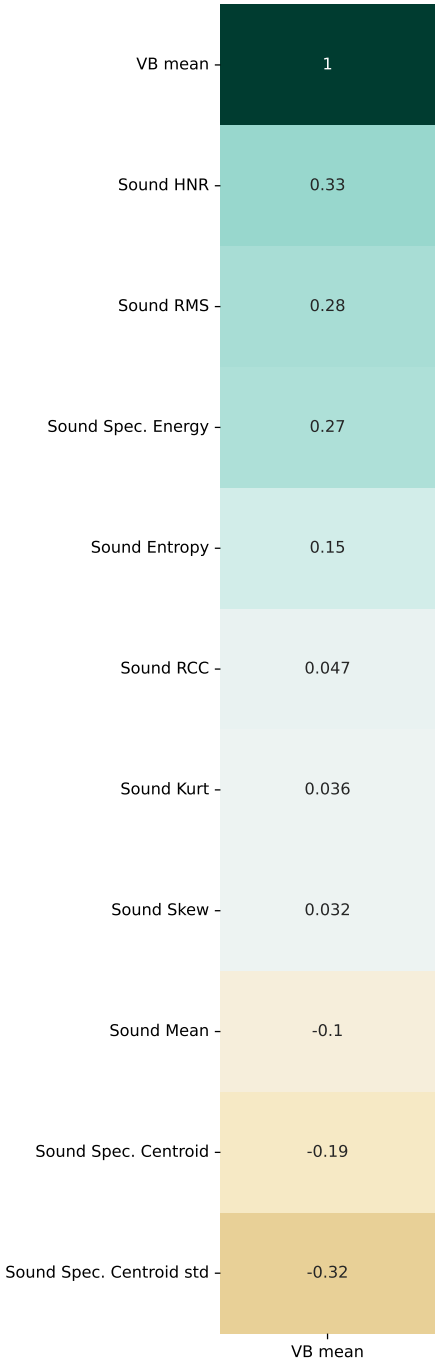
14.9 Features Selection

In the previous sections, 10 indicators were identified: 4 from the time domain and 6 from the frequency domain. These indicators are computed for each measured signal. For example, in the case of cutting forces, all ten indicators are calculated for the x, y, and z components of the force. As shown in Section 14.8, not all indicators exhibit a strong correlation with tool wear. It is therefore necessary to select the most relevant indicators that can be used by the various models to establish a relationship between the indicators and tool wear. Given the diversity of information in the database, feature selection was performed multiple times, each applied to different subsets of data. The first subset corresponds to the features extracted from each individual sensor. This allows the identification of the indicators most correlated with tool wear for each measured phenomenon. The process is repeated for each type of signal: cutting forces, acceleration, sound, and machine power resulting in four separate feature selection rounds. Feature

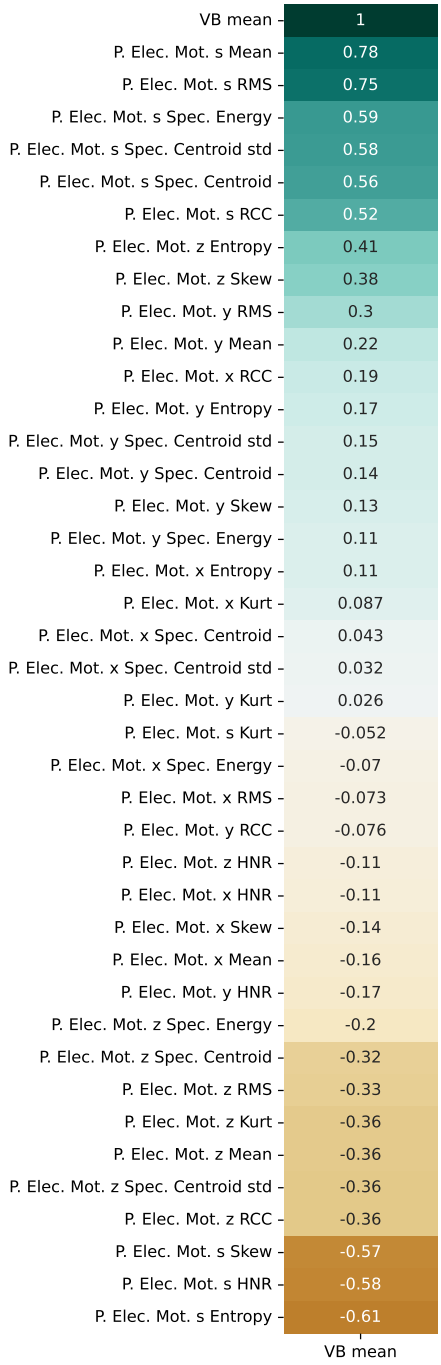


(a) Correlation of the cutting forces **(b)** Correlation of the accelerations

Figure 14.22: Correlation analysis of the different measured signals with respect to tool wear.



(c) Correlation of the acoustic pressure



(d) Correlation of the machine power

Figure 14.22: *...Continued*

selection is then performed again, this time by combining multiple types of signals. This approach enables a comparison of the impact of data fusion in the following sections. The feature selection process is the same for both subsets and is detailed hereafter.

The selection procedure follows several steps (Figure 14.23). The first step aims to identify the optimal number of features, beyond which adding more information no longer improves model performance. This ensures that the selected features are both non-redundant and strongly related to tool wear. Reducing the feature set is motivated by the fact that studies have shown that neural networks often perform better when provided with fewer, less redundant inputs, as an excessive number of irrelevant or overlapping features can degrade performance [180]. To estimate this minimal number of features, the Recursive Feature Elimination with Cross-Validation (RFECV) method from Scikit-Learn is applied [110]. This approach iteratively evaluates the performance of an estimator while removing the least important features, thereby identifying the optimal feature size. In this study, the estimator used is a Random Forest Regressor [110], which is well-suited for capturing non-linear relationships between features and tool wear. RFECV provides a performance curve as a function of the number of features, allowing us to select the optimal feature count (see Figure 14.23) .

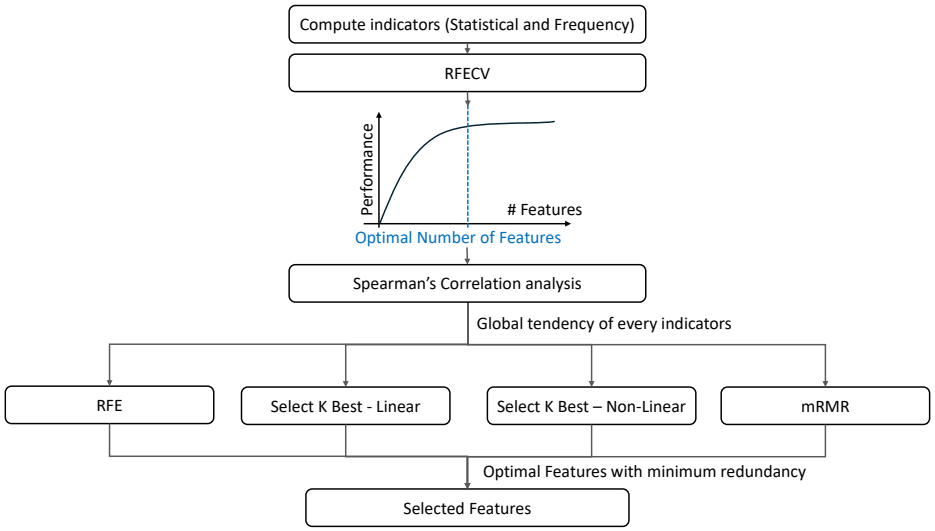


Figure 14.23: Feature selection process. First, the RFECV method is applied to determine the optimal number of features. Once this number is known, multiple selection methods are used to identify the most relevant feature combinations.

Once the optimal number of features is identified, several methods are used to determine the best combination of features within that limit. First, Recursive Feature Elimination (RFE) is applied again, this time with the knowledge of the optimal feature count to select the most relevant subset. To validate and complement these results, the "Select K Best" method is also employed [110]. This technique ranks features based on statistical tests, using both linear and non-linear

scoring functions [110]. Finally, the Minimum Redundancy Maximum Relevance (mRMR) method is applied to further evaluate each feature's importance and its redundancy with others [110]. Combining these approaches, along with prior knowledge of feature correlations, enables the selection of an optimal and minimal set of features to be used as inputs for the models. This process ensures that the chosen features are both highly informative and minimally redundant, with the objective of improving model efficiency and accuracy.

This feature selection approach was applied individually to data from each sensor. As a result, Table 14.10 summarizes the selected features for each sensor. In addition to analysing each sensor separately, a sensor fusion strategy is also adopted. This approach leverages information from multiple sensors, in order to observe if together they can provide better performance than any single sensor alone. Based on the feature analysis, three fusion families were identified:

Fusion 1 combines information from cutting forces, accelerations, and machine power. Sound information was excluded, as it showed poor correlation with tool wear.

Fusion 2 combines information from cutting forces and accelerations. This combination is motivated by the distinct nature of these two types of data, which are expected to provide complementary insights into tool wear.

Fusion 3 combines information from accelerations and machine power. This combination is tested because both signals are captured using sensors that are minimally intrusive to the cutting process.

This feature selection process therefore identifies seven combinations of data (see Table 14.10). These combinations will be used in the indirect approaches related to milling (see Chapters 15 and 16). In these chapters, the data grouped within each combination will be used to train the models, allowing for a comparison of performance based on the different input information sets. In addition to these seven groups, the following cutting conditions may optionally be included: Cumulative machining time, representing the age of the cutting tool, v_c and f_z, representing the cutting parameters. In total, 14 combinations will be tested for the indirect approaches: the seven identified in Table 14.10, each evaluated with and without the cutting conditions.

14.10 Training and Testing Dataset

In the turning database (see Chapter 6), the training, validation, and testing datasets were separated based on cutting conditions. This means that the test trajectories corresponded to degradation sequences involving variations in cutting parameters.

For the milling database, such a separation is not necessary, as it already includes sufficient variations in cutting speed to observe changes in the measured signals. Consequently, it is not possible to identify specific trajectories that would be more suitable than others. Therefore, the entire dataset is used for testing, rather than selecting specific trajectories.

Table 14.10: Feature selection sets by category. Note that the notation are expressed in the sensor frame.

Category	Channel	Mean	RMS	Skew.	Kurt.	Entropy	HNR	Centroid	Centroid Std.	Spec. Energy	RCC	Nb. features
Cutting Forces												
Force	x	✓										1
	y	✓		✓								2
	z	✓						✓				2
Accelerations												
Accelerometer	x			✓				✓		✓	✓	4
	y		✓	✓		✓	✓	✓				5
	z		✓							✓	✓	3
Acoustic Pressure												
Sound	–	✓	✓	✓		✓	✓	✓	✓	✓		8
Machine Power												
Power	S	✓	✓	✓		✓	✓	✓	✓	✓	✓	9
	y	✓	✓							✓		3
	z					✓						1
Fusion 1: Forces – Accelerations – Power												
Force	y	✓	✓									2
Force	z	✓				✓						2
Accelerometer	y						✓					1
Power	S	✓	✓				✓					3
Fusion 2: Forces – Accelerations												
Force	x	✓										1
Force	y	✓	✓									2
Force	z	✓				✓						2
Accelerometer	y							✓				1
Accelerometer	z	✓								✓		2
Fusion 3: Accelerations – Power												
Accelerometer	x			✓		✓						2
Accelerometer	y		✓	✓		✓	✓	✓		✓		6
Power	S	✓	✓							✓		3
Power	y	✓										1
Power	z	✓										1

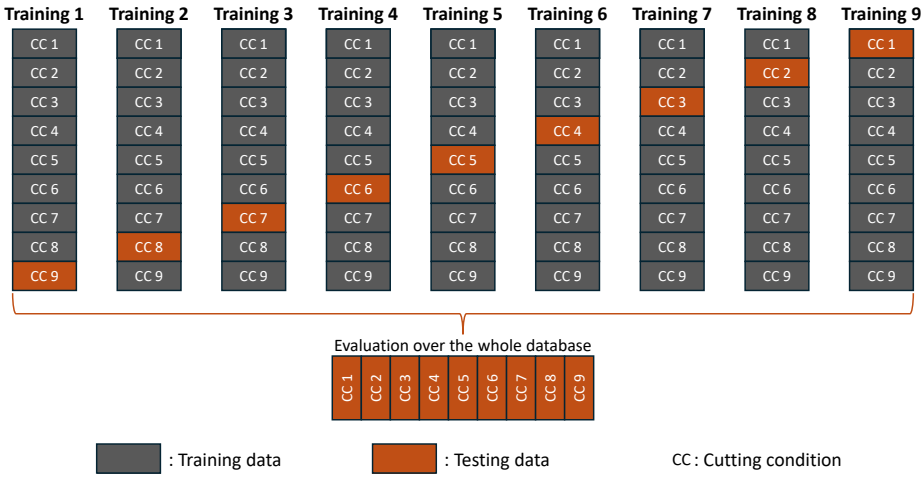


Figure 14.24: Division of the dataset into training and testing subsets based on the 9 distinct cutting conditions. The learning process is repeated 9 times, with each iteration using a different cutting condition as the test set. This ensures that the model is evaluated across all conditions, providing a complete assessment of its performance.

Since the full dataset is used for testing, the model must be trained multiple times, each time excluding a portion of the data. The selected approach involves grouping data by cutting conditions. For each iteration, the model is trained on all but one cutting condition and tested on the excluded condition. This process is repeated 9 times (one for each cutting condition), effectively evaluating the model's ability to generalize to unseen cutting scenarios (Figure 14.24).

14.11 Conclusions

This chapter presents the evolution of various signals and indicators measured during the machining experiments. It includes a detailed description of tool wear progression and its impact on the quality of the machined surface.

The analysis of both temporal and frequency-domain indicators across different sensors reveals distinct sensitivities to tool wear. For signals such as the cutting force and spindle power, temporal indicators such as the mean and RMS values exhibit strong correlations with wear. For these two informations, the temporal evolution of indicators are more strongly correlated to the wear than the frequency indicator. Nonetheless, spectral entropy in the spindle power signal consistently decreases over time, highlighting its potential as a wear-sensitive feature. For the acceleration signal, frequency-based indicators such as the HNR show a good correlation with the wear. The frequency indicators are better suited for this signal than the temporal one with a stronger correlation to wear. In contrast, sound-based indicators display weaker correlations for both temporal and frequency based indicators, likely due to external factors such as the equipment position within

the machine. These findings underscore the importance of selecting sensor-specific indicators.

The correlation of each indicator is also presented. Additionally, the feature selection process used by the various models in the remainder of this section is described. This process aims to identify the optimal number and most relevant indicators for training future models. Feature selection was designed to enable the evaluation of each sensor individually, and then in combination through sensor fusion, thereby exploring the benefits of integrating multiple physical phenomena to improve performance.

Finally, the dataset split into training and testing sets is also presented. This split groups together similar cutting conditions and trains the models on all but one condition. The process is repeated iteratively to assess model performance across the entire dataset.

Chapter 15

Monitoring the Cutting Tool Degradation from Multiple Sensors Signals in Milling

Indirect Monitoring with Regression

This chapter presents the application to milling of the methods developed for turning in Chapter 7.

15.1 Introduction

This chapter applies the methods developed for turning to the milling database. As a reminder, Chapter 7 used artificial neural networks to monitor the degradation of the cutting tool. To enhance this monitoring, a bootstrap method was introduced to estimate confidence intervals around the monitoring values. Additionally, an explainable boosting machine was employed to provide interpretability to the results of the monitoring model.

This chapter therefore focuses on applying the previously developed methods. Since the milling database is larger and includes more sensors than the turning database, some adaptations are necessary. The chapter is organized as follows (see Figure 15.1):

- Section 15.2: This section presents the results obtained with the ANN approach. More specifically:
  - Section 15.2.1: This section presents the optimization process used to identify the most suitable neural network architecture for monitoring the cutting tool, based on the characteristics of the milling database.
  - Section 15.2.2: This section presents the results obtained using the optimal network architecture for various input groups. It discusses the

use of each sensor as an individual input and explores the potential of sensor fusion.

- Section 15.3: This section introduces confidence interval estimation for the monitoring process. While the bootstrap method was used for the turning database, computational constraints prevent its reuse for milling. Therefore, an alternative probabilistic technique is proposed, along with a comparative discussion to help determine which confidence interval estimation method is more appropriate in different scenarios.
- Section 15.4: This section presents the monitoring performance of the explainable boosting machine.
- Section 15.5: This section concludes the chapter.

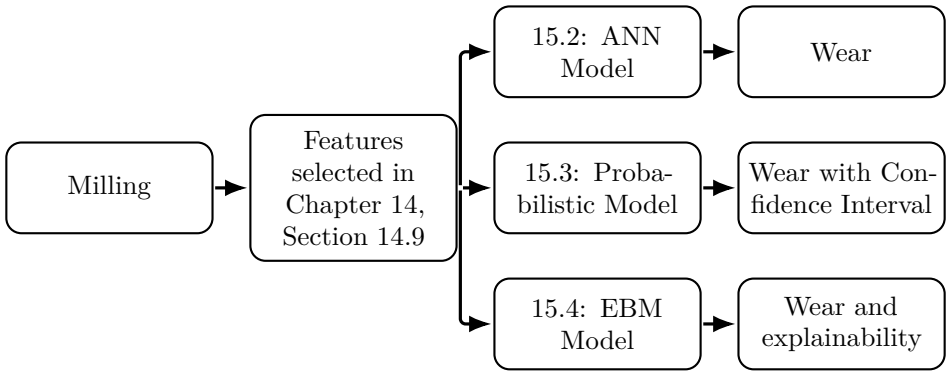


Figure 15.1: Summary of the approaches developed in this chapter.

15.2 Monitoring Cutting Tool Wear with ANN

15.2.1 ANN Model Selection

In Chapter 7, the model was selected through a trial-and-error approach. This method was feasible due to the small size of the database, which allowed for manual experimentation. However, for the milling database, which is significantly larger, this trial-and-error process must be automated and optimized. Therefore, to optimize the model architecture and its hyperparameters, the Optuna optimization framework is used [181]. This optimizer employs Bayesian optimization techniques to efficiently explore the hyperparameter space and identify its optimal combination.

Figure 15.2 illustrates the optimization scheme used to identify the best network architecture. The optimization loop begins with a hyperparameter suggestion. Table 15.1 summarizes the parameter ranges used during this process. These parameters were selected to cover a wide spectrum of architectures, ranging from deep networks with many neurons in the hidden layers to shallower networks with fewer neurons. Next, a model is created based on these hyperparameters and trained. Once trained, the model's performance is saved and compared with previously

tested models. If the tested model achieves better results than the previous ones, the optimization process adjusts the hyperparameters in the direction that led to the improvement. This optimization also helps rank the importance of each hyperparameter by observing how performance varies when they are modified. The process is repeated for a given number of trials. This number is set to 75, allowing the optimizer to test up to 75 different combinations of hyperparameters. This value is chosen to avoid unnecessary computations while still allowing the optimizer sufficient time to explore multiple combinations of hyperparameters. All ANN models are trained with a maximum of 300 epochs. Standard overfitting prevention techniques are also implemented to avoid misleading results.

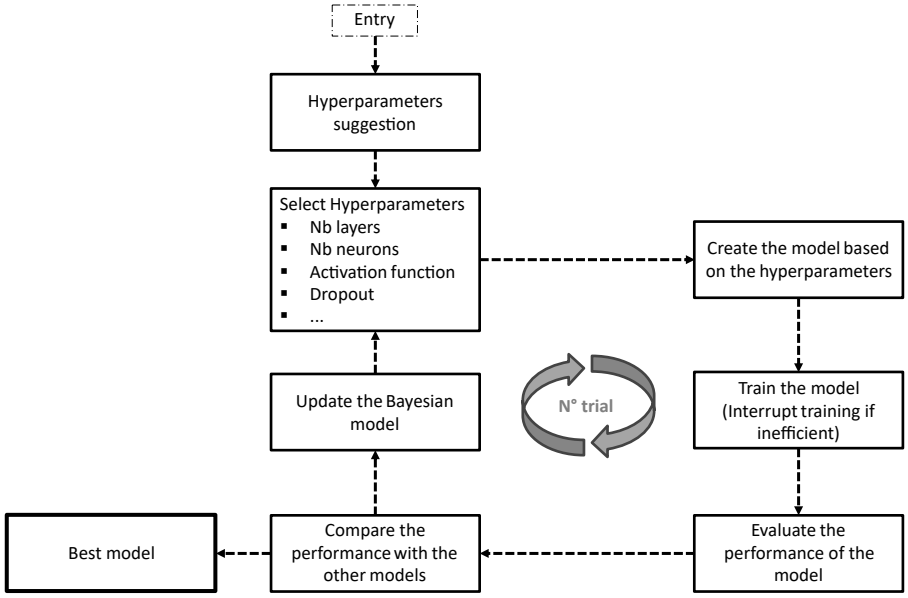


Figure 15.2: Architecture of the optimization scheme used to select the most suitable family of neural network models.

Table 15.1: List of hyperparameters optimized using the Optuna optimizer. For each hyperparameter, the tested value range is presented along with the optimal result obtained. The selected value within this optimal range is also indicated in brackets.

Hyperparameters	**Range**	**Optimal (selected)**
Number of layers	1 to 30	10 to 20 (15)
Number of neurons	5 to 80	40 to 80 (50)
Activation fct.	ReLU, Tanh and Sigmoid	(ReLu)
Dropout rate	0, 0.2 or 0.5	(0.0)

The optimization process is conducted using various group of input data (see Chapter 14 Section 14.9) and evaluated across the entire dataset. From this optimization, the goal is to find a family of neural networks delivering the best

performance in a robust manner. That is, networks sharing similar hyperparameters while maintaining comparable optimal levels of performance. This indicates that these kind of architecture families are robust, as small variations in the network do not significantly affect the results. Consequently, these architectures appear particularly well-suited for monitoring tool wear based on the characteristics of the given dataset.

The optimization process reveals that multiple architectures are capable of achieving comparable performance (Table 15.1). Consistently, the ReLU activation function emerges as the most effective choice. The optimal dropout rate is found to be zero, indicating that the network is not significantly prone to overfitting. In general, architectures with 10 to 20 layers and approximately 40 to 80 neurons per layer tend to perform similarly. Based on these findings, the architecture selected for the remainder of this chapter consists of 15 hidden layers, each with 50 neurons, using ReLU activation and no dropout (see Table 15.1 and Figure 15.3). The selected model aims to be as small as possible while maintaining high robustness. Accordingly, the chosen hyperparameters are selected either near the center of the optimal range (e.g., for the number of layers) or slightly biased toward the lower end of that range (e.g., for the number of neurons).

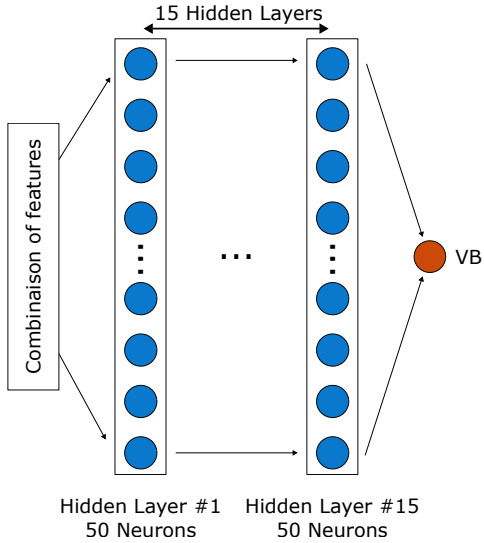


Figure 15.3: Selected architecture of the monitoring neural network for the milling database.

15.2.2 Monitoring Results with ANN

To compare the performance of different feature combinations, the neural network described in Section 15.2.1 is trained on all possible feature sets. Table 15.2 summarizes the mean R^2 scores and mean absolute error obtained across the entire dataset for each input configuration, both with and without the inclusion of cutting conditions. These performance indicators are evaluated on the whole database

through 9 independent runs of the model to evaluate the complete database (see. Chapter 14 Section 14.10). To compute the deviation around the value, this process is repeated 10 times. Each entry of the table therefore correspond to 90 different runs of the model. From Table 15.2, multiple observations can be made:

- For all input features, incorporating cutting conditions consistently improves tool wear estimation. The model uses machining time and cutting conditions as a reference to learn the wear progression, making it a key variable for the neural network's monitoring performance.
- When indicators are grouped by sensor and cutting conditions are excluded, cutting forces show the strongest performance to monitor tool wear, followed by accelerations and power indicators. Finally, the microphone yields the weakest monitoring performance.
- For Sensor Fusion 1, which combines cutting force, acceleration, and power indicators, the results are less accurate than using cutting force alone. A similar trend is observed with Sensor Fusion 2, which combines cutting force and acceleration. However, in this case, the performance gap is smaller compared to using only cutting force. These results suggest that adding additional sensor data to cutting force does not improve model performance and may even introduce "noise" that negatively impacts prediction accuracy.
- Sensor Fusion 3, which combines acceleration and machine power indicators, provides better performance than using each sensor individually. This suggests that when these sensors are used together, they offer complementary information that enhances tool condition monitoring.

Table 15.2 allows to compare the results based on the R^2 score and the mean absolute error. To observe how these scores affect monitoring performance, Figure 15.4 illustrates the different monitoring trajectories of the neural networks depending on their single sensor input configurations. In this figure, only the best-performing results for each sensor with and without cutting conditions are compared. Multiple observation can be made:

- **Cutting forces**. Figures 15.4a and 15.4b illustrate the monitoring performance using cutting force inputs alone, and cutting forces combined with cutting conditions, respectively. It is observed that degradation monitoring performs well in both cases. However, the inclusion of cutting conditions slightly improves the accuracy at the beginning of the degradation trajectory for this particular tool. Additionally, small steps can be seen in the monitoring results without cutting conditions. These are attributed to the fact that, during the final passes of the tool, the machining occurred along the edge of the workpiece, which may be slightly narrower than the rest. This leads to reduced cutting forces. These irregularities in the monitoring are therefore linked to the tool operating on this specific part of the workpiece. Such effects are smoothed out when cutting conditions are included in the model.
- **Accelerations**. Figures 15.4c and 15.4d present the monitoring performance using acceleration signals, without and with the inclusion of cutting conditions,

Table 15.2: R^2 score and mean absolute error for each combination of input features (as described in Table 14.10). Each value represents the mean R^2 score or the mean absolute error and its standard deviation over 10 independent runs of the same model. The performance indicators are computed on the whole dataset.

Input features	Without cutting conditions	With cutting conditions
R^2 Score		
Individual Sensor		
Cutting Forces	$91.85 \pm 0.56\,\%$	$92.12 \pm 1.05\,\%$
Accelerations	$66.96 \pm 2.63\,\%$	$86.59 \pm 1.98\,\%$
Acoustic Pressure	$37.36 \pm 2.84\,\%$	$80.76 \pm 2.06\,\%$
Machine Power	$60.52 \pm 2.61\,\%$	$90.84 \pm 1.29\,\%$
Sensor Fusion		
Fusion1	$80.75 \pm 2.47\,\%$	$83.77 \pm 1.11\,\%$
Fusion2	$86.19 \pm 1.03\,\%$	$90.27 \pm 0.76\,\%$
Fusion3	$71.95 \pm 1.21\,\%$	$89.05 \pm 1.47\,\%$
Mean Absolute Error		
Individual Sensor		
Cutting Forces	$16.66 \pm 0.54\,\mu m$	$17.49 \pm 1.35\,\mu m$
Accelerations	$33.43 \pm 1.02\,\mu m$	$22.61 \pm 1.82\,\mu m$
Acoustic Pressure	$53.59 \pm 1.17\,\mu m$	$26.86 \pm 1.71\,\mu m$
Machine Power	$46.85 \pm 2.55\,\mu m$	$19.96 \pm 1.63\,\mu m$
Sensor Fusion		
Fusion1	$29.19 \pm 2.55\,\mu m$	$24.81 \pm 1.20\,\mu m$
Fusion2	$21.86 \pm 0.89\,\mu m$	$18.60 \pm 0.78\,\mu m$
Fusion3	$30.66 \pm 0.84\,\mu m$	$20.62 \pm 1.47\,\mu m$

respectively. It is observed that the monitoring without cutting conditions is noisier and exhibits more oscillations compared to the version with cutting conditions. However, despite this noise, the monitoring, particularly towards the end of tool life, remains reasonably accurate. The addition of cutting conditions helps smooth the degradation curve, but the monitoring consistently underestimates the actual tool wear.

- **Acoustic pressure**. Figures 15.4e and 15.4f show the monitoring performance using the sound signal as input. As anticipated from the results in Table 15.2, the monitoring performance without cutting conditions is poor. However, the inclusion of cutting conditions helps smooth the degradation curve and improves the monitoring performance. Despite this improvement, this sensors appear less reliable compared to the other approaches.

- **Machine's power**. Figures 15.4g and 15.4h illustrate the monitoring per-

formance using machine power as input. Without the inclusion of cutting conditions, the monitoring fails to capture the initial wear progression—the tool wear remains flat during the early stages and only begins to evolve after approximately six minutes. Additionally, the monitoring signal is highly oscillatory and tends to underestimate the actual tool wear. When cutting conditions are incorporated, the monitoring performance improves significantly, showing an excellent fit throughout most of the tool's life.

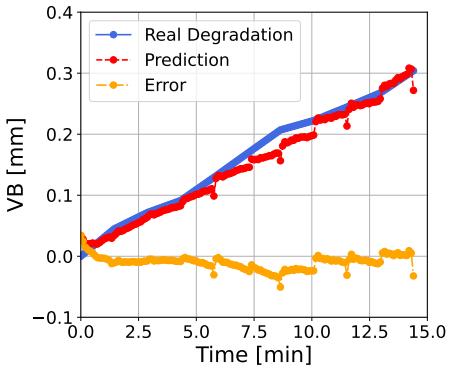
Figure 15.5 illustrates the different monitoring trajectories obtained with sensor fusion:

- **Fusion 1**. Figures 15.5a and 15.5b shows the performance using a combination of cutting force, acceleration, and machine power indicators. As anticipated from the results in Table 15.2, the monitoring performance is generally good. In the case without cutting conditions, there is a slight overestimation of wear at the beginning of the tool life, but the rest of the trajectory is accurately tracked. When cutting conditions are included, the model tends to consistently underestimate the wear, resulting in less accurate monitoring.

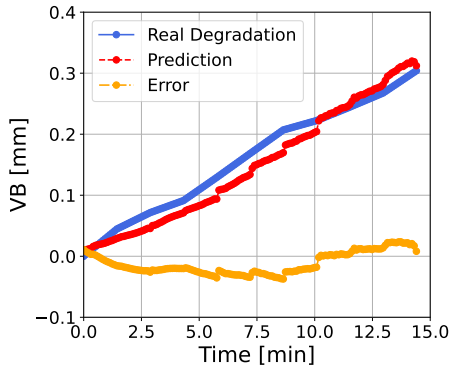
- **Fusion 2**. Figures 15.5c and 15.5d presents the performance using a combination of cutting force and acceleration indicators. In both scenarios (with and without cutting conditions), the monitoring closely follows the actual degradation, demonstrating reliable performance.

- **Fusion 3**. Figures 15.5e and 15.5f illustrates the performance using a combination of acceleration and machine power indicators. In the case without cutting conditions, the degradation trajectory closely resembles that observed in Figure 15.4c, but with slightly less noise. When cutting conditions are included, the model underestimates the actual wear progression.

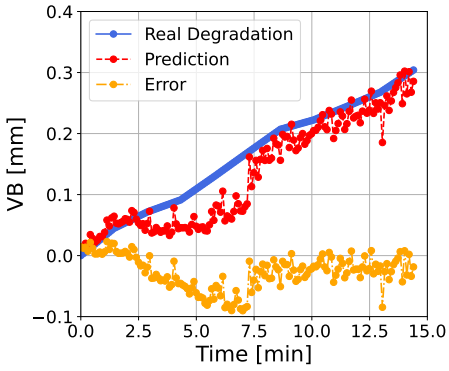
In conclusion, incorporating cutting conditions consistently leads to smoother and more accurate monitoring across all combination of inputs, underscoring their importance in robust tool wear estimation. Among single-sensor approaches, cutting force indicators offer the most reliable tracking of tool degradation, regardless of whether cutting conditions are included. While accelerometer and machine power indicators also provide useful insights, their accuracy is comparatively lower. However, both benefit significantly from sensor fusion, as combining them enhances the overall monitoring performance by leveraging complementary information. It should also be noted that, in this case, the monitoring system tends to underestimate tool degradation, contrary to what is observed in turning operations. This is most likely due to the specific characteristics of the milling database, which favours learning patterns that lean toward underestimation. To encourage a more conservative (and thus safer) approach, additional penalties could be introduced during training to promote overestimation of wear. However, this aspect was not considered in this chapter.



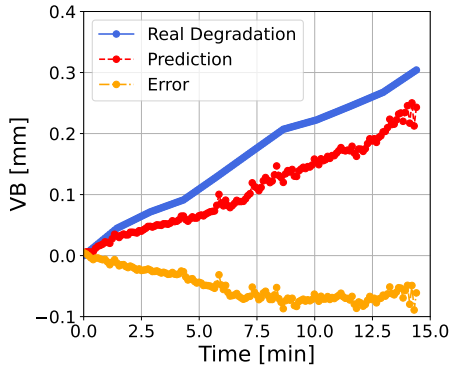
(a) Cutting Forces



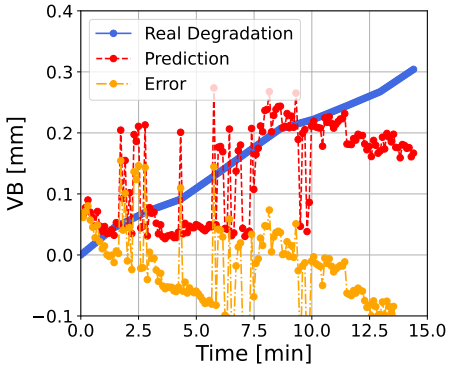
(b) Cutting Forces with cutting conditions



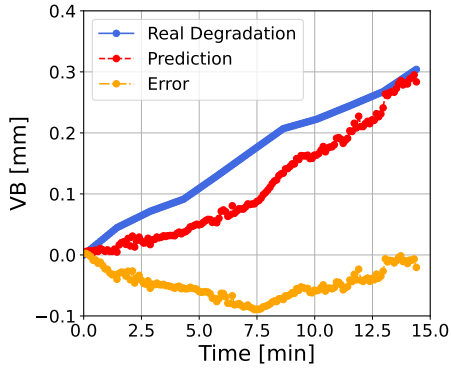
(c) Accelerations



(d) Accelerations with cutting conditions

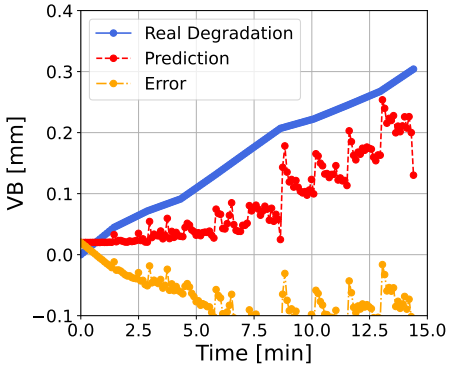


(e) Acoustic Pressure

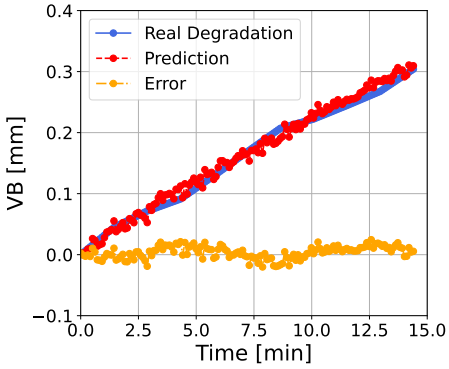


(f) Acoustic Pressure with cutting conditions

Figure 15.4: Comparison of different inputs on the monitoring capabilities for each sensor. The monitored tool is the tool 0020.



(g) Machine power



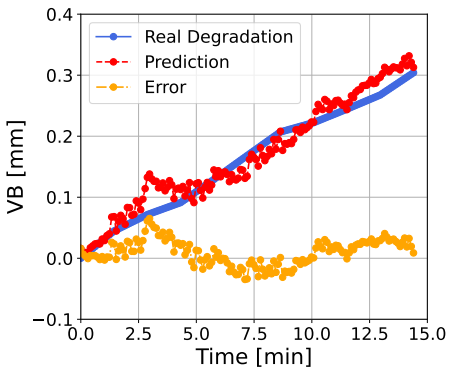
(h) Machine power with cutting conditions

Figure 15.4: *...Continued*

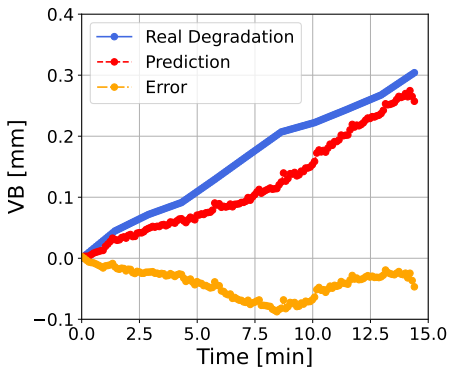
15.3 Confidence Interval Estimation with Probabilistic Networks

In Chapter 7, a Bootstrap method was used to estimate the confidence interval around the wear prediction. This method successfully provided both an accurate estimation of the wear and a corresponding confidence interval. Bootstrap was feasible in that context because the database and models were relatively small. In the case of milling, however, the larger size of both the dataset and the models makes this method less scalable. To recall, the turning dataset contains 192 passes, whereas the milling dataset includes 3,688 passes, over 19 times more, along with a greater quantity of measured data. The milling model is also significantly larger, consisting of 15 layers with 50 neurons each, compared to just 2 layers with 6 neurons in the turning model. As a result the computational resources required are significantly higher, and parallelizing such large models is more difficult on a single machine. As a matter of facts, in the previous section, Table 15.2 was obtained in around 2 days of computing on a RTX 4080 GPU (48.7 TFLOPS, see Chapter 2) without any possibility of parallelization. In this table, the standard deviation is estimated from 10 independent runs of the same model. Applying the bootstrap method, which typically requires around 100 runs, would result in an estimated computation time of 20 days. In contrast, for turning, 100 models were trained on an Intel i7-9750H CPU (0.337 TFLOPS, see Chapter 2), and the results were obtained in just 10 minutes. This highlights the scalability challenge of bootstrap in this context, especially when parallelization is limited on a single machine. While these limitations do not make bootstrap entirely unusable for milling, this chapter focuses on exploring an alternative approach that do not present these scalability issues.

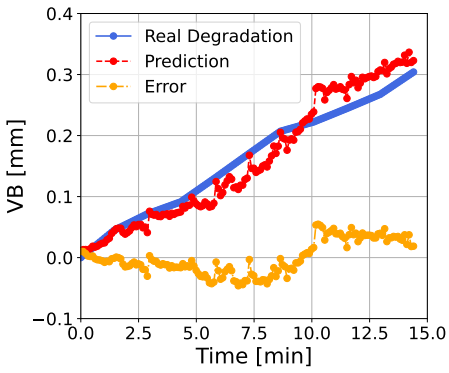
Instead of using multiple neural networks to estimate the distribution of tool wear, the proposed approach is to directly predict the mean (μ) and standard deviation (σ) of the wear distribution as outputs of a single model. This means the model has two outputs instead of one. These models are called probabilistic neural



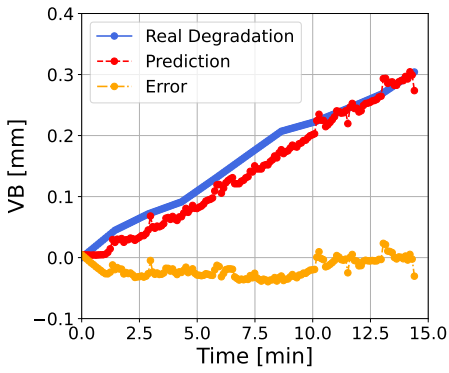
(a) Sensor fusion1



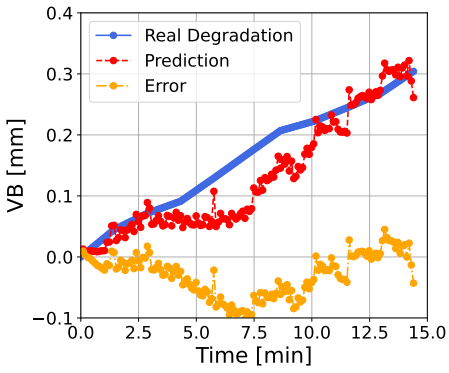
(b) Sensor fusion1 with cutting conditions



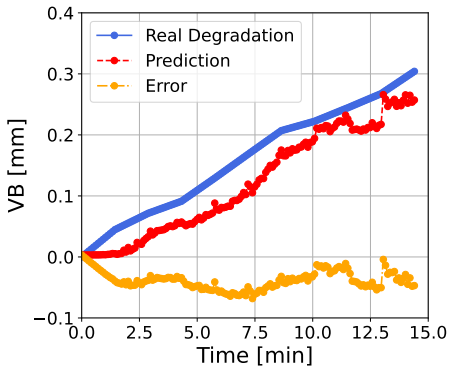
(c) Sensor fusion2



(d) Sensor fusion2 with cutting conditions



(e) Sensor fusion3



(f) Sensor fusion3 with cutting conditions

Figure 15.5: Monitoring capabilities with sensor fusion. The monitored tool is the tool 0020.

networks [182]. To train such models, the loss function must be adapted to reflect the probabilistic nature of the output. Specifically, the Negative Log-Likelihood loss ($\mathcal{L}_{\text{NLL}}$), defined in Equation (15.1), is used [183]. In this formulation, y represents the true wear value, while μ and σ are the predicted parameters of the normal distribution.

$$\mathcal{L}_{\text{NLL}}(y, \mu, \sigma) = \frac{1}{2}\log(2\pi\sigma^2) + \frac{(y-\mu)^2}{2\sigma^2} \tag{15.1}$$

The model architecture remains the same as previously developed, with one key difference: instead of producing a single output, it now generates two parameters, μ and σ, to estimate the confidence interval (Figure 15.6). Given the shared architecture, the network's performance closely mirrors the results presented in Section 15.2.2. Therefore, similar conclusions can be drawn regarding the quality of the monitoring based on the model's predicted mean. The primary advantage of this approach does not lie in improved monitoring accuracy, but rather in the added value of providing a confidence interval.

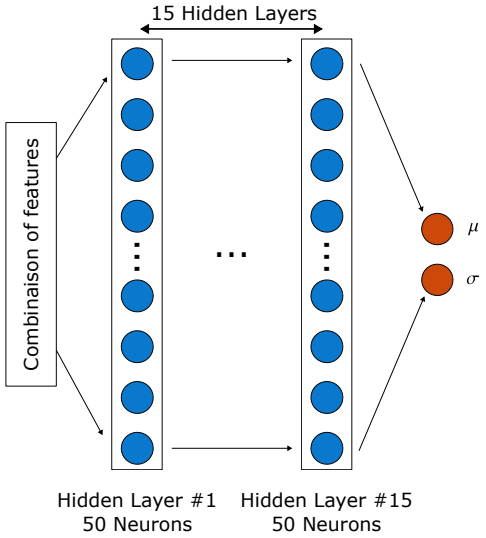
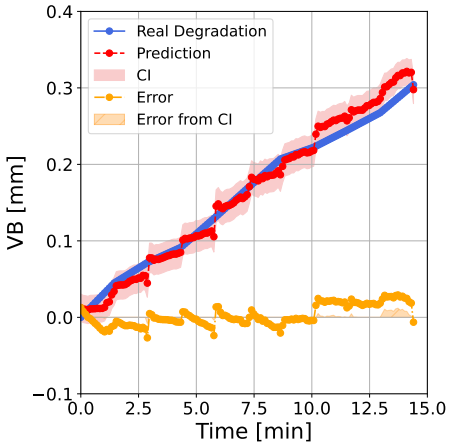


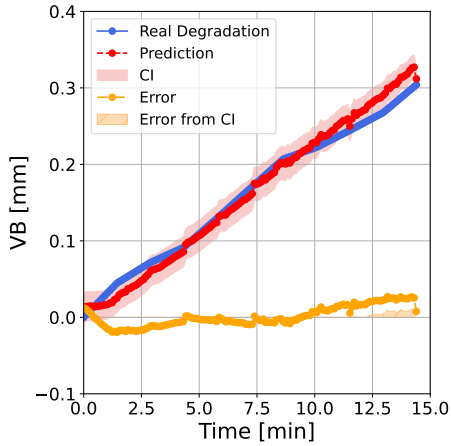
Figure 15.6: Selected architecture of the monitoring probabilistic neural network for the milling database. Compared to the Figure 15.3, this network have 2 outputs: one for μ and one for σ.

Figure 15.7 presents an example of a degradation trajectory predicted by the probabilistic neural network with the cutting force indicators (in all direction) as inputs. Surrounding the prediction, the confidence interval (CI) represents a 95 % confidence level. In addition to the standard prediction error shown in orange, an error relative to the confidence interval is also plotted. Figures 15.7a and 15.7b show similar trajectories using cutting force inputs, with and without cutting conditions, respectively. The predicted mean closely aligns with that of a standard (non-probabilistic) network. In both cases, the confidence interval remains relatively narrow and effectively encompasses the real degradation curve. This means that even when the predicted mean slightly deviates from the measured wear, the confidence interval still captures the real degradation trend.

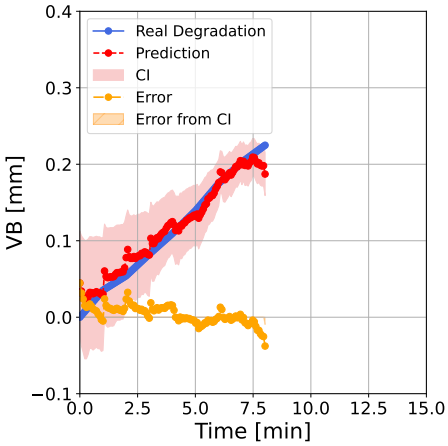
In rare cases, such as illustrated in Figure 15.7c, the confidence interval at the beginning of the tool's life can be significantly wide. This uncertainty is mitigated by incorporating cutting conditions, which help reduce the width of the confidence interval, particularly during the early stages of tool degradation. This further highlights the benefit of including cutting conditions as input features in the network.



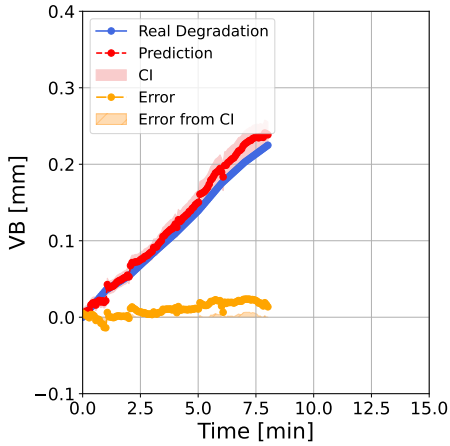
(a) Cutting Forces – Tool: 0020



(b) Cutting Forces with cutting conditions – Tool: 0020



(c) Cutting Forces – Tool: 0004



(d) Cutting Forces with cutting conditions – Tool: 0004

Figure 15.7: Comparison of different monitored trajectory with a probabilistic network.

15.4 Explainability of the Monitoring Results with EBMs

In Chapter 7, Explainable Boosting Machines are used to monitor the tool wear while providing local and global explainability. The same approach is used in this section.

15.4.1 EBMs Hyperparameters Selection

To obtain the best monitoring results, the hyperparameter of the model must first be optimized to the milling database. The same grid search optimisation approach as in Chapter 7 is applied. Table 15.3 shows the recommended hyperparameter by [148] and the selected hyperparameter to perform the monitoring.

Table 15.3: Hyperparameter of the EBM model with the recommended value by the InterpretML library [148] and the value selected through a grid search algorithm.

Hyperparameter	Recommended [148]	Selected
Max leaves	2 to 3	2
Learning rate	0.0025 to 0.2	0.2
Smoothing round	75 to 500	500
Interaction	Data specific	5
EarlyStopping rounds	100 to 200	200
Inner bags	0 or 20	20
Outer bags	14	14

15.4.2 EBMs Results

Table 15.4 compares the performance of the EBM model based on the available input features. As before, these results are computed over the entire evaluation dataset by dividing it into 9 subsets and grouping the cutting conditions. Several observations can be made from this table, especially when compared to the results in Table 15.2:

- Similar trends to those observed in Table 15.2 can be seen. The same performance hierarchy and order of magnitude are maintained, although the R^2 scores obtained with EBM are generally lower, except for the sound sensor. Sensors that performed poorly with neural networks show comparable performance with EBM, suggesting that EBM was not able to extract significantly more information from the data than the ANN.
- Including cutting conditions in the input data improves performance. This improvement is slightly more pronounced than with neural networks, with R^2 scores reaching around 93 %.

As in Chapter 7, EBM can be effectively applied to monitor tool degradation. However, this model presents the drawback of being less accurate than a dedicated ANN model. A comparison between Table 15.2 and Table 15.4 reveals a difference

of a few percentage points in the R^2 score and a variation of several microns in the mean absolute error. In practice, these slight differences have minimal impact on tool condition monitoring. Therefore, the accuracy achieved by the EBM model can be considered sufficient. Its main advantage lies in providing interpretability in addition to monitoring capabilities.

Table 15.4: R^2 score and mean absolute error for each combination of input features (as described in Table 14.10). The performance indicators are computed on the whole dataset.

Input features	Without cutting conditions	With cutting conditions
R^2 Score		
Individual Sensor		
Cutting Forces	89.28 %	93.31 %
Accelerations	62.06 %	87.86 %
Acoustic Pressure	45.92 %	87.66 %
Machine Power	51.16 %	94.62 %
Sensor Fusion		
Fusion1	80.72 %	91.08 %
Fusion2	84.69 %	92.70 %
Fusion3	67.55 %	91.81 %
Mean Absolute Error		
Individual Sensor		
Cutting Forces	18.92 µm	15.52 µm
Accelerations	37.38 µm	20.62 µm
Acoustic Pressure	50.99 µm	21.65 µm
Machine Power	44.32 µm	14.16 µm
Sensor Fusion		
Fusion1	24.32 µm	18.10 µm
Fusion2	22.67 µm	15.95 µm
Fusion3	37.29 µm	16.81 µm

15.4.3 Global Explanation

The advantage of the EBM approach is its ability to provide explainability for tool state estimation.

To obtain the most relevant global explanation based on the available data, the model is trained using all indicators from all sensors. Figure 15.8 highlights the 40 most important features when all indicators are considered. This ranking is determined by the model: the higher the value, the more important the feature is considered. Interestingly, the model identifies cutting time as the most influential

feature. This finding is consistent with the results observed in this chapter, where adding cutting conditions to the input data improved performance. Since tool degradation is almost linear, time information is crucial because it helps the model determine the current stage of tool wear. After cutting time, the next most significant inputs are cutting forces, particularly in the machining direction, followed by accelerations, cutting speed, sound and spindle power. These results align with the correlation analysis presented in Chapter 14, Section 14.8. The global analysis also highlights feature interactions detected by the model. However, these interactions are not strongly correlated with tool wear and therefore do not appear in the Figure. In fact, the model did not identify any interaction with a higher correlation than the individual features themselves.

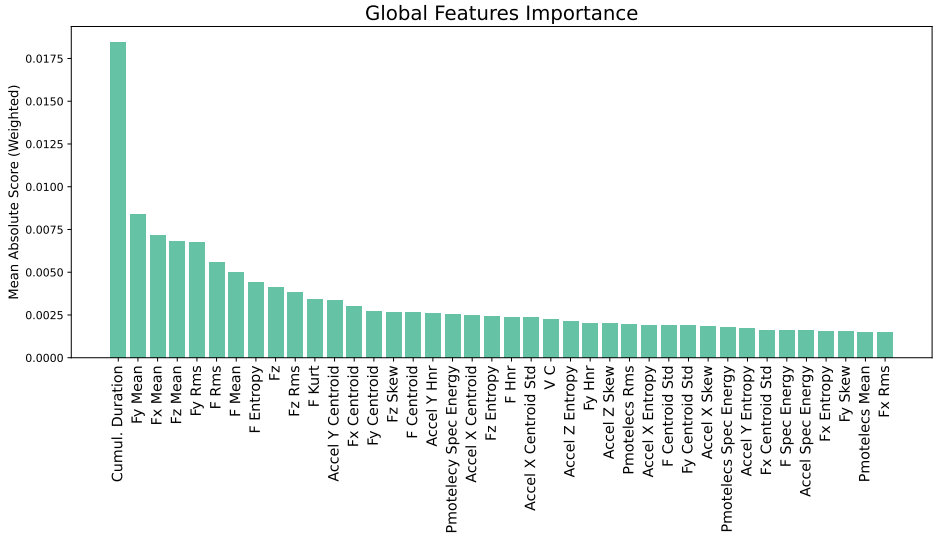


Figure 15.8: Global explanation showing the sorted average contribution of the 40 most important inputs ranked by the EBM model.

In addition to the previous ranking, it is possible to visualize the contribution of each feature. Figure 15.9 illustrates the global contribution of machining time as a function of its value. This confirms the earlier observation: the relationship between tool wear and time is essentially linear, meaning that wear increases steadily with machining time.

Figure 15.10 shows the global contribution of the F_y Mean feature, along with its distribution in the dataset. The plot reveals that as the cutting force increases, the contribution of this variable also rises, highlighting the direct relationship between higher cutting forces and the tool's condition.

15.5 Conclusions

This chapter applies the methods developed to monitor the tool degradation for turning in Chapter 7 to milling. Since the milling database is larger than the

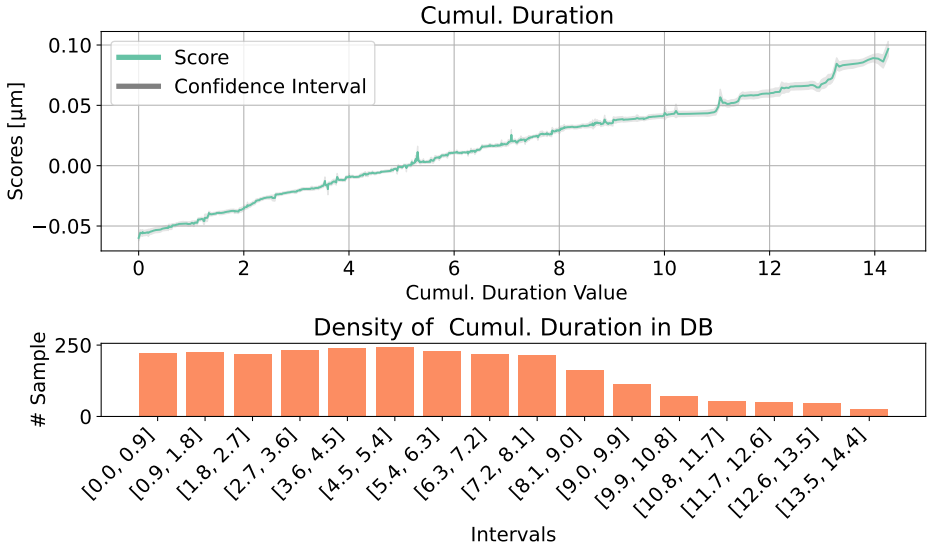


Figure 15.9: Global explanation of the selected features "cumulative duration" and its contribution to estimation of the tool wear with respect to the input value. The graph below shows density of each interval inside the database.

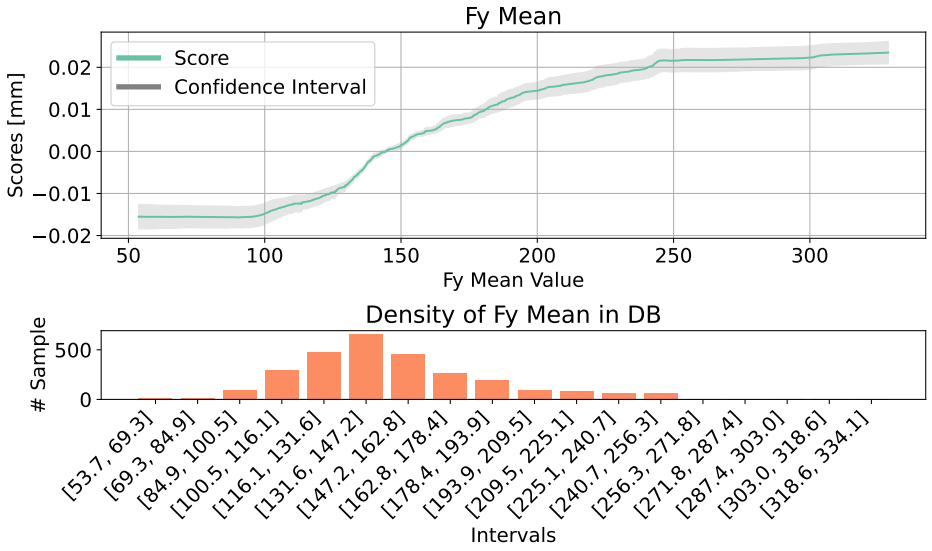


Figure 15.10: Global explanation of the selected features F_y Mean and its contribution to estimation of the tool wear with respect to the input value. The graph below shows density of each interval inside the database.

turning one, some approaches were adapted to improve performance. Several observations can be made:

- Instead of tuning the model through trial and error, hyperparameters were optimized using an automated optimization loop designed to explore the best configurations. This approach has the advantage of identifying robust hyperparameter families, enabling the selection of a reliable model architecture.
- The optimal neural network identified is then compared across different input combinations, incorporating various sensors and their fusion. A discussion on the performance of these features is provided, highlighting the trade-offs between different sensor configurations. For all sensors, it is evident that including cutting conditions and cutting time is important to improve the quality of the monitoring.
- To provide confidence intervals around neural network monitoring, a probabilistic network was employed. The bootstrap method developed in Chapter 7 was not applied to milling due to computational constraints and the size of the milling dataset. This new approach proved effective, achieving similar results to a standard neural network while additionally providing uncertainty estimates.
- Explainability of the results was also addressed using an Explainable Boosting Machine based method. After hyperparameter optimization, this model achieved monitoring performance comparable to the neural network. The global explanation highlights the importance of including cutting conditions and presents the hierarchy of sensor contributions discussed in previous sections.

In summary, this chapter successfully applies all methods previously developed for monitoring cutting tool degradation in turning to the milling process. The literature gaps identified, namely, the integration of confidence intervals and model interpretability, have both been addressed.

Chapter 16

Forecasting the Cutting Tool Degradation from Multiple Sensors Signals in Milling

Indirect Forecasting with Regression

This chapter presents the application to milling of the methods developed for turning in Chapter 8.

16.1 Introduction

In Chapter 8, a recurrent neural network-based approach was developed to forecast tool degradation up to 3 minutes into the future. The performance achieved in turning was very good, even allowing the model to track tool wear with an accuracy comparable to that of a monitoring-based approach.

The present chapter addresses the same objective, using recurrent neural networks to predict future tool degradation, but with the additional goal of testing various input combinations to evaluate how the choice of inputs affects the model's forecasting capabilities. This chapter is therefore organized as follows (see Figure 16.1):

- Section 16.2 recalls the model objective and defines the forecasting parameters (i.e., prediction horizon, delta time, and number of past observations). The recurrent model architecture is then optimized, and the results of this optimization are presented.
- Section 16.3 presents the model's performance in forecasting future tool degradation. The results are evaluated based on the R^2 score, the mean absolute error on VB, and the error in estimating the RUL.
- Section 16.4 concludes the chapter by summarizing the key findings.

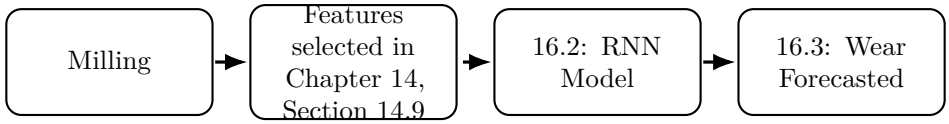


Figure 16.1: Summary of the approach developed in this chapter.

16.2 Model Parameters and Selection

Before defining the model architecture, it is necessary to first establish its objective. In Chapter 8, the selected model aimed to predict tool degradation with a forecasting horizon (H) of up to 3 minutes into the future. This duration was chosen because it approximately corresponds to the production time of a single part in the turning scenario, where the average tool life was about 20 minutes. Thus, the prediction range represented roughly 15 % of the tool's lifespan.

For milling, however, no individual parts are produced in our experimental work, so the same logic cannot be applied directly. The average lifetime for tools ranges between 8 and 10 minutes. To remain consistent with the turning case and to evaluate the model's ability to predict tool evolution, a forecasting horizon of 3 minutes is retained. This corresponds to up to 30 % of the tool's lifespan, meaning the forecasting window is proportionally twice as long as in turning. Maintaining the same prediction horizon despite the shorter lifespan aligns with real-world applications, where decisions must be made several minutes before initiating an operation. In this context, it also enables an exploration of how the model performs when the prediction horizon represents a significant portion of the tool's total lifespan.

As in the turning scenario, a time step of 1 minute is selected, and the last two time steps (i.e., 2 minutes of machining history) were used as input. All these parameters are synthesized in Table 16.1.

Table 16.1: Model parameters of the forecasting models.

	Parameter	**Value**
Model parameters	Horizon (H)	3
	Past values (P)	2
	Delta time [min]	1

Model selection follows the same optimization loop developed for the monitoring task (see Chapter 15, Section 15.2.1). This approach is preferred over manual trial-and-error, as used in turning, due to the significantly larger size of the milling dataset, which makes manual tuning less practical and less efficient.

The tested architectures follow the same principles as those described in Chapter 8, and their corresponding hyperparameters are detailed in Table 16.2. As in the turning case, these architectures consist of a recurrent component (bidirectional or not) followed by a fully connected (stacked) component (see Figure 16.2). The hyperparameters evaluated include the following:

- **Type of RNN**: Defines the type of unit used in the recurrent layers. The same types as in turning were tested: RNN, LSTM, and GRU.
- **Nb of recurrent Layers**: Specifies the number of layers in the recurrent part of the model. Configurations ranging from 1 to 5 layers were evaluated. A small value of dropout is applied between the recurrent layers to avoid overfitting.
- **Size of recurrent layers**: Controls the number of neurons in each recurrent layer. Values between 10 and 30 were tested.
- **Bidirectional**: Indicates whether the recurrent layers are bidirectional. This parameter is binary (True or False).
- **Stacked**: Determines whether a fully connected layer follows the recurrent layers. This parameter is also binary.
- **Nb of stacked layer**: If a fully connected layer is present (i.e., Stacked is True), this parameter defines its depth. Values between 1 and 5 were tested.
- **Nb of neurons in stacked layer**: If stacked layers are included, this parameter controls the number of neurons per layer. Values between 10 and 50 were tested. The ReLU activation function was used for these neurons, as it previously yielded good results in monitoring tasks.

The testing range for each parameter is selected to ensure that increasing or decreasing its value did not affect the results.

Table 16.2: List of hyperparameters optimized using the Optuna optimizer. For each hyperparameter, the tested value range is presented along with the optimal result obtained. The selected value within this optimal range is also indicated in brackets.

Hyperparameters	Range	Optimal (selected)
Type of RNN	RNN, LSTM, GRU	LSTM (LSTM)
Nb of recurrent layers	1 to 5	2 to 3 (2)
Size of recurrent layers	10 to 30	14 to 22 (15)
Bidirectional	True or False	(False)
Stacked	True or False	True or False (True)
Nb of stacked layer	1 to 5	1 to 2 (1)
Nb of neurons in stacked layer	10 to 50	20 to 30 (25)

As in the monitoring task, the objective of the optimization process is to identify a family of robust network architectures. The optimal configuration is presented in Table 16.2. Among the optimal parameters, the selected architecture consists of two non-bidirectional LSTM layers with 15 neurons (units) each, followed by a stacked component comprising one fully connected layer with 25 neurons. This overall architecture represents a compromise between model size and the optimal range identified. Compared to the turning case, where GRU neurons were selected, the recurrent layers in milling use LSTM neurons. As noted in Chapter 8, LSTM

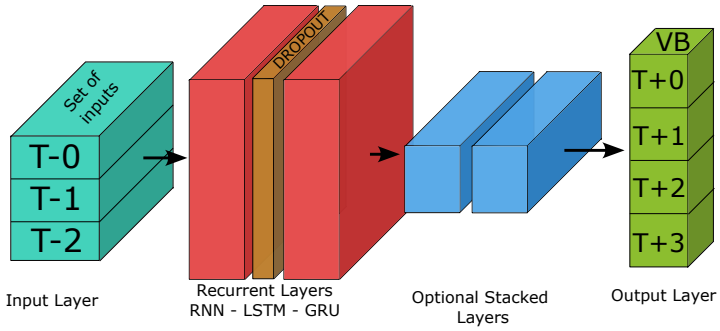


Figure 16.2: Layout of the recurrent architectures tested during the optimization process.

units are more complex than GRU and generally require larger datasets for effective training. This observation is confirmed here, with LSTM-based approaches yielding better performance on the larger milling dataset.

16.3 Forecasting Results

The model optimized in the previous section is then trained on the entire dataset while varying the input features. Table 16.3 presents the R^2 score and Table 16.4 the mean absolute error for the different input combinations, depending on the prediction horizon, along with the average value. Several observations can be drawn from these tables:

- Interestingly, the mean R^2 score (see Table 16.3) is significantly lower for all input data compared to the trends observed during monitoring (see Chapter 15). However, this low value does not affect the mean absolute error, which remains roughly in the same range as what was observed during monitoring. This low R^2 score is also not observed when examining the forecasting trajectories shown in the remainder of this chapter. This suggests that, in this specific case, the R^2 score does not appear to be a reliable metric for assessing the quality of the forecasting.
- The prediction at T+0 corresponds to the monitoring case. Therefore, the model's performance can be directly compared with the results presented in Chapter 15. For all input categories, the mean absolute error (see Table 16.4) is of the same order of magnitude as what was observed during monitoring. The only exception concerns the "Sound" and "Power" inputs, for which the forecasting model yields better results than the monitoring model. This indicates that the forecasting model performs at least as well for monitoring purposes as a dedicated monitoring model. This observation had already been made in the context of turning (see Chapter 8).
- The overall ranking of results follows the same trend observed in the monitoring task: cutting forces offer the most accurate tracking of degradation, followed by the accelerometer, power, and sound. Regarding sensor fusion, Fusion 2 delivers the best performance, followed by Fusion 1 and Fusion 3.

- For all feature combinations, the inclusion of cutting conditions improves the results. This trend was already observed in Chapter 15.
- For most approaches, a decline in the R^2 score and an increase in the mean absolute error are observed as the prediction horizon extends further into the future. This trend, also noted in Chapter 8, aligns with the inherent unpredictability of the wear phenomenon. While general degradation patterns can be identified, accurately forecasting specific causal events that may accelerate tool wear remains a challenge. The only exception to this trend is the accelerometer, which tends to maintain a consistent order of magnitude throughout the forecasting horizon.

These indicators allow for a comparison of the relative performance between models. To visualize how these indicators translate into forecasting application, Figure 16.3 compares the forecasted trajectories based on single sensor input data. The trajectories exhibit the following characteristics:

- **Cutting Forces**. Figures 16.3a and 16.3b shows the forecasted trajectory using input data from cutting forces, without and with cutting conditions respectively. In both scenarios, the forecasts closely match the actual degradation, demonstrating good predictive accuracy. However, the model that incorporates cutting conditions tends to slightly underestimate the degradation compared to the version that excludes them. These trends are similar with observations made during the monitoring phase (see Chapter 15).
- **Accelerations**. Figures 16.3c and 16.3d shows the forecasted trajectory using input data from the accelerometer, without and with cutting conditions respectively. In the case without cutting conditions, the beginning of the tool's life is not accurately predicted. Around 7.5 minutes, the forecast improves, and the end-of-life degradation is better tracked than the beginning. When cutting conditions are included, the degradation is very well tracked throughout the tool's life.
- **Acoustic pressure**. Figures 16.3e and 16.3f shows the forecasted trajectory using input data from sound, without and with cutting conditions respectively. Without cutting conditions, the tool's degradation is not properly tracked. The estimation is similar to the monitoring results using sound features observed in Chapter 15. Adding cutting conditions slightly improves the tracking at the end of the tool's life, but overall performance remains lower than with other sensors.
- **Machine power**. Figures 16.3g and 16.3h shows the forecasted trajectory using input data from power, without and with cutting conditions respectively. Without cutting conditions, the forecast underestimates degradation and shows high variability. Including cutting conditions significantly improves the quality of degradation tracking. In this case, the degradation is very well followed and closely matches the actual wear of the tool.

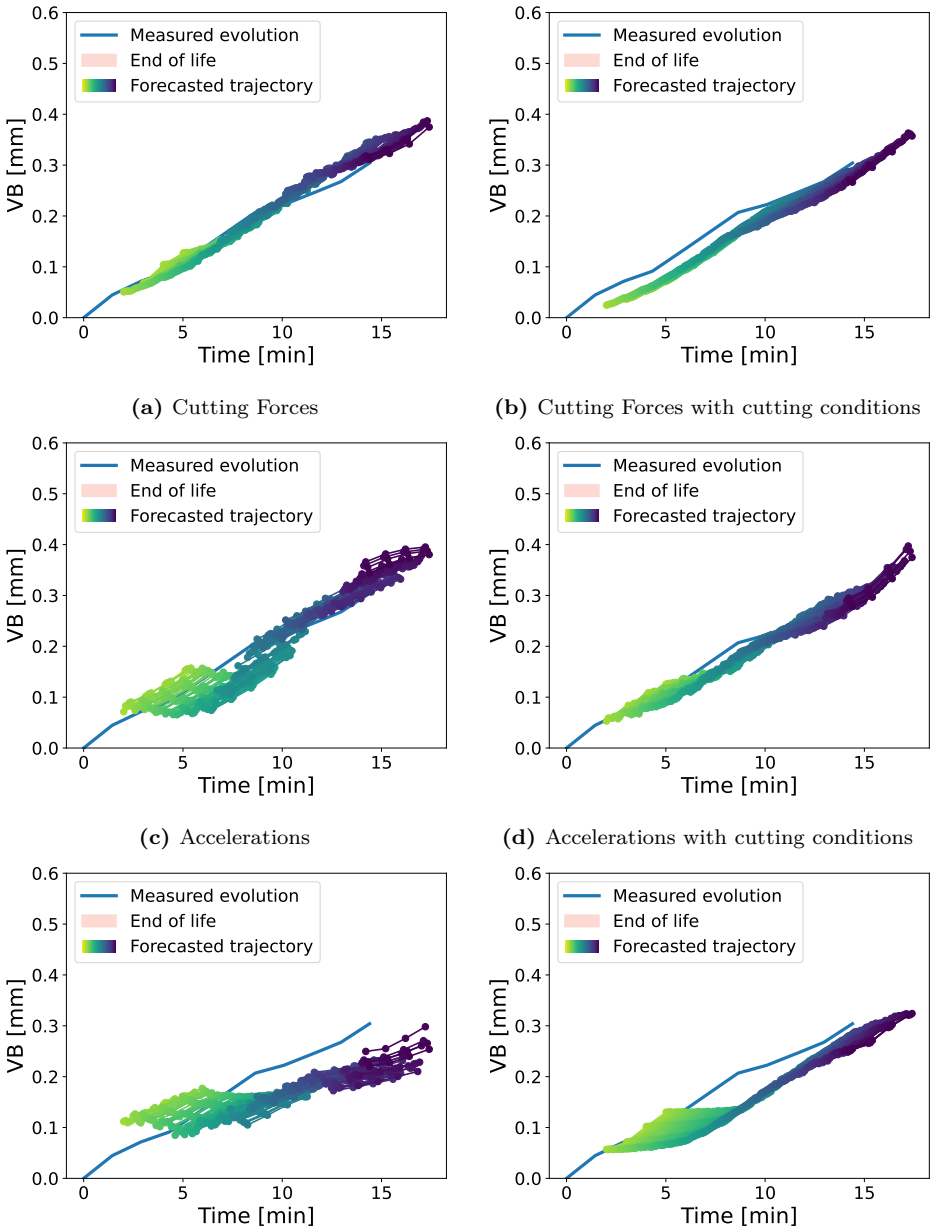
Figure 16.4 presents the forecasted trajectories using sensor fusion as input:

Table 16.3: R^2 score for each combination of input features (as described in Table 14.10). Each value represents the mean R^2 score and its standard deviation over 10 independent runs of the same model. The R^2 score is computed on the whole dataset.

Features	Horizon	Without cutting conditions	With cutting conditions
R^2 Score			
Individual Sensor			
Cutting Forces	T+0	$75.00 \pm 1.66\,\%$	$77.30 \pm 2.82\,\%$
	T+1	$68.71 \pm 1.64\,\%$	$76.64 \pm 1.20\,\%$
	T+2	$67.68 \pm 2.58\,\%$	$72.11 \pm 2.59\,\%$
	T+3	$66.78 \pm 3.96\,\%$	$71.78 \pm 3.77\,\%$
	Mean	$69.54 \pm 4.16\,\%$	$74.46 \pm 3.74\,\%$
Accelerations	T+0	$57.77 \pm 3.50\,\%$	$65.89 \pm 2.02\,\%$
	T+1	$52.42 \pm 3.44\,\%$	$63.59 \pm 2.31\,\%$
	T+2	$52.05 \pm 3.52\,\%$	$64.82 \pm 4.21\,\%$
	T+3	$52.54 \pm 3.97\,\%$	$67.69 \pm 4.29\,\%$
	Mean	$53.70 \pm 4.32\,\%$	$65.50 \pm 3.69\,\%$
Acoustic Pressure	T+0	$35.50 \pm 4.37\,\%$	$65.07 \pm 2.36\,\%$
	T+1	$31.60 \pm 2.82\,\%$	$61.44 \pm 3.40\,\%$
	T+2	$31.75 \pm 3.36\,\%$	$59.83 \pm 3.10\,\%$
	T+3	$32.19 \pm 5.89\,\%$	$62.53 \pm 4.42\,\%$
	Mean	$32.76 \pm 4.56\,\%$	$62.22 \pm 3.90\,\%$
Machine Power	T+0	$43.13 \pm 5.95\,\%$	$79.89 \pm 2.61\,\%$
	T+1	$45.33 \pm 4.16\,\%$	$75.70 \pm 3.78\,\%$
	T+2	$42.29 \pm 2.68\,\%$	$75.40 \pm 4.62\,\%$
	T+3	$44.31 \pm 3.71\,\%$	$70.32 \pm 3.61\,\%$
	Mean	$43.76 \pm 4.44\,\%$	$75.32 \pm 5.04\,\%$
Sensor Fusion			
Fusion1	T+0	$72.96 \pm 4.59\,\%$	$77.53 \pm 4.37\,\%$
	T+1	$63.61 \pm 6.33\,\%$	$73.41 \pm 3.82\,\%$
	T+2	$59.55 \pm 6.23\,\%$	$70.75 \pm 4.52\,\%$
	T+3	$58.50 \pm 4.06\,\%$	$68.07 \pm 3.97\,\%$
	Mean	$63.66 \pm 7.85\,\%$	$72.44 \pm 5.45\,\%$
Fusion2	T+0	$72.29 \pm 2.48\,\%$	$77.44 \pm 2.71\,\%$
	T+1	$72.92 \pm 2.50\,\%$	$73.80 \pm 2.88\,\%$
	T+2	$67.07 \pm 2.86\,\%$	$73.30 \pm 4.90\,\%$
	T+3	$64.81 \pm 3.74\,\%$	$69.90 \pm 4.59\,\%$
	Mean	$69.27 \pm 4.52\,\%$	$73.61 \pm 4.72\,\%$
Fusion3	T+0	$54.53 \pm 3.50\,\%$	$63.76 \pm 4.81\,\%$
	T+1	$54.24 \pm 4.96\,\%$	$61.45 \pm 5.22\,\%$
	T+2	$50.60 \pm 3.23\,\%$	$62.68 \pm 8.21\,\%$
	T+3	$47.68 \pm 3.55\,\%$	$58.93 \pm 7.91\,\%$
	Mean	$51.76 \pm 4.79\,\%$	$61.70 \pm 6.95\,\%$

Table 16.4: Mean absolute error for each combination of input features (as described in Table 14.10). Each value represents the mean absolute error and its standard deviation over 10 independent runs of the same model. The mean absolute error is computed on the whole dataset.

Features	Horizon	Without cutting conditions	With cutting conditions
Mean Absolute Error			
Individual Sensor			
Cutting Forces	T+0	$16.34 \pm 0.59\,\mu m$	$15.41 \pm 1.10\,\mu m$
	T+1	$18.59 \pm 0.80\,\mu m$	$16.27 \pm 0.75\,\mu m$
	T+2	$18.43 \pm 0.87\,\mu m$	$17.61 \pm 0.94\,\mu m$
	T+3	$18.75 \pm 1.02\,\mu m$	$17.25 \pm 0.72\,\mu m$
	Mean	$18.03 \pm 1.29\,\mu m$	$16.64 \pm 1.24\,\mu m$
Accelerations	T+0	$23.71 \pm 1.30\,\mu m$	$19.26 \pm 0.99\,\mu m$
	T+1	$24.79 \pm 1.36\,\mu m$	$20.18 \pm 0.91\,\mu m$
	T+2	$25.90 \pm 1.43\,\mu m$	$20.96 \pm 1.63\,\mu m$
	T+3	$24.98 \pm 1.37\,\mu m$	$19.95 \pm 1.36\,\mu m$
	Mean	$24.85 \pm 1.57\,\mu m$	$20.09 \pm 1.39\,\mu m$
Acoustic Pressure	T+0	$35.91 \pm 1.47\,\mu m$	$20.45 \pm 0.93\,\mu m$
	T+1	$34.57 \pm 1.19\,\mu m$	$22.02 \pm 1.33\,\mu m$
	T+2	$33.12 \pm 1.87\,\mu m$	$22.31 \pm 1.16\,\mu m$
	T+3	$29.49 \pm 2.10\,\mu m$	$20.66 \pm 1.17\,\mu m$
	Mean	$33.27 \pm 2.93\,\mu m$	$21.36 \pm 1.41\,\mu m$
Machine Power	T+0	$26.44 \pm 1.92\,\mu m$	$13.34 \pm 0.99\,\mu m$
	T+1	$26.25 \pm 1.49\,\mu m$	$14.69 \pm 1.21\,\mu m$
	T+2	$26.78 \pm 1.08\,\mu m$	$15.93 \pm 1.81\,\mu m$
	T+3	$25.10 \pm 1.55\,\mu m$	$17.78 \pm 1.64\,\mu m$
	Mean	$26.14 \pm 1.66\,\mu m$	$15.43 \pm 2.19\,\mu m$
Sensor Fusion			
Fusion1	T+0	$17.53 \pm 1.81\,\mu m$	$15.07 \pm 1.47\,\mu m$
	T+1	$20.59 \pm 2.39\,\mu m$	$16.75 \pm 1.63\,\mu m$
	T+2	$22.79 \pm 1.67\,\mu m$	$17.84 \pm 1.99\,\mu m$
	T+3	$23.31 \pm 1.28\,\mu m$	$18.42 \pm 1.66\,\mu m$
	Mean	$21.06 \pm 2.92\,\mu m$	$17.02 \pm 2.12\,\mu m$
Fusion2	T+0	$17.09 \pm 0.74\,\mu m$	$14.45 \pm 0.93\,\mu m$
	T+1	$17.67 \pm 0.66\,\mu m$	$16.28 \pm 1.34\,\mu m$
	T+2	$19.61 \pm 0.87\,\mu m$	$16.67 \pm 1.90\,\mu m$
	T+3	$20.51 \pm 1.51\,\mu m$	$16.74 \pm 1.40\,\mu m$
	Mean	$18.72 \pm 1.71\,\mu m$	$16.04 \pm 1.71\,\mu m$
Fusion3	T+0	$23.92 \pm 1.42\,\mu m$	$19.31 \pm 1.32\,\mu m$
	T+1	$24.42 \pm 1.37\,\mu m$	$20.64 \pm 2.14\,\mu m$
	T+2	$26.15 \pm 1.60\,\mu m$	$20.70 \pm 2.33\,\mu m$
	T+3	$25.70 \pm 2.26\,\mu m$	$21.20 \pm 2.64\,\mu m$
	Mean	$25.05 \pm 1.93\,\mu m$	$20.46 \pm 2.28\,\mu m$



(a) Cutting Forces

(b) Cutting Forces with cutting conditions

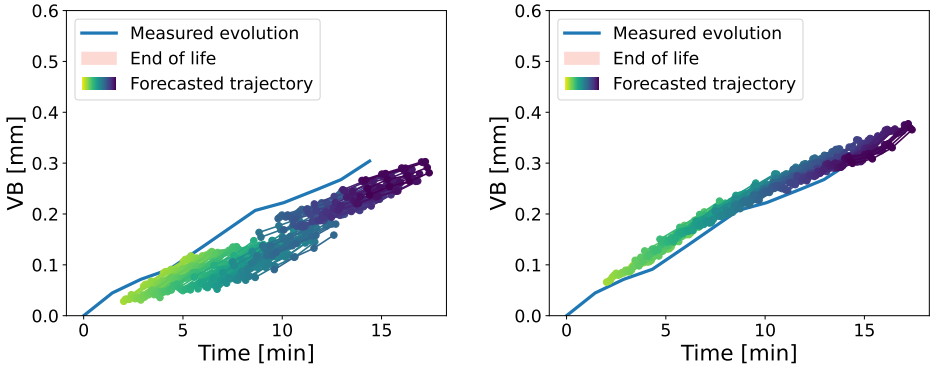
(c) Accelerations

(d) Accelerations with cutting conditions

(e) Acoustic pressure

(f) Acoustic pressure with cutting conditions

Figure 16.3: Comparison of different inputs on the forecasting capabilities for each sensor. The monitored tool is the tool 0020. The colour of each prediction curve reflects the time at which the forecast was made: the earlier the prediction, the lighter the curve.

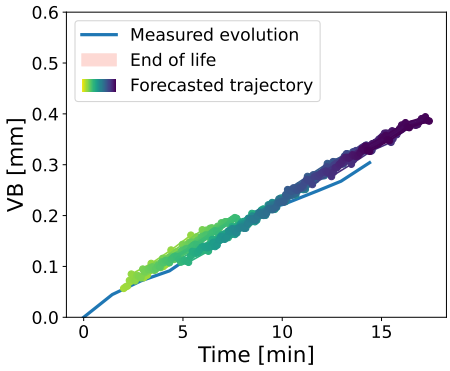


(g) Power **(h)** Power with cutting conditions

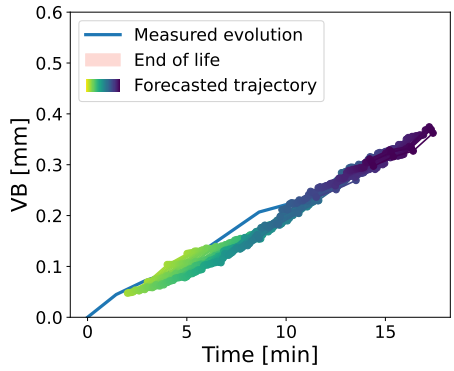
Figure 16.3: *...Continued*

- **Fusion 1**. Figures 16.4a and 16.4b shows the forecasted trajectory using input data from sensor fusion 1, without and with cutting conditions respectively. In both cases, the degradation is accurately tracked and predicted. Without cutting conditions, the model tends to overestimate the degradation, whereas with cutting conditions, it slightly underestimates it. This is the same observation as in Chapter 15
- **Fusion 2**. Figures 16.4c and 16.4d shows the forecasted trajectory using input data from sensor fusion 2, without and with cutting conditions respectively. Both trajectories are very similar, with a consistent trend of overestimation without cutting conditions and slight underestimation with them. This is the same observation as in Chapter 15
- **Fusion 3**. Figures 16.4e and 16.4f shows the forecasted trajectory using input data from sensor fusion 3, without and with cutting conditions respectively. Without cutting conditions, the degradation tracking resembles a blend of the results obtained with accelerometer data (Figure 15.4c) and power data (Figure 15.4g). The tracking is slightly better than either sensor used individually. Adding cutting conditions further improves the degradation tracking.

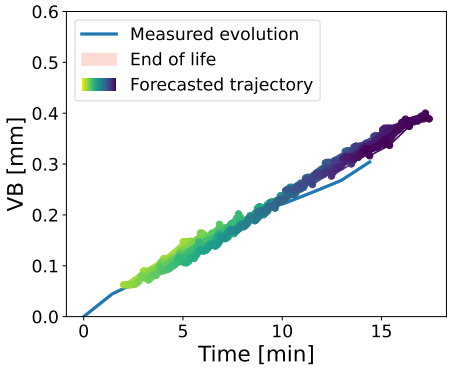
Previous analyses have shown how tool wear is estimated, but they do not clearly demonstrate how this estimation is used to detect the end of the tool's life. Table 16.5 presents the mean absolute RUL estimated by the model's forecasting for different input combinations when the wear reaches 200 µm. This threshold is chosen because all tools in the dataset reached at least this level of wear, allowing performance evaluation across the entire database. The table reinforces the earlier observation: including cutting conditions improves forecasting accuracy, as reflected by a lower RUL estimation error. The ranking of input combinations remains consistent, cutting forces yield the best results, with an RUL estimation error below 1 minute. The worst-performing input is still the acoustic signal without cutting conditions, resulting in an error of approximately 2.3 minutes. Similarly,



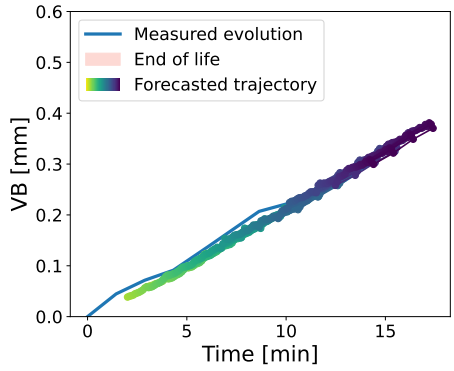
(a) Fusion 1



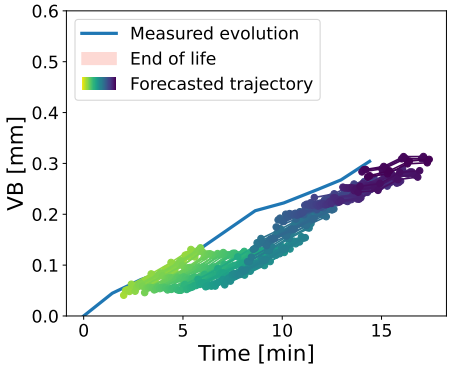
(b) Fusion 1 with cutting conditions



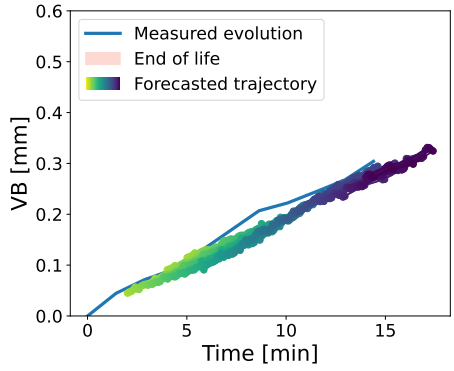
(c) Fusion 2



(d) Fusion 2 with cutting conditions



(e) Fusion 3



(f) Fusion 3 with cutting conditions

Figure 16.4: Comparison of different sensor fusion inputs on the forecasting capabilities. The monitored tool is the tool 0020.

power leads to an error above 2 minutes when cutting conditions are not provided. However, when cutting conditions are included, the error significantly decreases, placing power among the better-performing inputs. All other input combinations result in RUL estimation errors around 1 minute.

Table 16.5: Remaining Useful Life (RUL) mean absolute error for each combination of input features (as described in Table 14.10). The RUL is evaluated for wear reaching 200 µm. Each value represents the mean absolute error and its standard deviation over 10 independent runs of the same model. The mean absolute error is computed on the whole dataset.

Features	Without cutting conditions	With cutting conditions
Mean absolute RUL error		
Individual Sensor		
Cutting Forces	0.81 ± 0.02 min	0.78 ± 0.04 min
Accelerations	1.62 ± 0.14 min	1.21 ± 0.02 min
Acoustic Pressure	2.33 ± 0.12 min	1.41 ± 0.15 min
Machine Power	2.24 ± 0.21 min	0.77 ± 0.08 min
Sensor Fusion		
Fusion1	1.10 ± 0.02 min	0.82 ± 0.16 min
Fusion2	0.84 ± 0.09 min	0.83 ± 0.03 min
Fusion3	1.45 ± 0.14 min	1.00 ± 0.07 min

16.4 Conclusions

This chapter presents the use of recurrent neural networks for forecasting tool wear in milling operations. The objective is to apply the methods developed for turning (see Chapter 8) to the milling context. Therefore, the same constraints were applied to the models as in the turning case, namely predicting the tool state up to 3 minutes into the future. This choice is more demanding than in turning, as the tool lifespan in milling is nearly twice as short. As a result, this chapter explores the behavior of such models over a proportionally longer forecasting horizon. Several observations can be made:

- As in turning, various types of recurrent neural network cells were tested: RNN, LSTM, and GRU. After model optimization, it was found that the optimal model is a recurrent network based on LSTM cells, which differs from the approach used in turning (see Chapter 8), where GRU cells were employed. This difference is justified by the milling dataset being larger and more complex, making LSTM cells more suitable for this task.

- The performance of this optimal model is evaluated across all input configurations. The hierarchy of result quality is consistent with what was observed during monitoring (see Chapter 15): cutting forces yield the best performance, followed by power and accelerations. Acoustic signals do not provide reliable

forecasting of degradation. For all input types, adding cutting conditions significantly improves model performance.

- When comparing the forecasting model's performance at T+0 with the monitoring models developed in Chapter 15, it is observed that the forecasting model achieves results as good as a dedicated monitoring network. This observation was already made in Chapter 8. It implies that a forecasting network is at least as effective as a dedicated monitoring network. In cases where the input data are sound or power, the monitoring results are even better than those of a dedicated network.

- The further the prediction horizon extends, the greater the error in estimating the tool state. This observation was also made in Chapter 8. However, the performance degradation over the prediction span remains limited and the approach remains an asset for supporting decision-making.

In summary, this chapter successfully applies the forecasting methods developed for turning to the milling process.

Chapter 17

Classifying the Cutting Tool State from Tool Images in Milling

Direct Monitoring with Classification

This chapter presents the application to milling of the methods developed for turning in Chapter 9.

17.1 Introduction

This chapter applies the image-based tool classification methods developed in Chapter 9 to the milling context. The goal is to classify the condition of a tool based on images of its flank face. Three classification models are evaluated: VGG19, EfficientNetV2, and Vision Transformers. All of these models have been pre-trained on the ImageNet dataset. The structure of this chapter is outlined in Figure 17.1.

- Section 17.2 introduces the dataset and the image modifications applied to train the models.
- Section 17.3 provides a brief overview of the models used for image classification.
- Section 17.4 presents the results. It begins with a discussion of the overall performance, followed by an analysis of the distribution of classification errors, and concludes with a discussion on model explainability.

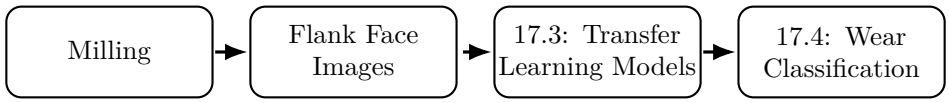


Figure 17.1: Summary of the approach developed in this chapter.

17.2 Classification and Modification of Images

The milling dataset contains 826 images of the cutting tool's flank face. Each image correspond to the flank face of one tooth of the tool, each tool therefore has 3 images by inspection. As in the turning case, the dataset is divided into three classes, each corresponding to a predefined range of average tool flank wear:

- **Class 1**. Corresponds to a mean flank wear between 0 and 150 µm
- **Class 2**. Corresponds to a mean flank wear comprises between 150 and 300 µm
- **Class 3**. Corresponds to a mean flank wear above 300 µm.

Similar to the turning case, the milling dataset exhibits an imbalanced class distribution: there are 415 images in class 1, 337 in class 2, and 74 in class 3. This corresponds to a distribution of 50.2 % for class 1, 40.8 % for class 2, and 9 % for class 3. In the turning dataset, oversampling was used to balance the class proportions. This approach was suitable due to the small dataset size, and it did not negatively affect the results. For the milling dataset, which contains significantly more data, the oversampling strategy can be adjusted to avoid increasing the total number of images while still addressing the class imbalance. Therefore, to mitigate this imbalance during model training, the loss function is weighted according to the class distribution. This means that underrepresented classes receive higher weights, ensuring that errors on these classes contribute more significantly to the overall loss. The model is trained using a weighted cross-entropy loss function ($\mathcal{L}$), defined as:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} w_{y_i} \cdot \log(p_{i,y_i}) \tag{17.1}$$

With

- N the number of samples in the batch,
- C the number of classes,
- $y_i \in \{0, 1, ..., C-1\}$ the true class label for sample i,
- $p_i = (p_{i1}, p_{i2}, ..., p_{iC})$ the predicted probabilities for sample i,
- w_c the weight assigned to the corresponding class,
- p_{i,y_i} is the predicted probability for the true class y_i of sample i,
- w_{y_i} is the weight corresponding to the true class of sample i.

This weighted loss helps to balance the learning process on imbalanced dataset. To evaluate the model, 25 % of the data from the least represented class (class 3) was selected for testing. This class contains 74 images, of which 20 ($\sim$25 %) are used as the testing dataset. To ensure a balanced and consistent test set, 20 images were also selected from class 1 and class 2. The testing dataset therefore consists

of 60 images, evenly distributed across the three classes, with 20 images per class. The remaining images are used to construct the training dataset, which is further split into 90 % for training and 10 % for validation. It is important to note that images are classified individually, meaning that it is possible for one tooth of a tool to belong to one class, while another tooth of the same tool belongs to a different class. As a result, classification performance is evaluated tooth by tooth, rather than tool by tool.

The raw images are cropped around the wear zone to better centre the tool's tooth in the frame. This ensures that the wear area occupies a larger portion of the image. Figure 17.2 shows an example of a cropped image of a cutting tool. Each image is resized to 224 by 224 pixels to be compatible with the classification models used (see Section 17.3).



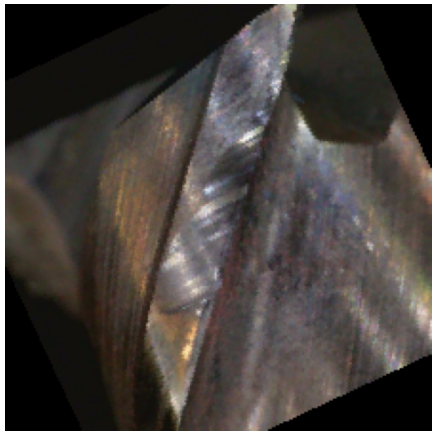
Figure 17.2: Cropped image from the database to focus on the wear zone. The resolution is reduced to the models input size (224 pixels by 224 pixels).

During training, data modification is randomly applied to improve generalization. As in Chapter 9, the following transformations are used:

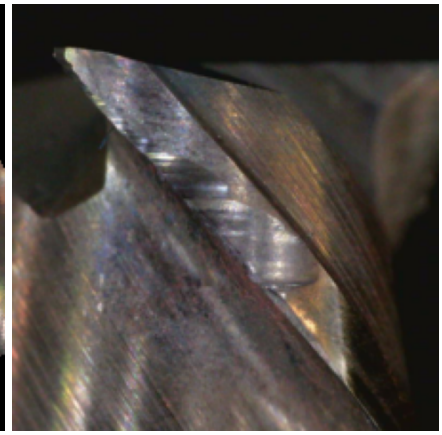
- Random rotation between −25 and 25° (Figure 17.3a)
- Horizontal flip with a probability of 50 % (Figure 17.3b)
- Brightness adjustment between −50 % (Figure 17.3d) and 50 % (Figure 17.3c)
- Contrast adjustment between −50 % (Figure 17.3f) and 50 % (Figure 17.3e)

17.3 Pre-Trained Models for Image Classification

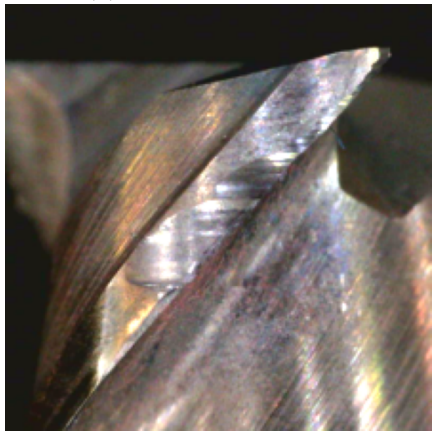
Motivated by the same considerations as those discussed in Chapter 9, three types of transfer learning models are evaluated: VGG19, EfficientNetV2, and Vision Transformers (ViTs). All these models are pre-trained on the ImageNet dataset.



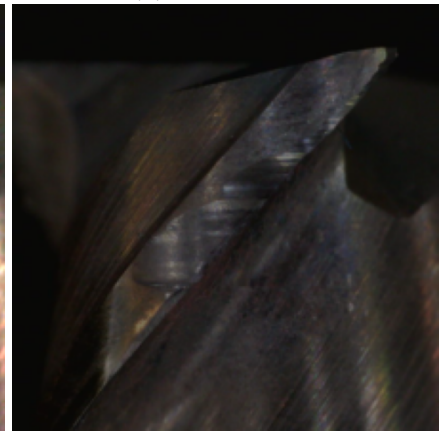
(a) Rotated Image by 25°



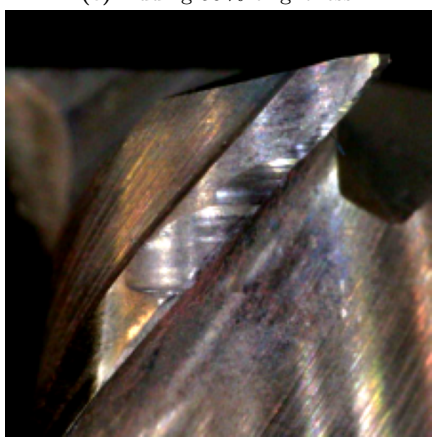
(b) Flipped Image



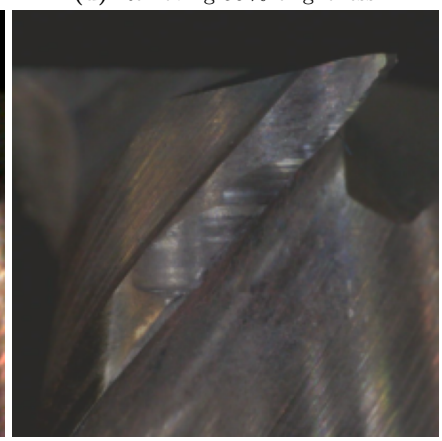
(c) Adding 50 % brightness



(d) Removing 50 % brightness



(e) Adding 50 % contrast



(f) Removing 50 % contrast

Figure 17.3: Maximum image modifications applied to image from Figure 17.2.

To adapt these models for tool wear classification, their final layers are modified. Specifically, linear layers with ReLU activation functions are added to progressively reduce the dimensionality of the model outputs. The final layer is a "Softmax" layer, which produces a probability distribution over the three wear classes, indicating the likelihood that an image belongs to each class. The architecture and configuration of these added layers were determined through trial and error, aiming to balance model complexity and classification performance. All models are trained using input images with a resolution of 224×224 pixels over 300 epochs, with a batch size of 75. All standard techniques for preventing overfitting and monitoring training progress are applied during model training. The Adam optimizer is used, with a learning rate of 1×10^{-3} for VGG19 and EfficientNetV2, and 1×10^{-5} for ViT. These values were obtained through trial and error.

17.4 Classification Results

17.4.1 Results

Table 17.1 presents a comparison of the performance of different models based on various image modifications, as well as their overall results. Among the three models, EfficientNetV2 performs the worst, consistently achieving lower classification scores compared to the other two. This underperformance may be attributed to the model's learning approach, which might be less suited to this specific task. In contrast, VGG19 and ViT exhibit comparable performance, with ViT demonstrating greater robustness to changes in contrast and brightness than the convolution-based VGG19. This result differs from what was observed in the turning context, where only VGG19 performed well. Here, Vision Transformers achieve a score similar to VGG19. In this case, the larger milling dataset appears to benefit the Vision Transformers.

Figure 17.4 shows the confusion matrices for the three models. These matrices are constructed using the same 10 runs used to generate the results in Table 17.1. The confusion matrices allow to observe, for each true class, where the model made prediction errors. This helps identify critical misclassifications, for example, incorrect predictions for class 3. Indeed, if the model fails to detect class 3, the information that the tool needs to be replaced can not be taken into account. Several observations can be made from these matrices:

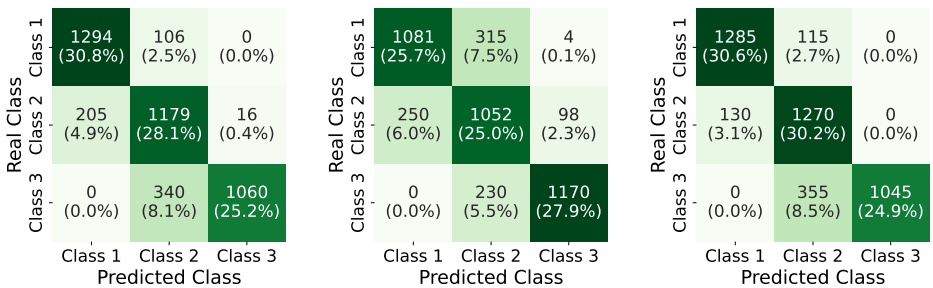
- The confusion matrix for VGG19 (see Figure 17.4a) shows that class 1 is mostly well predicted, with some misclassifications into class 2. This class is, in fact, the best predicted among the three models. Class 2 is slightly less accurate, with most errors being misclassified as class 1, and only a few misclassifications from class 2 to class 3. However, the model makes more errors between class 3 and predictions in class 2.

- The confusion matrix for EfficientNetV2 (see Figure 17.4b) shows similar trends to VGG19, but with more errors when the true class is 1, which is often predicted as class 2. Interestingly, EfficientNetV2 achieves the best classification performance for class 3 among the three models, making fewer misclassifications for this class.

Table 17.1: Comparison of classification accuracy across different architectures under various image modifications. Each value represents the mean accuracy over 10 independent runs of the same model.

Image Modification	VGG19	EfficientNetV2	ViT
Accuracy per Modification			
No modification	$83.5 \pm 3.2\,\%$	$81.3 \pm 2.3\,\%$	$84.4 \pm 3.4\,\%$
Flipped	$86.6 \pm 2.1\,\%$	$80.5 \pm 2.7\,\%$	$85.3 \pm 2.6\,\%$
Rotated	$89.5 \pm 2.1\,\%$	$82.1 \pm 3.2\,\%$	$86.1 \pm 3.4\,\%$
Brightness +50 %	$80.3 \pm 5.3\,\%$	$72.6 \pm 2.3\,\%$	$84.6 \pm 3.4\,\%$
Brightness -50 %	$84.6 \pm 4.2\,\%$	$79.8 \pm 3.2\,\%$	$87.8 \pm 2.8\,\%$
Contrast +50 %	$78.5 \pm 5.4\,\%$	$75.6 \pm 2.6\,\%$	$84.0 \pm 3.2\,\%$
Contrast -50 %	$85.6 \pm 3.6\,\%$	$78.3 \pm 4.2\,\%$	$87.8 \pm 2.3\,\%$
Overall Score			
Accuracy	$84.1 \pm 5.2\,\%$	$78.8 \pm 4.3\,\%$	$85.7 \pm 3.4\,\%$
F1 Score	$84.1 \pm 5.4\,\%$	$78.8 \pm 4.3\,\%$	$85.8 \pm 3.4\,\%$
Precision	$86.4 \pm 3.7\,\%$	$80.4 \pm 4.7\,\%$	$88.1 \pm 2.2\,\%$
Recall	$84.1 \pm 5.2\,\%$	$78.6 \pm 4.3\,\%$	$85.7 \pm 3.4\,\%$

- The confusion matrix for ViT (see Figure 17.4c) shows similar trends to those observed with VGG19. However, ViT performs better in classifying samples from class 2 with no error between class 2 and 3.



(a) VGG19 **(b)** EfficientNetV2 **(c)** ViT

Figure 17.4: Confusion matrices for the three tested models. Each value represents the cumulative results over 10 independent runs of the same model.

These matrices highlight the misclassification errors made by the models. Most of these errors occur between neighboring classes. For example, there are almost no misclassifications between class 1 and class 3, indicating that the models tend to confuse classes with similar characteristics. For instance, a tool with 149 µm of wear is classified as belonging to Class 1, even though it is just 1 µm away from the threshold for Class 2. In practice, such a fine distinction is ambiguous and has little industrial significance. A model that makes most of its errors near the boundaries

between classes is less prone to unadapted decision than one that misclassifies tools located well within a class interval.

Table 17.2 shows the distribution of classification errors across different models. As discussed in Chapter 9, transitions between classes are defined as ± 25 µm around the threshold values. For example, the transition between Class 1 and Class 2 spans from 125 to 175 µm, while the transition between Class 2 and Class 3 ranges from 275 to 325 µm.

From Table 17.2, it is observed that all models tend to make the majority of their errors within these transition zones. It is also shown that EfficientNetV2 makes significantly more error in Class 2 than the other models as it was highlighted with the confusion matrices. The VGG19 and ViT models, on the other hand, make fewer errors within the core of each class and more within the transition zones. An error pattern that, as previously explained, has less negative industrial impact.

One contributing factor to these transition-zone errors is the image resizing process. For the models evaluated here, images are downscaled to 224×224 pixels, which causes pixelation of the wear region. In the original microscope resolution, one pixel corresponds to approximately 5 µm. After resizing, this becomes roughly 17.85 µm per pixel due to a scaling factor of 3.57. As a result, a one-pixel misalignment becomes significantly more impactful. It's worth noting that this limitation can be mitigated. For example, ViT models can handle higher-resolution images, but this comes at the cost of increased model complexity. In this study, however, all models were evaluated under the same conditions to ensure a fair comparison.

Table 17.2: Position of errors in the classification. More errors in the transition have less impact in real milling application.

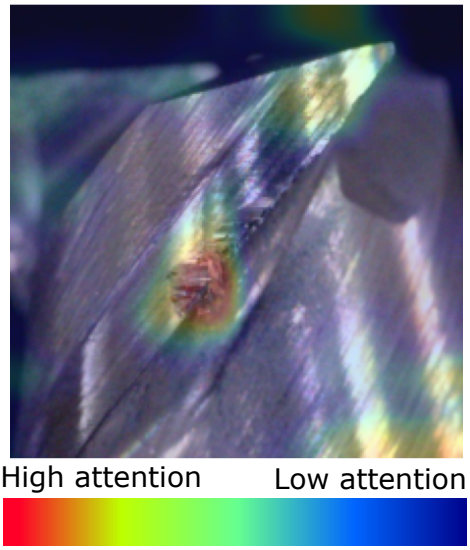
Position of errors in the classification					
Model	**Class 1**	**Transition 1-2**	**Class 2**	**Transition 2-3**	**Class 3**
VGG19	61	214	52	229	111
EfficientNetV2	101	341	225	81	149
ViT	22	218	5	262	93

In conclusion, for tool condition classification, the ViT model appears to be the most effective in this case. It achieves strong overall performance and makes most of its errors in transitions between neighboring classes. Although EfficientNetV2 obtained the highest accuracy for classifying class 3, its overall performance is lower compared to the other models.

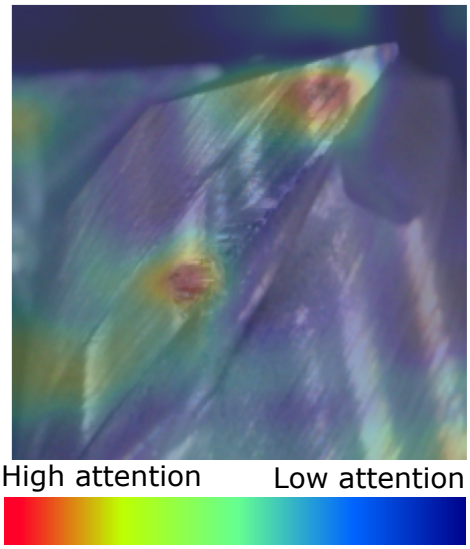
17.4.2 Explainability of the models

In addition to performance evaluation, Convolutional Neural Networks (CNNs), such as VGG19, can greatly benefit from Grad-CAM analysis, which enhances the interpretability of their predictions. This technique generates a heatmap that highlights the regions of the input image that most strongly influence the model's decision. Figure 17.5 illustrates the attention regions on two images, obtained

using an XGrad-CAM implementation. XGrad-CAM is an advanced variant of Grad-CAM, designed to improve both the sensitivity and accuracy of the resulting attention maps [184]. Figures 17.5a and 17.5b illustrate the model's behavior on the same image under different conditions. Figure 17.5a displays the attention map generated by the VGG19 model for an unaltered image of the tool. In this case, the model clearly focuses on the edge of the tool, indicating that it has successfully identified the wear area. In contrast, Figure 17.5b shows the same image, but with a 50 % reduction in contrast. Here, the model's attention is more dispersed and less concentrated on the wear area, leading to an incorrect classification. This type of interpretation helps ensure that the model has genuinely learned to detect tool wear and provides explainability to its predictions. It is worth noting that attention maps are also available for Vision Transformers, and their interpretation follows the same principles as those used in Grad-CAM analyses.



(a) Attention map on the original image. The wear zone is correctly located by the model. The model correctly classify the tool as class 1.



(b) Attention map on the same image but with a decrease in contrast. The attention zone is more diffused on the image and the model therefore incorrectly classify the tool as class 2.

Figure 17.5: XGrad-CAM attention map for VGG19 classification with the original image and the same image with a decrease in contrast. Warmer colours indicate stronger attention.

17.5 Conclusions

The models developed in this chapter demonstrate strong performance, confirming that transfer learning is an effective approach for building high-performing classifiers. Interestingly, the most robust models in this case are Vision Transformers (ViTs). This finding contrasts with the conclusion drawn in Chapter 9, where it was noted that although ViTs could theoretically outperformed convolutional models,

they required significantly more training data. In the milling dataset, which is considerably larger than the turning dataset, this limitation was lifted. As a result, it is now evident that transformer-based models are not only highly effective but also robust to image alterations.

From a practitioner's perspective, it is important to note that a three-class classification may have limited practical relevance. In real-world applications, tool replacement is typically only triggered when the tool reaches the third class, indicating severe wear. However, in this chapter, it was chosen to maintain a three-class classification to demonstrate that the models are not only capable of detecting tool wear but also of quantifying it.

Chapter 18

Monitoring the Cutting Tool Degradation from Tool Images in Milling

Direct Monitoring with Regression

This chapter presents the application to milling of the methods developed for turning in Chapter 10.

18.1 Introduction

In Chapter 10, image segmentation models were developed to detect wear zones on cutting tools. Although the detection accuracy was high, limitations in the turning dataset prevented further insights into the industrial applicability of such models. Therefore, when creating the milling dataset, special attention was given to acquiring higher-quality and more consistent images (see Chapter 13).

This chapter builds upon the work done for turning, but applies it to milling. Specifically, it uses image segmentations to estimate tool wear which was not possible with previous development. The chapter is structured as follows (Figure 18.1):

- Section 18.2 describes the images used and how they are preprocessed to ensure the models generalize as effectively as possible.
- Section 18.3 recalls the 3 U-Net models used to perform the segmentation task.
- Section 18.4 presents and compares the tool segmentation results obtained from the three U-Net models. Then, the procedure used to measure tool wear from the segmentation maps is explained, followed by a comparison of measurement performance based on the segmentation outputs from the three models.

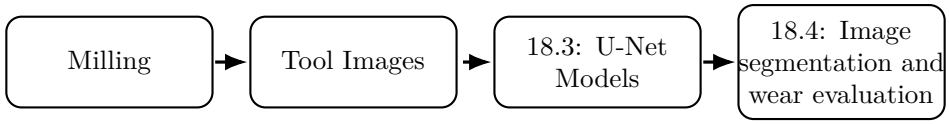


Figure 18.1: Summary of the approach developed in this chapter.

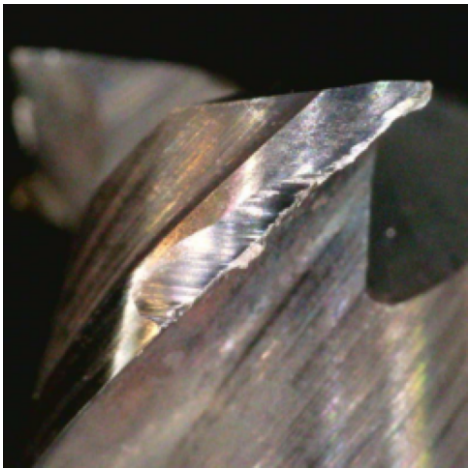
18.2 Image Preparation for Segmentation

The dataset used to train the models consists of images of cutting tools captured from both the flank face and the tip of the tool. For each tool inspection, six images are collected: three of the flank face and three of the tool tip, one for each tooth. Therefore, in total, the dataset includes 1,652 images, evenly distributed between flank and tip views. Figures 18.2b and 18.3a show an example of the flank face images with its associated mask whereas Figures 18.2c and 18.2d show a tip image and its mask. Unlike the turning process, the milling dataset does not exhibit any built-up edges on the tool. As a result, the mask used in this context exclusively represents tool wear. The quantity of images is higher than for the turning database and all images were acquired under similar conditions. The images are split into 75 % for the training dataset, 20 % for the testing dataset, and the remaining 5 % for the validation dataset. These proportions are similar to those used in previous experiments, though slightly different. In earlier approaches, the test sets were grouped either by complete trajectory (for indirect methods) or by class (for direct classification), which imposed constraints on the dataset proportions. In this chapter, there are no such constraints on test group selection, allowing the use of a standard testing proportion of 20 %.

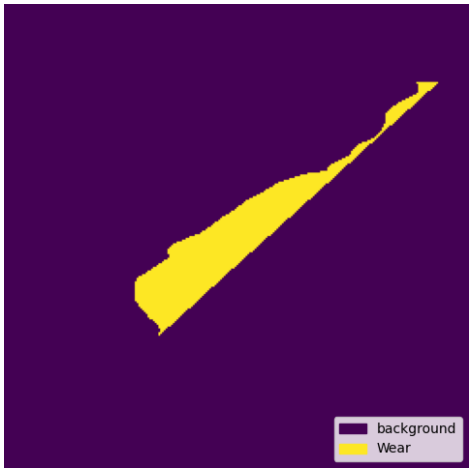
To mitigate the risk of overfitting and evaluate the model under various image transformations, data augmentation techniques are applied. As in previous chapters involving image-based models, the following transformations are used:

- Random rotation between -25 and $25°$
- Random horizontal flip with a probability of 50 %
- Random brightness adjustment between -50 % and 50 %
- Random contrast adjustment between -50 % and 50 %

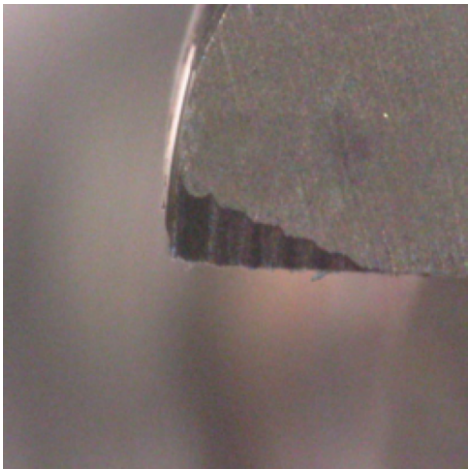
To prepare the images for use by the models, they are first zoomed in on the wear area before being resized to 256 by 256 pixels. This input size of 256×256 pixels corresponds to the required dimensions for the U-Net models (see Section 18.3). It differs from the classification models used previously, where the input size was constrained to 224×224 pixels due to the need to adapt to pre-existing architectures. By centering the image on the wear zone and applying a zoom prior to downscaling, the wear area occupies a larger portion of the final image. This not only benefits the learning process by dedicating more image space to the relevant features, but more importantly, it helps keeping measurement accuracy. Ensuring that each pixel represents a smaller physical area reduces the risk of losing resolution in the process of wear measurement.



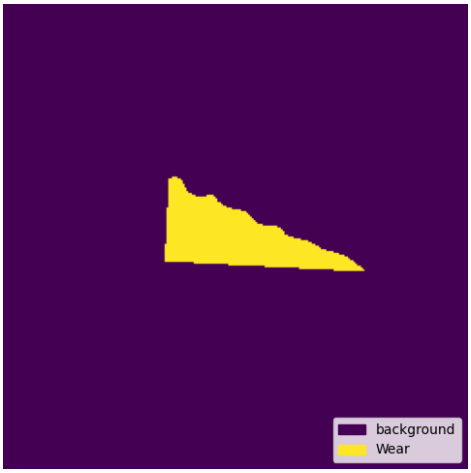
(a) Example of flank image



(b) Associated flank Mask



(c) Example of tip image



(d) Associated tip Mask

Figure 18.2: Example of flank and tip images from the database with their associated mask. The images are in the model's input resolution (256 by 256 pixels).

18.3 Segmentation Models

To perform image segmentation, the same models developed in Chapter 10 are used: Classic U-Net, U-Net++, and Double U-Net. These models retain the same characteristics as those applied in the turning context, with an input image size of 256×256 pixels to limit the number of parameters. All models are trained under identical conditions. The maximum number of epochs is set to 100, with a batch size of 16. Overfitting is monitored throughout the training process, but no signs of overfitting were observed for any of the models. Each model is trained on the same image dataset, with random transformations applied during training to enhance generalization.

18.4 Results

18.4.1 Segmentation Results

Table 18.1 presents the results of the three models developed for image segmentation. The performance metric used is the Intersection over Union (IoU), as defined in Chapter 10, which measures the overlap between the predicted segmentation and the ground truth. A score of 100 % indicates a perfect segmentation.

All three models exhibit strong performance across various image modifications, highlighting their robustness. Their overall metrics are also very similar and lie within the same range, making it difficult to identify a clearly superior model based solely on IoU performance.

Table 18.1: Comparison of IoU accuracy across different architectures under various image modifications. Each value represents the mean IoU over 10 independent runs of the same model.

Image Modification	U-Net	U-Net ++	DU-Net
IoU per Modification			
No modification	88.4 ± 0.7 %	88.7 ± 0.6 %	88.3 ± 0.6 %
Flipped	88.5 ± 0.7 %	88.7 ± 0.5 %	88.5 ± 0.6 %
Rotated	86.6 ± 1.3 %	87.4 ± 0.8 %	86.9 ± 0.7 %
Brightness +50 %	87.5 ± 0.9 %	87.9 ± 0.7 %	87.4 ± 0.9 %
Brightness -50 %	88.1 ± 0.8 %	88.5 ± 0.6 %	87.6 ± 0.7 %
Contrast +50 %	87.8 ± 0.7 %	88.0 ± 0.6 %	87.3 ± 0.9 %
Contrast -50 %	87.4 ± 1.4 %	87.5 ± 0.9 %	87.5 ± 1.0 %
Overall Score			
IoU	87.8 ± 1.1 %	88.1 ± 0.8 %	87.6 ± 0.9 %

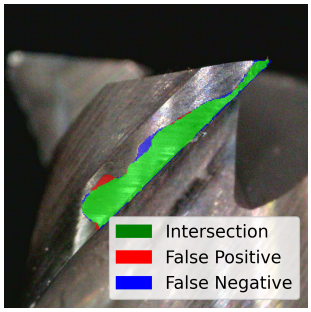
To visualise, the difference of performance, Figure 18.3 compares the predictions of the three models on the same flank wear image (see Figure 18.3a). Figure 18.3b shows the prediction from the U-Net model. The prediction is fairly good but not perfect: wear is slightly overestimated (visible as red regions on the tool), and some areas are not detected (blue regions). The predictions from the U-Net++ and DU-Net models (see Figures 18.3c and 18.3d, respectively) are very similar, with errors limited to a few isolated pixels.

In conclusion, based solely on the numbers, the segmentation analysis results indicate that U-Net++ is the most effective model overall. In practice, all three models demonstrate similar performance, with no clear winner. The quality of segmentation produced by each model is excellent, with segmentation errors typically limited to only a few pixels. To further refine the analysis and fully evaluate the approaches, it is necessary to assess the segmentation quality based on its effectiveness in accurately measuring tool wear.

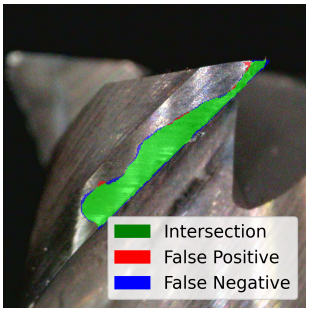
Figure 18.4 presents the same comparison as in Figure 18.3, but on a tip wear image. In this case, the differences between the three models are not visually noticeable and are limited to a few pixel-level variations.



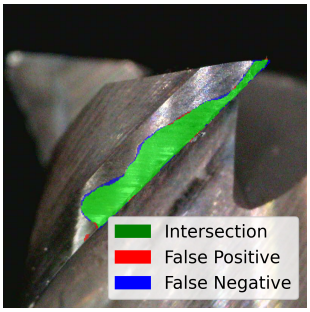
(a) Reference image



(b) U-Net Prediction



(c) U-Net++ Prediction

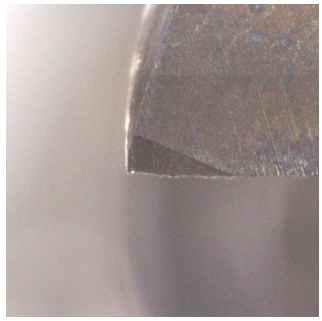


(d) DU-Net Prediction

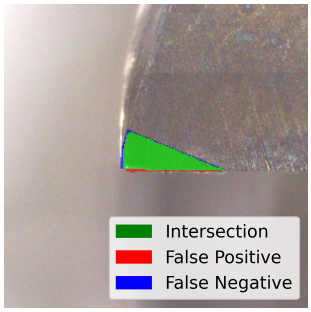
Figure 18.3: Comparison of the predictions from three models on the same flank wear image. Green areas indicate correctly predicted regions (intersection between ground truth and prediction). Red areas represent false positives (predicted regions not present in the ground truth), while blue areas highlight false negatives (ground truth regions missed by the prediction).

18.4.2 Evaluating the Wear from Segmentation

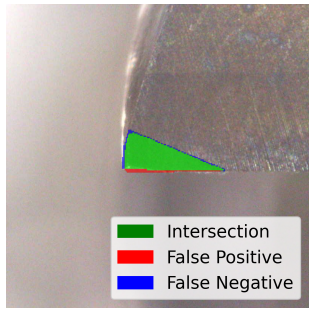
Since the image acquisition process was carried out consistently across the entire dataset, tool wear can be directly estimated from the segmentation masks generated by the models. This constitutes a post-processing step applied to the segmentation results produced by the AI models. The software developed for data management and wear measurement in the milling dataset (see Chapter 13) is reused for this purpose. Based on the wear segmentation, the software measures the size of the wear width (VB) in accordance with the ISO 8688 standard [28]. In the following, the segmentation masks corresponding to tool wear, generated by the models, are therefore used with this software. These masks are then measured and overlaid on the original images. Figure 18.5 presents the wear measurements derived from the predicted masks. Among the three models, the U-Net provides the least accurate estimation of VB_{max}, with a value of 463 µm, compared to the 423 µm measured on the image. This discrepancy is due to an overestimation of the wear area, as illustrated in Figure 18.3b. The U-Net++ and DU-Net models yield VB_{max} values of 428 µm and 418 µm, respectively, both very close to the actual value of 423 µm. As for the estimation of VB_{mean}, all three models produce similar results, within 3 µm of each other, around 255 µm, compared to the ground truth value of 240 µm.



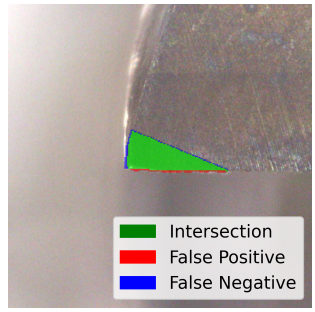
(a) Reference image



(b) U-Net Prediction



(c) U-Net++ Prediction

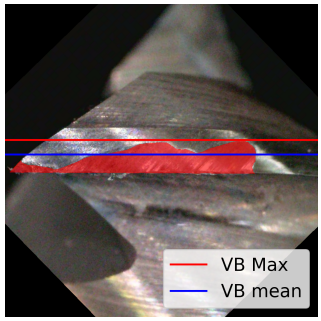


(d) DU-Net Prediction

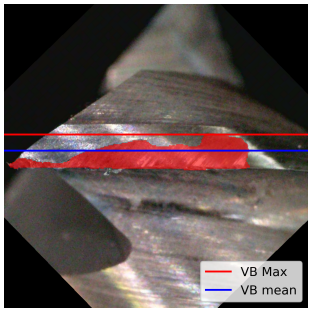
Figure 18.4: Comparison of the predictions from three models on the same tip wear image. Green areas indicate correctly predicted regions (intersection between ground truth and prediction). Red areas represent false positives (predicted regions not present in the ground truth), while blue areas highlight false negatives (ground truth regions missed by the prediction).

The error in this particular prediction is therefore around 15 µm, which remains relatively small.

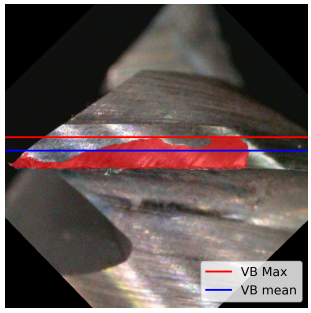
To assess the overall performance of the wear estimation, Table 18.2 presents the mean error and standard deviation of the wear predictions generated by the models across the entire test dataset. The values are averaged over 10 independent runs of each model. The average reference values for the evaluation trajectories are VB_{max} at 338.9 µm and VB_{mean} at 166.9 µm. The results show that the U-Net++ approach achieves the lowest average error, both for VB_{mean} and VB_{max} estimations. It is also worth noting that all methods yield average errors on the order of ten microns, which is an excellent result. This is particularly impressive considering that the models do not process the images at the original microscope resolution. The input images are downsampled, which can introduce errors of several 10 µm, as each pixel then corresponds to a larger physical area on the tool surface. Since the models cannot operate at sub-pixel precision, the effective resolution is limited to the pixel size. In the model's input resolution, one pixel corresponds to approximately 15.625 µm. Note that it is not inconsistent to observe deviations smaller than this value when estimating VB_{mean}, since this metric represents the average wear over the entire worn area.



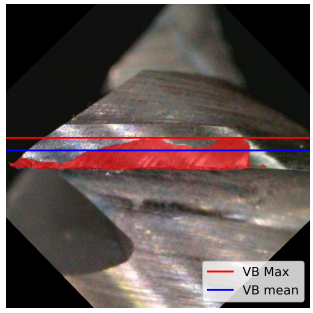
(a) True measurement:
VB_{max}: 423.6 µm
VB_{mean}: 240.9 µm



(b) U-Net Prediction:
VB_{max}: 463.1 µm
VB_{mean}: 252.7 µm



(c) U-Net++ measurement:
VB_{max}: 428.6 µm
VB_{mean}: 254.2 µm



(d) DU-Net measurement:
VB_{max}: 418.7 µm
VB_{mean}: 257.9 µm

Figure 18.5: Example of flank wear measurement based on the ground truth mask and the predicted masks.

Table 18.2: Mean error to estimate the cutting flank wear. Each value represents the mean error over 10 independent runs of the same model. The mean target value of VB_{max} is 338.9 µm and VB_{mean} is 166.9 µm.

	U-Net	**U-Net ++**	**DU-Net**
Mean error on VB_{max} [µm]	10.9 ± 25.3	8.7 ± 26.0	11.9 ± 24.2
Mean error on VB_{mean} [µm]	9.6 ± 14.8	8.1 ± 14.1	11.4 ± 15.0

18.5 Conclusions

In this chapter, U-Net, U-Net++, and DUNet models were trained to detect wear zones on cutting tools using image data. The developed models achieved high segmentation accuracy, with Intersection over Union (IoU) scores around 88 %, corresponding to an average wear estimation error of approximately 10 µm. Among the tested architectures, U-Net++ delivered the best performance, offering an optimal balance between accuracy and model complexity.

Thanks to training with augmented and altered images, the models also demonstrated strong robustness to variations in image conditions. This is particularly

valuable from an industrial perspective, where maintaining consistent lighting during image acquisition is often challenging.

It is also worth noting that an error margin of around 10 µm is practically negligible. Even manual measurements struggle to achieve such repeatability, highlighting the potential of these models for reliable and precise tool wear monitoring in real-world manufacturing environments.

Compared to the results obtained in the turning context (see Chapter 10), this chapter demonstrates excellent segmentation performance, now coupled with the ability to exploit model outputs for direct tool wear measurement. This extension complements the previous chapter by applying the measurement method end-to-end, thereby highlighting the industrial relevance of the proposed approach.

Chapter 19

General Conclusions on the Application of Tool Condition Monitoring Strategies in Milling

This Part III presented the application of TCM strategies developed in Part II for turning.

First, Chapters 13 and 14 describes the milling database developed in this thesis. It consists of the degradation of 27 cutting tools under 9 different cutting conditions. The database includes multiple sensor signals acquired during machining: cutting forces, accelerations, sound, and the power consumption of the CNC machine axes. It also contains images of the tool's cutting edges throughout the machining process until its end-of-life. A control part was also machined to keep track of the surface quality. This database, being more comprehensive than the one used for turning, allows testing various sensor combinations to monitor tool wear. It also provides a larger amount of data, enabling model evaluation across more cutting conditions.

This dataset enables the application of both direct and indirect approaches thanks to the variety of collected signals. For indirect approaches, it allows testing models with different input combinations to evaluate performance when using a single sensor or when fusing information from multiple sensors. For indirect approaches:

- Chapter 15 applies the methods developed in Chapter 7 to monitor tool degradation using different combinations of cutting signals. The use of artificial neural networks (ANNs) shows that cutting force is the most accurate sensor for monitoring tool wear. The bootstrap methods developed for turning could not be directly applied due to the significantly larger size of the milling dataset and size of the networks. Therefore, an alternative approach based on probabilistic networks was proposed, yielding similar monitoring results to ANNs but with added confidence intervals. Finally, explainability was

introduced through Explainable Boosting Machines, which highlighted the importance of cutting forces and cutting conditions in achieving optimal monitoring performance.

- Chapter 16 applies the methods developed in Chapter 8 to forecast tool degradation up to three minutes ahead. Model optimization shows that the best performance is achieved using LSTM neurons. As in turning, the forecasting model performs as well as a dedicated monitoring model in estimating the current state of tool degradation. In the milling case, the forecast horizon was set to three minutes, which is proportionally twice as long as in turning, relative to tool life. The results show that the further ahead the prediction, the lower the accuracy. However, the forecasting performance remains strong and provides useful insight into the future evolution of tool wear. The hierarchy of input effectiveness remains consistent with previous milling results, with cutting forces providing the highest forecasting performance.

From these indirect approaches, similar performance hierarchies were observed, allowing the contribution of each sensor to be quantified. In all cases, cutting force measurements yielded the best results. The next most relevant informations were acceleration and machine power, which provided satisfactory, though slightly lower, performance compared to cutting forces. Sound, on the other hand, consistently failed to produce good results across all indirect approaches. Overall, providing complementary information about cutting conditions proved beneficial to model performance.

For the direct approaches, the images from the milling dataset are used. These images were acquired under consistent orientation and lighting conditions, which helps overcome the limitations of the turning dataset. For the direct approaches:

- Chapter 17 applies the transfer learning methods developed in Chapter 9 to classify tool condition from images. For the milling dataset, transformer-based models achieved the best classification performance. This result differs from Chapter 9, where the best results were obtained using a VGG19 model. This difference is explained by the fact that the milling dataset is larger and more diverse than the turning dataset. Since transformer models require more data than convolutional approaches, the milling dataset allows them to outperform other models.

- Chapter 18 applies the encoder-decoder approaches developed in Chapter 10 to measure tool wear from images. Thanks to more consistent image quality in the milling dataset compared to turning, these approaches could be fully exploited. Tool condition was directly measured from the segmentation produced by the encoder-decoder. As in turning, the U-Net++ model achieved the best results, with an average error in the range of tens of microns for tool wear estimation from images.

This part of the document has therefore applied all TCM approaches developed in Part II to the milling process. As for turning, applying all these TCM methods to a single database is a unique contribution to the literature.

Part IV

Conclusions and Perspectives

Chapter 20

Conclusions of the Thesis and Outlooks

20.1 Summary of the Thesis Contribution for Tool Condition Monitoring and Conclusions

Throughout this thesis, multiple significant contributions have been presented, each addressing different aspects of TCM and its application in industrial machining processes.

To **validate TCM approaches against traditional time-based tool replacement**, a **simulation** was conducted to **assess the economic and environmental effects** of implementing these strategies. Although the simulation simplifies real production chains, it enabled a comparative analysis of multiple strategies. The results showed that replacing cutting tools based on monitored wear can **reduce production costs** by approximately 8 %, while also **lowering environmental impact** by around 9 %. This simulation clearly demonstrates the value of implementing TCM methods.

Before developing new TCM approaches, **a systematic literature review was conducted**. This review identified all existing TCM strategies and **highlighted several gaps in the literature**, which helped guiding the development of the methods proposed in this thesis. Knowing these literature gaps, this thesis has contributed to exploring various approaches to TCM in both turning and milling. While the turning dataset originates from a previous thesis conducted in the lab, **the milling dataset was developed** as part of this work. Consequently, all approaches could be tested in both contexts (see Figure 20.1).

For indirect monitoring regression approaches, i.e., those based on signal acquisition during machining to estimate the tool state:

- The **use of neural networks** proved to be a versatile approach, capable of adapting to different datasets and achieving monitoring results with **errors**

limited to a few tens of microns on VB. This result was valorized in a publication [123].

- Methods for **estimating prediction confidence intervals** were also tested. For turning, given the small dataset and smaller networks, **a bootstrap method was employed**. For milling, this approach was not feasible due to scalability concerns. However, **a probabilistic neural network** approach was successfully developed. In both cases, confidence intervals were obtained, evolving consistently with the inherent uncertainties of the manufacturing process. The bootstrap results was valorized in a publication [69]

- Finally, **explainability techniques** using Explainable Boosting Machines **were implemented**. This approach proved particularly convincing, as it achieved results comparable to neural networks while providing interpretability of the monitoring. This approach was valorized in a publication [136].

- Although none of the developed methods can simultaneously provide both confidence intervals and explainability, it is proposed to **use these two methods in parallel** to ensure the **highest possible reliability** in tool condition monitoring.

For **indirect forecasting regression** approaches, i.e., those aiming to predict the future evolution of the tool based on signals collected during machining:

- This approach represents an **original contribution** of the thesis, addressing a gap in the literature where very few studies focus on the future evolution of tool condition. The results are being prepared for a future publication.

- Methods based on **recurrent neural networks** were developed and demonstrated strong predictive capabilities, achieving very good performance, as **prediction errors** over the forecast horizon were **limited to a few microns**.

- In turning and milling operations, real-time estimation of tool condition is **as accurate as** that provided by **a dedicated monitoring model**. In other words, using a predictive model does not compromise the effectiveness of tool condition monitoring.

For **direct classification approaches**, i.e., methods that use images of the tool to determine the class to which it belongs:

- **To classify tool images, transfer learning techniques were employed**. These approaches leverage models such as VGG19, EfficientNetV2, and Vision Transformers. On the turning dataset, which contains relatively few images, VGG19 proved to be the most effective approach for classifying tool condition. For milling, where the dataset is larger, Vision Transformers achieved the best performance. In both cases, **classification accuracy was around** 90 %, with most errors occurring in transition zones between classes, corresponding to deviations of approximately 25 µm. This approach was valorized in a publication [156].

Finally, for **direct monitoring regression** approaches, i.e., those that use tool images to directly measure wear:

- **Encoder–decoder-based methods** of varying depth **were developed**. For both turning and milling, the approaches achieved similar scores, but U-Net++ architectures appeared to deliver better segmentation results.
- Measuring tool condition from the segmentation produced by the models could not be completed for the turning dataset, as its images were not homogeneous. For milling, segmentation enabled **tool wear measurement** with an accuracy on the order of **tens of microns**.
- For both datasets, the segmentation generated by the models was **more precise than manual segmentation**.

This thesis has made it possible to develop a set of solutions adapted to several TCM strategies. All the proposed approaches were tested under both turning and milling conditions, demonstrating their validity, scalability, and adaptability. In the context of Industry 4.0 and the increasing availability of real-time machining data, this work highlights the importance of using appropriate methods to assist practitioners in deciding whether or not to replace a cutting tool. Since each approach has its own advantages and limitations, it is ultimately up to the practitioner to select the one that best suits their specific needs.

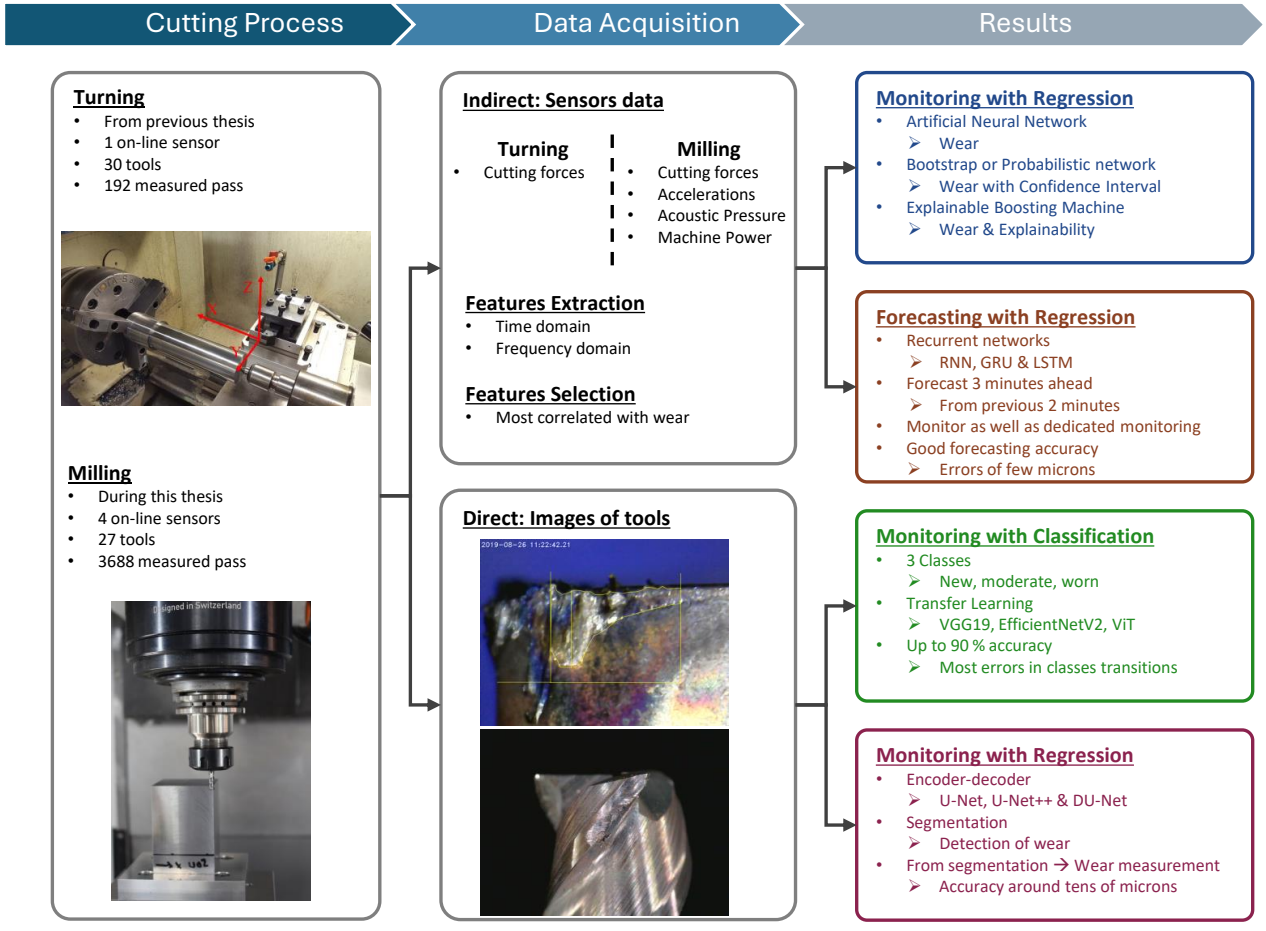


Figure 20.1: Schematic overview of all topics addressed in the thesis and their key conclusions.

20.2 Outlooks

A thesis is just the starting point of a broader research effort, and several aspects remain to be explored. Some outlooks can be suggested:

- The databases used in this thesis are limited to laboratory machining conditions. An approach incorporating industrial conditions could help assess the practical applicability of the developed methods.
- The thesis focused solely on the machining of C45 steel. Therefore, multi-material approaches could not be evaluated. A likely approach would be to use the specific cutting pressure to generalize the models, but this could not be tested during the thesis. However, it is still unclear how to create more general models trained on multiple materials. In particular, how to characterize the materials so that the models can adapt their predictions on a case-by-case basis.
- All the results of this thesis were obtained offline, without specific constraints on computation time. To deploy these models in edge or fog computing environments, a reflection on their size and required computational power is necessary.
- The thesis focuses solely on estimating tool condition. Integrating this indicator into advanced maintenance policies tailored to different types of workpieces could provide industry with a clearer understanding of how to apply these methods. However, the relationship between workpiece quality and tool condition was not addressed in this thesis. Future research could explore the correlation between the cutting tool's condition and the quality of the produced surface.
- Deep learning approaches in indirect approaches were not addressed in this document. However, deep learning techniques for automatic feature extraction could be considered in future work.
- The methods were developed for monitoring a manufacturing process. It is conceivable to apply them to other processes beyond milling and turning, for example, in grinding or even in industrial 3D printing.

Bibliography

[1] G. de la banque mondiale. "Banque mondiale." (), [Online]. Available: https://www.banquemondiale.org/ext/fr/home (visited on 2026).

[2] N. Haraguchi, C. F. C. Cheng, and E. Smeets, "The importance of manufacturing in economic development: Has this changed?" *World Development*, vol. 93, pp. 293–315, 2017, ISSN: 0305750X. DOI: 10.1016/j.worlddev.2016.12.013.

[3] F. Klocke, *Manufacturing Processes 1* (RWTHedition). Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, ISBN: 978-3-642-11978-1. DOI: 10.1007/978-3-642-11979-8.

[4] C. Lauro, L. Brandão, D. Baldo, R. Reis, and J. Davim, "Monitoring and processing signal applied in machining processes – a review," *Measurement*, vol. 58, pp. 73–86, 2014, ISSN: 02632241. DOI: 10.1016/j.measurement.2014.08.035.

[5] D. Goyal and B. Pabla, "Condition based maintenance of machine tools—a review," *CIRP Journal of Manufacturing Science and Technology*, vol. 10, pp. 24–35, 2015, ISSN: 17555817. DOI: 10.1016/j.cirpj.2015.05.004.

[6] L. Equeter, F. Ducobu, E. Rivière-Lorphèvre, R. Serra, and P. Dehombreux, "An analytic approach to the cox proportional hazards model for estimating the lifespan of cutting tools," *Journal of Manufacturing and Materials Processing*, vol. 4, no. 27, pp. 1–13, 2020. DOI: 10.3390/jmmp4010027.

[7] A. E. Diniz, Á. R. Machado, and J. G. Corrêa, "Tool wear mechanisms in the machining of steels and stainless steels," *The International Journal of Advanced Manufacturing Technology*, vol. 87, no. 9, pp. 3157–3168, 2016, ISSN: 0268-3768, 1433-3015. DOI: 10.1007/s00170-016-8704-3.

[8] P. Gowthaman, S. Jeyakumar, and B. Saravanan, "Machinability and tool wear mechanism of duplex stainless steel – a review," *Materials Today: Proceedings*, vol. 26, pp. 1423–1429, 2020, ISSN: 22147853. DOI: 10.1016/j.matpr.2020.02.295.

[9] G. Krolczyk *et al.*, "Ecological trends in machining as a key factor in sustainable production – a review," *Journal of Cleaner Production*, vol. 218, pp. 601–615, 2019, ISSN: 09596526. DOI: 10.1016/j.jclepro.2019.02.017.

[10] C. Agrawal, J. Wadhwa, A. Pitroda, C. I. Pruncu, M. Sarikaya, and N. Khanna, "Comprehensive analysis of tool wear, tool life, surface roughness, costing and carbon emissions in turning ti–6al–4v titanium alloy: Cryogenic versus wet machining," *Tribology International*, vol. 153, p. 106 597, 2021, ISSN: 0301679X. DOI: 10.1016/j.triboint.2020.106597.

[11] Z. Said *et al.*, "A comprehensive review on minimum quantity lubrication (MQL) in machining processes using nano-cutting fluids," *The International Journal of Advanced Manufacturing Technology*, vol. 105, no. 5, pp. 2057–2086, 2019, ISSN: 0268-3768, 1433-3015. DOI: 10.1007/s00170-019-04382-x.

[12] W. Xu and L. Cao, "Optimal tool replacement with product quality deterioration and random tool failure," *International Journal of Production Research*, vol. 53, no. 6, pp. 1736–1745, 2015, ISSN: 0020-7543, 1366-588X. DOI: 10.1080/00207543.2014.957878.

[13] J. G. Wager and M. M. Barash, "Study of the distribution of the life of HSS tools," *Journal of Engineering for Industry*, vol. 93, no. 4, pp. 1044–1050, 1971, ISSN: 0022-0817. DOI: 10.1115/1.3428041.

[14] Z. Vagnorius, M. Rausand, and K. Sørby, "Determining optimal replacement time for metal cutting tools," *European Journal of Operational Research*, vol. 206, no. 2, pp. 407–416, 2010, ISSN: 03772217. DOI: 10.1016/j.ejor.2010.03.023.

[15] Y. Liu, X. Hu, and J. Jin, "Remaining useful life prediction of cutting tools based on deep adversarial transfer learning," in *Proceedings of the 2019 8th International Conference on Computing and Pattern Recognition*, 2019, pp. 434–439. DOI: 10.1145/3373509.3373543.

[16] R. U. Baig, S. Javed, M. Khaisar, M. Shakoor, and P. Raja, "Development of an ANN model for prediction of tool wear in turning EN9 and EN24 steel alloy," *Advances in Mechanical Engineering*, vol. 13, no. 6, p. 168 781 402 110 267, 2021, ISSN: 1687-8140, 1687-8140. DOI: 10.1177/16878140211026720.

[17] L. Equeter, "Optimal replacement of cutting tools through degradation monitoring techniques and residual lifetime prediction models," Ph.D. dissertation, University of Mons, 2020.

[18] "Machinability and surface integrity," in *Cutting Tool Technology*, London: Springer London, 2008, pp. 269–379, ISBN: 978-1-84800-204-3 978-1-84800-205-0. DOI: 10.1007/978-1-84800-205-0_7.

[19] M. Iliyas Ahmad, Y. Yusof, M. E. Daud, K. Latiff, A. Z. Abdul Kadir, and Y. Saif, "Machine monitoring system: A decade in review," *The International Journal of Advanced Manufacturing Technology*, vol. 108, no. 11, pp. 3645–3659, 2020, ISSN: 0268-3768, 1433-3015. DOI: 10.1007/s00170-020-05620-3.

[20] B. Kilundu, "Détection et diagnostic vibratoire de dégradations mécaniques : Apport du data mining," Ph.D. dissertation, University of Mons, 2008.

[21] O. Basile, "Prise en compte de l'incertitude dans les modèles fiabilistes de maintenance industrielle. extension aux sollicitations variables.," Ph.D. dissertation, University of Mons, 2007.

[22] C. Letot, "Maintenance prévisionnelle d'équipements industriels basée sur la modélisation, l'estimation et la simulation de lois de dégradation," Ph.D. dissertation, University of Mons, 2013.

[23] A. Muller, A. Crespo Marquez, and B. Iung, "On the concept of e-maintenance: Review and current research," *Reliability Engineering & System Safety*, vol. 93, no. 8, pp. 1165–1187, 2008, ISSN: 09518320. DOI: 10.1016/j.ress.2007.08.006.

[24] F. Klocke, *Manufacturing Processes 1* (RWTHedition). Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, ISBN: 978-3-642-11978-1. DOI: 10.1007/978-3-642-11979-8.

[25] Sandvik Coromant, *Training Handbook - Metal Cutting Technology*. AB Sandvik Coromant, 2017.

[26] Seco Tools, *Turning catalog and technical guide*, 2nd ed. SECO TOOLS AB, 2018.

[27] "IS0 3685 - Tool life testing with single-point turning tools," 1993.

[28] International Organization for Standardization, *ISO 8688-2:1989 tool life testing in milling - end milling*, 1989.

[29] D. T. Pham and P. T. Pham, "Artificial intelligence in engineering," *International Journal of Machine Tools and Manufacture*, vol. 39, no. 6, pp. 937–949, 1999, ISSN: 08906955. DOI: 10.1016/S0890-6955(98)00076-5.

[30] *Oxford dictionnary*. [Online]. Available: https://www.oxfordlearnersdictionaries.com/.

[31] J. Mccarthy, "What is Artificial Intelligence?," pp. 1–14, 2004.

[32] I. El Naqa, R. Li, and M. J. Murphy, Eds., *Machine Learning in Radiation Oncology*. Cham: Springer International Publishing, 2015, ISBN: 978-3-319-18304-6. DOI: 10.1007/978-3-319-18305-3.

[33] L. Colantonio, L. Equeter, P. Dehombreux, and F. Ducobu, "A systematic literature review of cutting tool wear monitoring in turning by using artificial intelligence techniques," *Machines*, vol. 9, no. 12, p. 351, 2021, ISSN: 2075-1702. DOI: 10.3390/machines9120351.

[34] A. Abraham, "Artificial Neural Networks," in *Handbook of Measuring System Design*, Chichester, UK: John Wiley & Sons, Ltd, 2005, ISBN: 0470021438. DOI: 10.1002/0471497398.mm421.

[35] E. Fiesler and R. Beale, *Handbook of neural computation*. Bristol, Philadelphia and New York: IOP Publishing and Oxford University Press, 1996, ISBN: 0-7503-0312-3.

[36] J. Wen, S. Li, Z. Lin, Y. Hu, and C. Huang, "Systematic literature review of machine learning based software development effort estimation models," *Information and Software Technology*, vol. 54, no. 1, pp. 41–59, 2012. DOI: 10.1016/j.infsof.2011.09.002.

[37] C. S. H. (U. H. Ontario, "Neural network-based receiver for wireless communications," *Electronics Letters*, vol. 35, no. 3, p. 203, 1999, ISSN: 00135194. DOI: 10.1049/el:19990177.

[38] O. A. Montesinos López, A. Montesinos López, and J. Crossa, *Multivariate Statistical Machine Learning Methods for Genomic Prediction*. Cham: Springer International Publishing, 2022, ISBN: 978-3-030-89009-4 978-3-030-89010-0. DOI: 10.1007/978-3-030-89010-0.

[39] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986, ISSN: 0028-0836, 1476-4687. DOI: 10.1038/323533a0.

[40] L. Bottou, F. E. Curtis, and J. Nocedal, "Optimization methods for large-scale machine learning," *SIAM Review*, vol. 60, no. 2, pp. 223–311, 2018, ISBN: 9781665413640 _eprint: 1606.04838, ISSN: 0036-1445. DOI: 10.1137/16M1080173.

[41] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convolutional neural networks: Analysis, applications, and prospects," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 12, pp. 6999–7019, 2022, Publisher: Institute of Electrical and Electronics Engineers (IEEE), ISSN: 2162-237X, 2162-2388. DOI: 10.1109/tnnls.2021.3084827.

[42] K. O'Shea and R. Nash, "An Introduction to Convolutional Neural Networks," pp. 1–11, 2015. arXiv: 1511.08458.

[43] A. Vaswani *et al.*, *Attention is all you need*, Version Number: 7, 2017. DOI: 10.48550/ARXIV.1706.03762.

[44] A. Dosovitskiy *et al.*, *An image is worth 16x16 words: Transformers for image recognition at scale*, Version Number: 2, 2020. DOI: 10.48550/ARXIV.2010.11929.

[45] Y. Lou, R. Caruana, J. Gehrke, and G. Hooker, "Accurate intelligible models with pairwise interactions," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, Chicago Illinois USA: ACM, 2013, pp. 623–631, ISBN: 978-1-4503-2174-7. DOI: 10.1145/2487575.2487579.

[46] A. Paszke *et al.*, *PyTorch: An imperative style, high-performance deep learning library*, Version Number: 1, 2019. DOI: 10.48550/ARXIV.1912.01703.

[47] T. Developers, *TensorFlow*, version v2.20.0, 2025. DOI: 10.5281/ZENODO.4724125.

[48] T. pandas development team, *Pandas-dev/pandas: Pandas*, version v2.3.2, 2025. DOI: 10.5281/ZENODO.3509134.

[49] C. R. Harris *et al.*, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, 2020, ISSN: 0028-0836, 1476-4687. DOI: 10.1038/s41586-020-2649-2.

[50] P. Virtanen *et al.*, "SciPy 1.0: Fundamental algorithms for scientific computing in python," *Nature Methods*, vol. 17, no. 3, pp. 261–272, 2020, ISSN: 1548-7091, 1548-7105. DOI: 10.1038/s41592-019-0686-2.

[51] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007, ISSN: 1521-9615. DOI: 10.1109/MCSE.2007.55.

[52] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, Version Number: 9, 2014. DOI: 10.48550/ARXIV.1412.6980.

[53] M. M. Bejani and M. Ghatee, *A systematic review on overfitting control in shallow and deep neural networks*. Springer Netherlands, 2021, vol. 54, 6391–6438, ISSN: 15737462 Publication Title: Artificial Intelligence Review, ISBN: 0-12-345678-9. DOI: 10.1007/s10462-021-09975-1.

[54] L. Colantonio, G. Moroncini, O. Sénéchal, P. Dehombreux, F. Ducobu, and L. Equeter, "Cost and environmental assessment of tool replacement strategies under imperfect wear monitoring in ti6al4v milling," *International Journal of Prognostics and Health Management*, Under review.

[55] A. Furberg, R. Arvidsson, and S. Molander, "Environmental life cycle assessment of cemented carbide (WC-co) production," *Journal of Cleaner Production*, vol. 209, pp. 1126–1138, 2019, ISSN: 09596526. DOI: 10.1016/j.jclepro.2018.10.272.

[56] I. Fairoz and A. Shokrani, "Life cycle assessment of cutting tool coatings (in press)," *Procedia CIRP*, 2025.

[57] Electricity Maps. "Electricity maps - belgium, january 2025," Electricity Maps. (2025), [Online]. Available: https://app.electricitymaps.com/zone/BE/12mo/monthly (visited on 2025).

[58] I. García Gutiérrez, C. Pina, R. Tobajas, and D. Elduque, "Incorporating composition into life cycle assessment of steel grades," *Journal of Cleaner Production*, vol. 472, p. 143 538, 2024, ISSN: 09596526. DOI: 10.1016/j.jclepro.2024.143538.

[59] A. D. La Rosa, I. Åsen, G. Ingarao, and L. Fratini, "Life cycle assessment of a steel component produced from electric arc furnace," *CIRP Journal of Manufacturing Science and Technology*, vol. 59, pp. 127–134, 2025, ISSN: 17555817. DOI: 10.1016/j.cirpj.2025.03.007.

[60] J. Suer, M. Traverso, and N. Jäger, "Review of life cycle assessments for steel and environmental analysis of future steel production scenarios," *Sustainability*, vol. 14, no. 21, p. 14 131, 2022, ISSN: 2071-1050. DOI: 10.3390/su142114131.

[61] P. Renzulli, B. Notarnicola, G. Tassielli, G. Arcese, and R. Di Capua, "Life cycle assessment of steel produced in an italian integrated steel mill," *Sustainability*, vol. 8, no. 8, p. 719, 2016, ISSN: 2071-1050. DOI: 10.3390/su8080719.

[62] Hoffmann Group. "Hoffmann group," Hoffmann Group. (), [Online]. Available: https://www.hoffmann-group.com/ (visited on 2025).

[63] M. Aramesh, Y. Shaban, S. Yacout, M. H. Attia, H. A. Kishawy, and M. Balazinski, "Survival life analysis applied to tool life estimation with variable cutting conditions when machining titanium metal matrix composites (ti-MMCs)," *Machining Science and Technology*, vol. 20, no. 1, pp. 132–147, 2016, ISSN: 1091-0344, 1532-2483. DOI: 10.1080/10910344.2015.1133916.

[64] B. H. Nystad, G. Gola, and J. E. Hulsund, "Lifetime models for remaining useful life estimation with randomly distributed failure thresholds," *PHM Society European Conference*, vol. 1, no. 1, 2012, ISSN: 2325-016X, 2325-016X. DOI: 10.36001/phme.2012.v1i1.1442.

[65] Pakshal Steel. "Titanium price per kg," Pakshal Steel. (2025), [Online]. Available: https://www.pakshalsteel.com/titanium-metal-price.html (visited on 2025).

[66] M. Triebe, G. Mendis, F. Zhao, and J. Sutherland, "Understanding energy consumption in a machine tool through energy mapping," *Procedia CIRP*, vol. 69, pp. 259–264, 2018, ISSN: 22128271. DOI: 10.1016/j.procir.2017.11.041.

[67] J. B. Dahmus and T. G. Gutowski, "An environmental analysis of machining," in *Manufacturing Engineering and Materials Handling Engineering*, Anaheim, California, USA: ASMEDC, 2004, pp. 643–652, ISBN: 978-0-7918-4713-8. DOI: 10.1115/IMECE2004-62600.

[68] D. Fratila, "Sustainable manufacturing through environmentally-friendly machining," in *Green Manufacturing Processes and Systems*, J. P. Davim, Ed., Series Title: Materials Forming, Machining and Tribology, Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 1–21, ISBN: 978-3-642-33791-8 978-3-642-33792-5. DOI: 10.1007/978-3-642-33792-5_1.

[69] L. Colantonio, L. Equeter, P. Dehombreux, and F. Ducobu, "Confidence interval estimation for cutting tool wear prediction in turning using bootstrap-based artificial neural networks," *Sensors*, vol. 24, no. 11, p. 3432, 2024, ISSN: 1424-8220. DOI: 10.3390/s24113432.

[70] T. Wong, W. Kim, and P. Kwon, "Experimental support for a model-based prediction of tool wear," *Wear*, vol. 257, no. 7, pp. 790–798, 2004, ISSN: 00431648. DOI: 10.1016/j.wear.2004.03.010.

[71] B. Kitchenham, O. Pearl Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, "Systematic literature reviews in software engineering – A systematic literature review," *Information and Software Technology*, vol. 51, no. 1, pp. 7–15, 2009, ISSN: 09505849. DOI: 10.1016/j.infsof.2008.09.009.

[72] B. Kitchenham, "Procedures for performing a systematic reviews," Keele University, Keele, Staffs, UK, Tech. Rep., 2004.

[73] M. J. Page *et al.*, "The PRISMA 2020 statement: An updated guideline for reporting systematic reviews," *BMJ*, n71, 2021, ISSN: 1756-1833. DOI: 10.1136/bmj.n71.

[74] *Scopus*. [Online]. Available: https://www.elsevier.com/solutions/scopus/why-choose-scopus.

[75] B. Sick, "On-line and inderect tool wear monitoring in turning with artificial neural networks: A review of more than a decade of research," *Mechanical Systems and Signal Processing*, vol. 16, no. 4, pp. 487–546, 2002, ISSN: 08883270. DOI: 10.1006/mssp.2001.1460.

[76] J. R. Landis and G. G. Koch, "The Measurement of Observer Agreement for Categorical Data," *Biometrics*, vol. 33, no. 1, p. 159, 1977. DOI: 10.2307/2529310.

[77] A. Siddhpura and R. Paurobally, "A review of flank wear prediction methods for tool condition monitoring in a turning process," *The International Journal of Advanced Manufacturing Technology*, vol. 65, no. 1-4, pp. 371–393, 2013, ISSN: 0268-3768. DOI: 10.1007/s00170-012-4177-1.

[78] M. Cheng, L. Jiao, X. Shi, X. Wang, P. Yan, and Y. Li, "An intelligent prediction model of the tool wear based on machine learning in turning high strength steel," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 234, no. 13, pp. 1580–1597, 2020, ISSN: 0954-4054. DOI: 10.1177/0954405420935787.

[79] C. Scheffer and P. Heyns, "Wear monitoring in turning operations using vibration and strain measurements," *Mechanical Systems and Signal Processing*, vol. 15, no. 6, pp. 1185–1202, 2001. DOI: 10.1006/mssp.2000.1364.

[80] M. Du, P. Wang, J. Wang, Z. Cheng, and S. Wang, "Intelligent Turning Tool Monitoring with Neural Network Adaptive Learning," *Complexity*, vol. 2019, pp. 1–21, 2019, ISSN: 1076-2787. DOI: 10.1155/2019/8431784.

[81] T. Segreto, D. D'Addona, and R. Teti, "Tool wear estimation in turning of Inconel 718 based on wavelet sensor signal analysis and machine learning paradigms," *Production Engineering*, vol. 14, no. 5-6, pp. 693–705, 2020, ISSN: 0944-6524. DOI: 10.1007/s11740-020-00989-2.

[82] T. SK, S. Shankar, M. T, and D. K, "Tool wear prediction in hard turning of EN8 steel using cutting force and surface roughness with artificial neural network," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 234, no. 1, pp. 329–342, 2020, ISSN: 0954-4062. DOI: 10.1177/0954406219873932.

[83] P. Twardowski and M. Wiciak-Pikuła, "Prediction of Tool Wear Using Artificial Neural Networks during Turning of Hardened Steel," *Materials*, vol. 12, no. 19, p. 3091, 2019, ISSN: 1996-1944. DOI: 10.3390/ma12193091.

[84] A. Caggiano, "Tool Wear Prediction in Ti-6Al-4V Machining through Multiple Sensor Monitoring and PCA Features Pattern Recognition," *Sensors*, vol. 18, no. 3, p. 823, 2018, ISSN: 1424-8220. DOI: 10.3390/s18030823.

[85] S. Karam, P. Centobelli, D. M. D'Addona, and R. Teti, "Online Prediction of Cutting Tool Life in Turning via Cognitive Decision Making," *Procedia CIRP*, vol. 41, pp. 927–932, 2016, ISSN: 22128271. DOI: 10.1016/j.procir.2016.01.002.

[86] S. Zare Chavoshi, "Tool flank wear prediction in CNC turning of 7075 AL alloy SiC composite," *Production Engineering*, vol. 5, no. 1, pp. 37–47, 2011, ISSN: 0944-6524. DOI: 10.1007/s11740-010-0282-x.

[87] P. Palanisamy and S. Shanmugasundaram, "Modelling of tool wear and surface roughness in hard turning using regression and artificial neural network," *International Journal of Machining and Machinability of Materials*, vol. 4, no. 1, p. 76, 2008, ISSN: 1748-5711. DOI: 10.1504/IJMMM.2008.020912.

[88] C. H. S. Rao, D. N. Rao, and R. N. S. Rao, "Online prediction of diffusion wear on the flank through tool tip temperature in turning using artificial neural networks," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 220, no. 12, pp. 2069–2076, 2006, ISSN: 0954-4054. DOI: 10.1243/09544054JEM541.

[89] S. Tamang and M. Chandrasekaran, "Modeling and optimization of parameters for minimizing surface roughness and tool wear in turning Al/SiCp MMC, using conventional and soft computing techniques," *Advances in Production Engineering and Management*, vol. 10, no. 2, pp. 59–72, 2015, ISSN: 18546250. DOI: 10.14743/apem2015.2.192.

[90] A. Caggiano, F. Napolitano, and R. Teti, "Dry Turning of Ti6Al4V: Tool Wear Curve Reconstruction Based on Cognitive Sensor Monitoring," *Procedia CIRP*, vol. 62, pp. 209–214, 2017, ISSN: 22128271. DOI: 10.1016/j.procir.2017.03.046.

[91] C. S. Rao and R. R. Srikant, "Tool wear monitoring—an intelligent approach," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 218, no. 8, pp. 905–912, 2004, ISSN: 0954-4054. DOI: 10.1243/0954405041486028.

[92] S. M. Ji, L. Zhang, Y. Wan, X. Zhang, J. L. Yuan, and L. Zhang, "Research of Intelligent Tool Condition Monitoring Technology Based on Multi-Information Fusion," *Materials Science Forum*, vol. 471-472, pp. 187–191, 2004, ISSN: 1662-9752. DOI: 10.4028/www.scientific.net/MSF.471-472.187.

[93] R. R. Srikant, P. V. Krishna, and N. D. Rao, "Online tool wear prediction in wet machining using modified back propagation neural network," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 225, no. 7, pp. 1009–1018, 2011, ISSN: 0954-4054. DOI: 10.1177/0954405410395854.

[94] C. Scheffer, H. Engelbrecht, and P. S. Heyns, "A comparative evaluation of neural networks and hidden Markov models for monitoring turning tool wear," *Neural Computing and Applications*, vol. 14, no. 4, pp. 325–336, 2005, ISSN: 0941-0643. DOI: 10.1007/s00521-005-0469-9.

[95] M. Balazinski, E. Czogala, K. Jemielniak, and J. Leski, "Tool condition monitoring using artificial intelligence methods," *Engineering Applications of Artificial Intelligence*, vol. 15, no. 1, pp. 73–80, 2002, ISSN: 09521976. DOI: 10.1016/S0952-1976(02)00004-0.

[96] S. Choudhury and G. Bartarya, "Role of temperature and surface finish in predicting tool wear using neural network and design of experiments," *International Journal of Machine Tools and Manufacture*, vol. 43, no. 7, pp. 747–753, 2003, ISSN: 08906955. DOI: 10.1016/S0890-6955(02)00166-9.

[97] A. Salimi asl, A. Erdem, and M. Rafighi, "Applying a multi sensor system to predict and simulate the tool wear using of artificial neural networks," *Scientia Iranica*, vol. 24, no. 6, 2017, ISSN: 2345-3605. DOI: 10.24200/sci.2017.4247.

[98] P. S. Paul and A. Varadarajan, "ANN assisted sensor fusion model to predict tool wear during hard turning with minimal fluid application," *International Journal of Machining and Machinability of Materials*, vol. 13, no. 4, p. 398, 2013, ISSN: 1748-5711. DOI: 10.1504/IJMMM.2013.054263.

[99] A. Cakan, F. Evrendilek, and V. Ozkaner, "Data-driven simulations of flank wear of coated cutting tools in hard turning," *Mechanika*, vol. 21, no. 6, pp. 486–492, 2016, ISSN: 2029-6983. DOI: 10.5755/j01.mech.21.6.12199.

[100] W.-Y. Chang, S.-J. Wu, and J.-W. Hsu, "Investigated iterative convergences of neural network for prediction turning tool wear," *The International Journal of Advanced Manufacturing Technology*, vol. 106, no. 7-8, pp. 2939–2948, 2020. DOI: 10.1007/s00170-019-04821-9.

[101] L. Kang, X. Xiong, L. Yi, and Y. Guo, "A Study of Cutting Tool Wear Prediction Utilizing Generalized Regression Neural Network with Improved Fruit Fly Optimization," in *2018 Prognostics and System Health Management Conference (PHM-Chongqing)*, 2018, pp. 1–7, ISBN: 978-1-5386-5380-7. DOI: 10.1109/PHM-Chongqing.2018.00008.

[102] R. Silva and A. Araújo, "A Novel Approach to Condition Monitoring of the Cutting Process Using Recurrent Neural Networks," *Sensors*, vol. 20, no. 16, p. 4493, 2020, ISSN: 1424-8220. DOI: 10.3390/s20164493.

[103] T. Mikołajczyk, K. Nowicki, A. Bustillo, and D. Yu Pimenov, "Predicting tool life in turning operations using neural networks and image processing," *Mechanical Systems and Signal Processing*, vol. 104, pp. 503–513, 2018, ISSN: 08883270. DOI: 10.1016/j.ymssp.2017.11.022.

[104] D. Ojha and U. Dixit, "An economic and reliable tool life estimation procedure for turning," *The International Journal of Advanced Manufacturing Technology*, vol. 26, no. 7-8, pp. 726–732, 2005, ISSN: 0268-3768. DOI: 10.1007/s00170-003-2049-4.

[105] M. A. Xavior and S. M. Anouncia, "Artificial neural networks and case-based reasoning models for predicting tool life and tool-shim interface temperature," *International Journal of Services and Operations Management*, vol. 24, no. 3, p. 379, 2016, ISSN: 1744-2370. DOI: 10.1504/IJSOM.2016.076906.

[106] N. Senthilkumar and T. Tamizharasan, "Flank wear and surface roughness prediction in hard turning via artificial neural network and multiple regressions," *Australian Journal of Mechanical Engineering*, vol. 13, no. 1, pp. 31–45, 2015, ISSN: 1448-4846. DOI: 10.7158/M13-045.2015.13.1.

[107] D. Rajeev, D. Dinakaran, N. Kanthavelkumaran, and N. Austin, "Predictions of Tool Wear in Hard Turning of AISI4140 Steel through Artificial Neural Network, Fuzzy Logic and Regression Models," *International Journal of Engineering*, vol. 31, no. 1, pp. 32–37, 2018, ISSN: 17281431. DOI: 10.5829/ije.2018.31.01a.05.

[108] A. de Farias, S. L. R. de Almeida, S. Delijaicov, V. Seriacopi, and E. C. Bordinassi, "Simple machine learning allied with data-driven methods for monitoring tool wear in machining processes," *The International Journal of Advanced Manufacturing Technology*, vol. 109, no. 9-12, pp. 2491–2501, 2020, ISSN: 0268-3768. DOI: 10.1007/s00170-020-05785-x.

[109] *TensorFlow*. [Online]. Available: https://www.tensorflow.org/?hl=en.

[110] *Scikit-learn*. [Online]. Available: https://scikit-learn.org/stable/.

[111] P. Adler *et al.*, "Auditing black-box models for indirect influence," *Knowledge and Information Systems*, vol. 54, no. 1, pp. 95–122, 2018. DOI: 10.1007/s10115-017-1116-3.

[112] G. Ras, M. van Gerven, and P. Haselager, "Explanation Methods in Deep Learning: Users, Values, Concerns and Challenges," in February 2021, 2018, pp. 19–36, ISBN: 9783319981314. DOI: 10.1007/978-3-319-98131-4_2. arXiv: 1803.07517.

[113] H. Miao, Z. Zhao, C. Sun, B. Li, and R. Yan, "A U-Net-Based Approach for Tool Wear Area Detection and Identification," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–10, 2021, ISSN: 0018-9456. DOI: 10.1109/TIM.2020.3033457.

[114] L. Pagani, P. Parenti, S. Cataldo, P. J. Scott, and M. Annoni, "Indirect cutting tool wear classification using deep learning and chip colour analysis," *The International Journal of Advanced Manufacturing Technology*, vol. 111, no. 3-4, pp. 1099–1114, 2020, ISSN: 0268-3768. DOI: 10.1007/s00170-020-06055-6.

[115] H. Mamledesai, M. A. Soriano, and R. Ahmad, "A Qualitative Tool Condition Monitoring Framework Using Convolution Neural Network and Transfer Learning," *Applied Sciences*, vol. 10, no. 20, p. 7298, 2020, ISSN: 2076-3417. DOI: 10.3390/app10207298.

[116] *Nasa*. [Online]. Available: https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/%7B%5C#%7Dmilling.

[117] *PHM society*. [Online]. Available: https://phmsociety.org/phm%7B%5C_%7Dcompetition/2010-phm-society-conference-data-challenge/.

[118] D. Y. Pimenov, A. Bustillo, S. Wojciechowski, V. S. Sharma, M. K. Gupta, and M. Kuntoğlu, "Artificial intelligence systems for tool condition monitoring in machining: Analysis and critical review," *Journal of Intelligent Manufacturing*, vol. 34, no. 5, pp. 2079–2121, 2023, ISSN: 0956-5515, 1572-8145. DOI: 10.1007/s10845-022-01923-2.

[119] D. Y. Pimenov, L. R. R. Da Silva, A. Ercetin, O. Der, T. Mikolajczyk, and K. Giasin, "State-of-the-art review of applications of image processing techniques for tool condition monitoring on conventional machining processes," *The International Journal of Advanced Manufacturing Technology*, vol. 130, no. 1, pp. 57–85, 2024, ISSN: 0268-3768, 1433-3015. DOI: 10.1007/s00170-023-12679-1.

[120] S. Kasiviswanathan, S. Gnanasekaran, M. Thangamuthu, and J. Rakkiyannan, "Machine-learning- and internet-of-things-driven techniques for monitoring tool wear in machining process: A comprehensive review," *Journal of Sensor and Actuator Networks*, vol. 13, no. 5, p. 53, 2024, ISSN: 2224-2708. DOI: 10.3390/jsan13050053.

[121] A. Vaswani *et al.*, "Attention is all you need," DOI: https://doi.org/10.48550/arXiv.1706.03762.

[122] L. Colantonio, L. Equeter, P. Dehombreux, and F. Ducobu, "Comparison of cutting tool wear classification performance with artificial intelligence techniques," *Materials Research Proceedings*, vol. 28, pp. 1265–1274, 2023. DOI: 10.21741/9781644902479-137.

[123] L. Colantonio, "Comparison of artificial intelligence techniques for cutting tool condition monitoring," presented at the Material Forming, 2024, pp. 1962–1971. DOI: 10.21741/9781644903131-217.

[124] Seco Tools, *Turning catalog and technical guide*, 2nd ed. SECO TOOLS AB, 2018.

[125] C. W. Wegst and M. Wegst, *Stahlschlüssel*, 20. Aufl. Marbach/N: Verlag Stahlschlüssel Wegst GmbH, 2004, 719 pp., ISBN: 978-3-922599-20-3.

[126] "IS0 3685 - tool life testing with single-point turning tools," 1993.

[127] T. Segreto, D. D'Addona, and R. Teti, "Tool wear estimation in turning of inconel 718 based on wavelet sensor signal analysis and machine learning paradigms," *Production Engineering*, vol. 14, no. 5, pp. 693–705, 2020, ISBN: 0123456789, ISSN: 0944-6524. DOI: 10.1007/s11740-020-00989-2.

[128] M. Cheng, L. Jiao, X. Shi, X. Wang, P. Yan, and Y. Li, "An intelligent prediction model of the tool wear based on machine learning in turning high strength steel," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 234, no. 13, pp. 1580–1597, 2020, ISSN: 0954-4054. DOI: 10.1177/0954405420935787.

[129] P. Twardowski and M. Wiciak-Piku{\textbackslash}la, "Prediction of tool wear using artificial neural networks during turning of hardened steel," *Materials*, vol. 12, no. 19, p. 3091, 2019, ISSN: 1996-1944. DOI: 10.3390/ma12193091.

[130] P. Welch, "The use of fast fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms," *IEEE Transactions on Audio and Electroacoustics*, vol. 15, no. 2, pp. 70–73, 1967, ISSN: 0018-9278. DOI: 10.1109/TAU.1967.1161901.

[131] A. J. Bishara and J. B. Hittner, "Testing the significance of a correlation with nonnormal data: Comparison of pearson, spearman, transformation, and resampling approaches.," *Psychological Methods*, vol. 17, no. 3, pp. 399–417, 2012, ISSN: 1939-1463, 1082-989X. DOI: 10.1037/a0028087.

[132] R. R. Wilcox, "Comparing pearson correlations: Dealing with heteroscedasticity and nonnormality," *Communications in Statistics: Simulation and Computation*, vol. 38, no. 10, pp. 2220–2234, 2009, ISSN: 15324141. DOI: 10.1080/03610910903289151.

[133] B. Kilundu, P. Dehombreux, and X. Chiementin, "Tool wear monitoring by machine learning techniques and singular spectrum analysis," *Mechanical Systems and Signal Processing*, vol. 25, no. 1, pp. 400–415, 2011, ISSN: 08883270. DOI: 10.1016/j.ymssp.2010.07.014.

[134] X. Chuangwen, D. Jianming, C. Yuzhen, L. Huaiyuan, S. Zhicheng, and X. Jing, "The relationships between cutting parameters, tool wear, cutting force and vibration," *Advances in Mechanical Engineering*, vol. 10, no. 1, p. 1 687 814 017 750 434, 2018, ISSN: 1687-8132, 1687-8140. DOI: 10.1177/1687814017750434.

[135] K. S. Lee and C. S. Gan, "Material tungst : E carbi," vol. 29, no. 3, pp. 295–303, 1989.

[136] L. Colantonio, L. Equeter, P. Dehombreux, and F. Ducobu, "Explainable AI for tool condition monitoring using explainable boosting machine," *Procedia CIRP*, vol. 133, pp. 138–143, 2025, ISSN: 22128271. DOI: 10.1016/j.procir.2025.02.025.

[137] A. Khosravi, S. Nahavandi, D. Creighton, and A. F. Atiya, "Comprehensive review of neural network-based prediction intervals and new advances," *IEEE Transactions on Neural Networks*, vol. 22, no. 9, pp. 1341–1356, 2011, ISSN: 10459227. DOI: 10.1109/TNN.2011.2162110.

[138] H. Li, *Machine Learning Methods*. Singapore: Springer Nature Singapore, 2024, ISBN: 978-981-9939-16-9 978-981-9939-17-6. DOI: 10.1007/978-981-99-3917-6.

[139] G. Kim *et al.*, "Bayesian-based uncertainty-aware tool-wear prediction model in end-milling process of titanium alloy," *Applied Soft Computing*, vol. 148, p. 110 922, 2023, ISSN: 15684946. DOI: 10.1016/j.asoc.2023.110922.

[140] E. Zio, "A study of the bootstrap method for estimating the accuracy of artificial neural networks in predicting nuclear transient processes," *IEEE Transactions on Nuclear Science*, vol. 53, no. 3, pp. 1460–1478, 2006, ISSN: 00189499. DOI: 10.1109/TNS.2006.871662.

[141] M. K. Tiwari and C. Chatterjee, "Uncertainty assessment and ensemble flood forecasting using bootstrap based artificial neural networks (BANNs)," *Journal of Hydrology*, vol. 382, no. 1, pp. 20–33, 2010, Publisher: Elsevier B.V., ISSN: 00221694. DOI: 10.1016/j.jhydrol.2009.12.013.

[142] J. M. Chambers, W. S. Cleveland, B. Kleiner, and P. A. Tukey, *Graphical methods for data analysis*. 2018, 1–395, Publication Title: Graphical Methods for Data Analysis, ISBN: 978-1-351-08075-0. DOI: 10.1201/9781351072304.

[143] D. Zwillinger, Ed., *CRC standard mathematical tables and formulae*, 32. ed, Boca Raton London: CRC Press, 2012, 1 p., ISBN: 978-1-4398-3548-7.

[144] P. Linardatos, V. Papastefanopoulos, and S. Kotsiantis, "Explainable AI: A review of machine learning interpretability methods," *Entropy*, vol. 23, no. 1, p. 18, 2020, ISSN: 1099-4300. DOI: 10.3390/e23010018.

[145] Z. Chen, F. Xiao, F. Guo, and J. Yan, "Interpretable machine learning for building energy management: A state-of-the-art review," *Advances in Applied Energy*, vol. 9, p. 100 123, 2023, ISSN: 26667924. DOI: 10.1016/j.adapen.2023.100123.

[146] H. Guo, Y. Zhang, and K. Zhu, "Interpretable deep learning approach for tool wear monitoring in high-speed milling," *Computers in Industry*, vol. 138, p. 103 638, 2022, ISSN: 01663615. DOI: 10.1016/j.compind.2022.103638.

[147] A. S. Kumar, A. Agarwal, V. G. Jansari, K. A. Desai, C. Chattopadhyay, and L. Mears, "HG-XAI: Human-guided tool wear identification approach through augmentation of explainable artificial intelligence with machine vision," *Journal of Intelligent Manufacturing*, 2024, ISSN: 0956-5515, 1572-8145. DOI: 10.1007/s10845-024-02476-2.

[148] H. Nori, S. Jenkins, P. Koch, and R. Caruana, *InterpretML: A unified framework for machine learning interpretability*, 2019. arXiv: 1909.09223[cs]. [Online]. Available: http://arxiv.org/abs/1909.09223 (visited on 2024).

[149] C. Sun, M. Ma, Z. Zhao, S. Tian, R. Yan, and X. Chen, "Deep transfer learning based on sparse autoencoder for remaining useful life prediction of tool in manufacturing," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2416–2425, 2019, ISSN: 1551-3203. DOI: 10.1109/TII.2018.2881543.

[150] Y. Hao *et al.*, "Ball-end tool wear monitoring and multi-step forecasting with multi-modal information under variable cutting conditions," *Journal of Manufacturing Systems*, vol. 76, pp. 234–258, 2024, ISSN: 02786125. DOI: 10.1016/j.jmsy.2024.08.002.

[151] H. Hewamalage, C. Bergmeir, and K. Bandara, "Recurrent neural networks for time series forecasting: Current status and future directions," *International Journal of Forecasting*, vol. 37, no. 1, pp. 388–427, 2021, ISSN: 01692070. DOI: 10.1016/j.ijforecast.2020.06.008.

[152] D. M. Vock *et al.*, "Adapting machine learning techniques to censored time-to-event health record data: A general-purpose approach using inverse probability of censoring weighting," *Journal of Biomedical Informatics*, vol. 61, pp. 119–131, 2016, ISSN: 15320464. DOI: 10.1016/j.jbi.2016.03.009.

[153] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, and S. Valaee, *Recent advances in recurrent neural networks*, Version Number: 3, 2018. DOI: 10.48550/ARXIV.1801.01078.

[154] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997, ISSN: 0899-7667, 1530-888X. DOI: 10.1162/neco.1997.9.8.1735.

[155] M. Schuster and K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997, ISSN: 1053587X. DOI: 10.1109/78.650093.

[156] L. Colantonio, L. Equeter, H. Giovannelli, P. Dehombreux, S. Mahmoudi, and F. Ducobu, "Image processing with deep-learning and transfer learning for cutting tool degradation monitoring," *Procedia CIRP*, vol. 132, pp. 50–55, 2025, ISSN: 22128271. DOI: 10.1016/j.procir.2025.01.009.

[157] P. Muthuswamy and S. K, "Artificial intelligence based tool condition monitoring for digital twins and industry 4.0 applications," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 17, no. 3, pp. 1067–1087, 2023, ISSN: 1955-2513, 1955-2505. DOI: 10.1007/s12008-022-01050-5.

[158] L. Pagani, P. Parenti, S. Cataldo, P. J. Scott, and M. Annoni, "Indirect cutting tool wear classification using deep learning and chip colour analysis," *The International Journal of Advanced Manufacturing Technology*, vol. 111, no. 3, pp. 1099–1114, 2020, ISSN: 0268-3768. DOI: 10.1007/s00170-020-06055-6.

[159] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, Version Number: 6, 2014. DOI: 10.48550/ARXIV.1409.1556.

[160] X. Wu, Y. Liu, X. Zhou, and A. Mou, "Automatic identification of tool wear based on convolutional neural network in face milling process," *Sensors*, vol. 19, no. 18, p. 3817, 2019, ISSN: 1424-8220. DOI: 10.3390/s19183817.

[161] M. Iman, H. R. Arabnia, and K. Rasheed, "A review of deep transfer learning and recent advancements," *Technologies*, vol. 11, no. 2, p. 40, 2023, ISSN: 2227-7080. DOI: 10.3390/technologies11020040.

[162] F. Zhuang *et al.*, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2021, ISSN: 0018-9219, 1558-2256. DOI: 10.1109/JPROC.2020.3004555.

[163] M. Tan and Q. V. Le, "EfficientNetV2: Smaller models and faster training," 2021, Publisher: arXiv Version Number: 3. DOI: 10.48550/ARXIV.2104.00298.

[164] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 91–99, 2022, ISSN: 2666285X. DOI: 10.1016/j.gltp.2022.04.020.

[165] K. P. Murphy, *Probabilistic machine learning: an introduction* (Adaptive computation and machine learning). Cambridge, Massachusetts London, England: The MIT Press, 2022, 1 p., ISBN: 978-0-262-04682-4 978-0-262-36930-5.

[166] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," *International Journal of Computer Vision*, vol. 128, no. 2, pp. 336–359, 2020, ISSN: 0920-5691, 1573-1405. DOI: 10.1007/s11263-019-01228-7.

[167] T. Bergs, C. Holst, P. Gupta, and T. Augspurger, "Digital image processing with deep learning for automated cutting tool wear detection," *Procedia Manufacturing*, vol. 48, pp. 947–958, 2020, Publisher: Elsevier B.V., ISSN: 23519789. DOI: 10.1016/j.promfg.2020.05.134.

[168] L. Li and Q. An, "An in-depth study of tool wear monitoring technique based on image segmentation and texture analysis," *Measurement*, vol. 79, pp. 44–52, 2016, ISSN: 02632241. DOI: 10.1016/j.measurement.2015.10.029.

[169] W. Kentaro, *Labelme: Image polygonal annotation with python*.

[170] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, *Image segmentation using deep learning: A survey*, Version Number: 5, 2020. DOI: 10.48550/ARXIV.2001.05566.

[171] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds., vol. 9351, Series Title: Lecture Notes in Computer Science, Cham: Springer International Publishing, 2015, pp. 234–241, ISBN: 978-3-319-24573-7 978-3-319-24574-4. DOI: 10.1007/978-3-319-24574-4_28.

[172] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A nested u-net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, D. Stoyanov *et al.*, Eds., vol. 11045, Series Title: Lecture Notes in Computer Science, Cham: Springer International Publishing, 2018, pp. 3–11, ISBN: 978-3-030-00888-8 978-3-030-00889-5. DOI: 10.1007/978-3-030-00889-5_1.

[173] D. Jha, M. A. Riegler, D. Johansen, P. Halvorsen, and H. D. Johansen, *DoubleU-net: A deep convolutional neural network for medical image segmentation*, Version Number: 2, 2020. DOI: 10.48550/ARXIV.2006.04868.

[174] F. G. Antonaci *et al.*, "Workplace well-being in industry 5.0: A worker-centered systematic review," *Sensors*, vol. 24, no. 17, p. 5473, 2024, ISSN: 1424-8220. DOI: 10.3390/s24175473.

[175] A. Jerri, "The shannon sampling theorem—its various extensions and applications: A tutorial review," *Proceedings of the IEEE*, vol. 65, no. 11, pp. 1565–1596, 1977, ISSN: 0018-9219. DOI: 10.1109/PROC.1977.10771.

[176] I. O. for Standardization, *ISO 21920-1:2021 geometrical product specifications (GPS) — surface texture: Profile*.

[177] R. Munaro, A. Attanasio, and A. Del Prete, "Tool wear monitoring with artificial intelligence methods: A review," *Journal of Manufacturing and Materials Processing*, vol. 7, no. 4, p. 129, 2023, ISSN: 2504-4494. DOI: 10.3390/jmmp7040129.

[178] C. Nath, "Integrated tool condition monitoring systems and their applications: A comprehensive review," *Procedia Manufacturing*, vol. 48, pp. 852–863, 2020, ISSN: 23519789. DOI: 10.1016/j.promfg.2020.05.123.

[179] Z. Li, R. Liu, and D. Wu, "Data-driven smart manufacturing: Tool wear monitoring with audio signals and machine learning," *Journal of Manufacturing Processes*, vol. 48, pp. 66–76, 2019, ISSN: 15266125. DOI: 10.1016/j.jmapro.2019.10.020.

[180] G. Wei, J. Zhao, Y. Feng, A. He, and J. Yu, "A novel hybrid feature selection method based on dynamic feature importance," *Applied Soft Computing*, vol. 93, p. 106 337, 2020, ISSN: 15684946. DOI: 10.1016/j.asoc.2020.106337.

[181] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, *Optuna: A next-generation hyperparameter optimization framework*, Version Number: 1, 2019. DOI: 10.48550/ARXIV.1907.10902.

[182] D. F. Specht, "Probabilistic neural networks," *Neural Networks*, vol. 3, no. 1, pp. 109–118, 1990, ISSN: 08936080. DOI: 10.1016/0893-6080(90)90049-Q.

[183] C. M. Bishop, *Neural networks for pattern recognition*, Reprinted. Oxford: Oxford University Press, 2010, 482 pp., ISBN: 978-0-19-853864-6.

[184] R. Fu, Q. Hu, X. Dong, Y. Guo, Y. Gao, and B. Li, *Axiom-based grad-CAM: Towards accurate visualization and explanation of CNNs*, Version Number: 4, 2020. DOI: 10.48550/ARXIV.2008.02312.

[185] K. Shantisagar, R. Jegadeeshwaran, G. Sakthivel, and T. M. Alamelu Manghai, "Vibration Based Tool Insert Health Monitoring Using Decision Tree and Fuzzy Logic," *Structural Durability & Health Monitoring*, vol. 13, no. 3, pp. 303–316, 2019, ISSN: 1930-2991. DOI: 10.32604/sdhm.2019.00355.

[186] Q. Ren, L. Baron, M. Balazinski, R. Botez, and P. Bigras, "Tool wear assessment based on type-2 fuzzy uncertainty estimation on acoustic emission," *Applied Soft Computing*, vol. 31, pp. 14–24, 2015, ISSN: 15684946. DOI: 10.1016/j.asoc.2015.02.037.

[187] R. Kuo, "Multi-sensor integration for on-line tool wear estimation through artificial neural networks and fuzzy neural network," *Engineering Applications of Artificial Intelligence*, vol. 13, no. 3, pp. 249–261, 2000, ISSN: 09521976. DOI: 10.1016/S0952-1976(00)00008-7.

[188] Q. Ren, M. Balazinski, and L. Baron, "Uncertainty prediction for tool wear condition using type-2 tsk fuzzy approach," in *2009 IEEE International Conference on Systems, Man and Cybernetics*, 2009, pp. 660–665, ISBN: 978-1-4244-2793-2. DOI: 10.1109/ICSMC.2009.5346690.

[189] Y.-M. Chiang, N.-Z. Hsu, and K.-L. Lin, "Driver Assistance System Based on Monocular Vision," in *New Frontiers in Applied Artificial Intelligence*, N. T. Nguyen, L. Borzemski, A. Grzech, and M. Ali, Eds., vol. 5027 LNAI, Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 1–10, ISBN: 354069045X. DOI: 10.1007/978-3-540-69052-8_1.

[190] S. Achiche, M. Balazinski, L. Baron, and K. Jemielniak, "Tool wear monitoring using genetically-generated fuzzy knowledge bases," *Engineering Applications of Artificial Intelligence*, vol. 15, no. 3-4, pp. 303–314, 2002, ISSN: 09521976. DOI: 10.1016/S0952-1976(02)00071-4.

[191] M. Marani, M. Zeinali, J. Kouam, V. Songmene, and C. K. Mechefske, "Prediction of cutting tool wear during a turning process using artificial intelligence techniques," *The International Journal of Advanced Manufacturing Technology*, vol. 111, no. 1-2, pp. 505–515, 2020, ISSN: 0268-3768. DOI: 10.1007/s00170-020-06144-6.

[192] B. Zhang, C. Katinas, and Y. C. Shin, "Robust Tool Wear Monitoring Using Systematic Feature Selection in Turning Processes With Consideration of Uncertainties," *Journal of Manufacturing Science and Engineering*, vol. 140, no. 8, 2018, ISSN: 1087-1357. DOI: 10.1115/1.4040267.

[193] Q. Ren, S. Achiche, K. Jemielniak, and P. Bigras, "An enhanced adaptive neural fuzzy tool condition monitoring for turning process," in *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2016, pp. 1976–1982, ISBN: 978-1-5090-0626-7. DOI: 10.1109/FUZZ-IEEE.2016.7737934.

[194] M. Rizal, J. A. Ghani, M. Z. Nuawi, and C. H. C. Haron, "Online tool wear prediction system in the turning process using an adaptive neuro-fuzzy inference system," *Applied Soft Computing*, vol. 13, no. 4, pp. 1960–1968, 2013, ISSN: 15684946. DOI: 10.1016/j.asoc.2012.11.043.

[195] A. Gajate, R. Haber, R. del Toro, P. Vega, and A. Bustillo, "Tool wear monitoring using neuro-fuzzy techniques: a comparative study in a turning process," *Journal of Intelligent Manufacturing*, vol. 23, no. 3, pp. 869–882, 2012, ISSN: 0956-5515. DOI: 10.1007/s10845-010-0443-y.

[196] S. S. Gill, R. Singh, J. Singh, and H. Singh, "Adaptive neuro-fuzzy inference system modeling of cryogenically treated AISI M2 HSS turning tool for estimation of flank wear," *Expert Systems with Applications*, vol. 39, no. 4, pp. 4171–4180, 2012, ISSN: 09574174. DOI: 10.1016/j.eswa.2011.09.117.

[197] A. Basari, A. Jaya, M. Muhamad, M. Rahman, S. Hashim, and H. Haron, "Application of ANFIS in Predicting TiAlN Coatings Flank Wear," in *2011 Third International Conference on Computational Intelligence, Modelling & Simulation*, 2011, pp. 57–62, ISBN: 978-1-4577-1797-0. DOI: 10.1109/CIMSim.2011.20.

[198] Q. Ren, M. Balazinski, L. Baron, and K. Jemielniak, "TSK fuzzy modeling for tool wear condition in turning processes: An experimental study," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 2, pp. 260–265, 2011, ISSN: 09521976. DOI: 10.1016/j.engappai.2010.10.016.

[199] A. Gajate, R. E. Haber, J. R. Alique, and P. I. Vega, "Transductive-Weighted Neuro-Fuzzy Inference System for Tool Wear Prediction in a Turning Process," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, E. Corchado, X. Wu, E. Oja, Á. Herrero, and B. Baruque, Eds., vol. LNAI 5572, Berlin Heidelberg: Springer, 2009, pp. 113–120, ISBN: 3540878262. DOI: 10.1007/978-3-642-02319-4_14.

[200] F. Penedo, R. E. Haber, A. Gajate, and R. M. del Toro, "Hybrid Incremental Modeling Based on Least Squares and Fuzzy K-NN for Monitoring Tool Wear in Turning Processes," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 4, pp. 811–818, 2012, ISSN: 1551-3203. DOI: 10.1109/TII.2012.2205699.

[201] O. Massol, X. Li, R. Gouriveau, J. H. Zhou, and O. P. Gan, "An exTS based neuro-fuzzy algorithm for prognostics and tool condition monitoring," in *2010 11th International Conference on Control Automation Robotics & Vision*, 2010, pp. 1329–1334, ISBN: 978-1-4244-7814-9. DOI: 10.1109/ICARCV.2010.5707842.

[202] H. Gao, M. Xu, X. Shi, and H. Huang, "Tool Wear Monitoring Based on Localized Fuzzy Neural Networks for Turning Operation," in *2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, vol. 4, 2009, pp. 417–420, ISBN: 978-0-7695-3735-1. DOI: 10.1109/FSKD.2009.754.

[203] S.-P. Lo, "The Application of an ANFIS and Grey System Method in Turning Tool-Failure Detection," *The International Journal of Advanced Manufacturing Technology*, vol. 19, no. 8, pp. 564–572, 2002, ISSN: 0268-3768. DOI: 10.1007/s001700200061.

[204] V. S. Sharma, S. K. Sharma, and A. K. Sharma, "Cutting tool wear estimation for turning," *Journal of Intelligent Manufacturing*, vol. 19, no. 1, pp. 99–108, 2008, ISSN: 0956-5515. DOI: 10.1007/s10845-007-0048-2.

[205] T.-I. Liu, S.-D. Song, G. Liu, and Z. Wu, "Online monitoring and measurements of tool wear for precision turning of stainless steel parts," *The International Journal of Advanced Manufacturing Technology*, vol. 65, no. 9-12, pp. 1397–1407, 2013, ISSN: 0268-3768. DOI: 10.1007/s00170-012-4265-2.

[206] M. L. Huang, J. S. Yang, J. Y. Wu, and S. H. Chang, "Wear value prediction of CNC turning tools based on ν-GSVR with a new hybrid evolutionary algorithm," *Journal of Applied Science and Engineering*, vol. 23, no. 2, pp. 369–378, 2020, ISSN: 15606686. DOI: 10.6180/jase.202006_23(2).0021.

[207] D. Kong, Y. Chen, and N. Li, "Monitoring tool wear using wavelet package decomposition and a novel gravitational search algorithm–least square support vector machine model," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 234, no. 3, pp. 822–836, 2020, ISSN: 0954-4062. DOI: 10.1177/0954406219887318.

[208] K. Vikash Kumar and R. Sivasankara Raju, "Statistical Modeling and Optimization of Al-MMCs Reinforced with Coconut Shell Ash Particulates," in *Lecture Notes in Mechanical Engineering*, B. Deepak, D. Parhi, and P. C. Jena, Eds., Singapore: Springer Nature Singapore, 2020, pp. 703–712, ISBN: 9789811526954. DOI: 10.1007/978-981-15-2696-1_67.

[209] D. Kong, Y. Chen, N. Li, and S. Tan, "Tool wear monitoring based on kernel principal component analysis and v-support vector regression," *The International Journal of Advanced Manufacturing Technology*, vol. 89, no. 1-4, pp. 175–190, 2017, ISSN: 0268-3768. DOI: 10.1007/s00170-016-9070-x.

[210] D. Salgado and F. Alonso, "An approach based on current and sound signals for in-process tool wear monitoring," *International Journal of Machine Tools and Manufacture*, vol. 47, no. 14, pp. 2140–2152, 2007, ISSN: 08906955. DOI: 10.1016/j.ijmachtools.2007.04.013.

[211] N. Li, Y. Chen, D. Kong, and S. Tan, "Force-based tool condition monitoring for turning process using v-support vector regression," *The International Journal of Advanced Manufacturing Technology*, vol. 91, no. 1-4, pp. 351–361, 2017, ISSN: 0268-3768. DOI: 10.1007/s00170-016-9735-5.

[212] L. C. Brito, M. B. da Silva, and M. A. V. Duarte, "Identification of cutting tool wear condition in turning using self-organizing map trained with imbalanced data," *Journal of Intelligent Manufacturing*, vol. 32, no. 1, pp. 127–140, 2021, ISSN: 0956-5515. DOI: 10.1007/s10845-020-01564-3.

[213] R. G. Silva and S. J. Wilcox, "Feature evaluation and selection for condition monitoring using a self-organizing map and spatial statistics," *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 33, no. 1, pp. 1–10, 2019, ISSN: 0890-0604. DOI: 10.1017/S0890060417000518.

[214] J. Kang and L. N. Guan, "Tool Wear Intelligence Measure in Cutting Process Based on HMM," *Applied Mechanics and Materials*, vol. 52-54, pp. 482–487, 2011, ISSN: 1662-7482. DOI: 10.4028/www.scientific.net/AMM.52-54.482.

[215] L. Wang, M. G. Mehrabi, and E. Kannatey-Asibu, "Hidden Markov Model-based Tool Wear Monitoring in Turning," *Journal of Manufacturing Science and Engineering*, vol. 124, no. 3, pp. 651–658, 2002, ISSN: 1087-1357. DOI: 10.1115/1.1475320.

[216] L. Kang, S. Wang, S. Wang, C. Ma, L. Yi, and H. Zou, "Tool wear monitoring using generalized regression neural network," *Advances in Mechanical Engineering*, vol. 11, no. 5, p. 168 781 401 984 917, 2019, ISSN: 1687-8140. DOI: 10.1177/1687814019849172.

[217] N. Mandal, B. Mondal, and B. Doloi, "Application of Back Propagation Neural Network Model for Predicting Flank Wear of Yttria Based Zirconia Toughened Alumina (ZTA) Ceramic Inserts," *Transactions of the Indian Institute of Metals*, vol. 68, no. 5, pp. 783–789, 2015, ISSN: 0972-2815. DOI: 10.1007/s12666-015-0511-2.

[218] D. K. Sarma and U. S. Dixit, "Neural network modelling of forces and indirect prediction of tool wear in turning of grey cast iron with ceramic tool," *International Journal of Machining and Machinability of Materials*, vol. 8, no. 1-2, pp. 55–75, 2010, ISSN: 1748572X. DOI: 10.1504/IJMMM.2010.03448.

[219] I. Deiab, K. Assaleh, and F. Hammad, "Application of sensor fusion and polynomial classifiers to tool wear monitoring," in *2008 5th International Symposium on Mechatronics and Its Applications*, 2008, pp. 1–6, ISBN: 978-1-4244-2033-9. DOI: 10.1109/ISMA.2008.4648808.

[220] X. Wang, W. Wang, Y. Huang, N. Nguyen, and K. Krishnakumar, "Design of neural network-based estimator for tool wear modeling in hard turning," *Journal of Intelligent Manufacturing*, vol. 19, no. 4, pp. 383–396, 2008, ISSN: 0956-5515. DOI: 10.1007/s10845-008-0090-8.

[221] T. Özel and A. Nadgir, "Prediction of flank wear by using back propagation neural network modeling when cutting hardened H-13 steel with chamfered and honed CBN tools," *International Journal of Machine Tools and Manufacture*, vol. 42, no. 2, pp. 287–297, 2002, ISSN: 08906955. DOI: 10.1016/S0890-6955(01)00103-1.

[222] Y. Shen *et al.*, "Predicting tool wear size across multi-cutting conditions using advanced machine learning techniques," *Journal of Intelligent Manufacturing*, 2020, ISSN: 0956-5515. DOI: 10.1007/s10845-020-01625-7.

[223] B. Neef, J. Bartels, and S. Thiede, "Tool Wear and Surface Quality Monitoring Using High Frequency CNC Machine Tool Current Signature," in *2018 IEEE 16th International Conference on Industrial Informatics (INDIN)*, 2018, pp. 1045–1050, ISBN: 978-1-5386-4829-2. DOI: 10.1109/INDIN.2018.8472037.

[224] S. Yang, S. Wang, L. Yi, H. Xue, Y. Cao, and S. Sun, "A novel monitoring method for turning tool wear based on support vector machines," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 230, no. 8, pp. 1359–1371, 2016, ISSN: 0954-4054. DOI: 10.1177/0954405416645998.

[225] K.-f. Zhang, H.-q. Yuan, and P. Nie, "A method for tool condition monitoring based on sensor fusion," *Journal of Intelligent Manufacturing*, vol. 26, no. 5, pp. 1011–1026, 2015, ISSN: 0956-5515. DOI: 10.1007/s10845-015-1112-y.

[226] N. N. Bhat, S. Dutta, T. Vashisth, S. Pal, S. K. Pal, and R. Sen, "Tool condition monitoring by SVM classification of machined surface images in turning," *The International Journal of Advanced Manufacturing Technology*, vol. 83, no. 9-12, pp. 1487–1502, 2016, ISSN: 0268-3768. DOI: 10.1007/s00170-015-7441-3.

[227] S. Dutta, S. K. Pal, and R. Sen, "On-machine tool prediction of flank wear from machined surface images using texture analyses and support vector regression," *Precision Engineering*, vol. 43, pp. 34–42, 2016, ISSN: 01416359. DOI: 10.1016/j.precisioneng.2015.06.007.

[228] R. Tao, Y. Zhang, L. Wang, and X. Zhao, "Research of Tool State Recognition Based on CEEMD-WPT," in ser. Lecture Notes in Computer Science, X. Sun, Z. Pan, and E. Bertino, Eds., vol. 11064, Cham: Springer International Publishing, 2018, pp. 59–70, ISBN: 978-3-030-00008-0. DOI: 10.1007/978-3-030-00009-7_6.

[229] J. Sun, G. S. Hong, M. Rahman, and Y. S. Wong, "Identification of feature set for effective tool condition monitoring by acoustic emission sensing," *International Journal of Production Research*, vol. 42, no. 5, pp. 901–918, 2004, ISSN: 0020-7543. DOI: 10.1080/00207540310001626652.

[230] Yiqiu Qian, Jia Tian, Libing Liu, Yu Zhang, and Yingshu Chen, "A tool wear predictive model based on SVM," in *2010 Chinese Control and Decision Conference*, IEEE, 2010, pp. 1213–1217, ISBN: 978-1-4244-5181-4. DOI: 10.1109/CCDC.2010.5498161.

[231] R. G. Silva and R. Reis, "Adaptive self-organizing map applied to lathe tool condition monitoring," in *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2017, pp. 1–6, ISBN: 978-1-5090-6505-9. DOI: 10.1109/ETFA.2017.8247641.

[232] R. G. Silva, "Condition monitoring of the cutting process using a self-organizing spiking neural network map," *Journal of Intelligent Manufacturing*, vol. 21, no. 6, pp. 823–829, 2010, ISSN: 0956-5515. DOI: 10.1007/s10845-009-0258-x.

[233] N. K. A. Al-Sahib and A. M. Bachaa, "Tool Wear Monitoring in Turning Operation Using Vibration and Strain Measurement With Neural Network," in *Manufacturing Engineering and Materials Handling, Parts A and B*, 2005, pp. 783–792, ISBN: 0-7918-4223-1. DOI: 10.1115/IMECE2005-80699.

[234] A. Elsheikh, S. Yacout, M.-S. Ouali, and Y. Shaban, "Failure time prediction using adaptive logical analysis of survival curves and multiple machining signals," *Journal of Intelligent Manufacturing*, vol. 31, no. 2, pp. 403–415, 2020, ISSN: 0956-5515. DOI: 10.1007/s10845-018-1453-4.

[235] J. Saran, M. Elangovan, and V. Sugumaran, "Tool wear monitoring using NNGE algorithm," *International Journal of Mechanical and Production Engineering Research and Development*, vol. 9, no. 5, pp. 53–62, 2019, ISSN: 2249–6890. DOI: 10.24247/ijmperdoct20196.

[236] M. Murua, A. Suárez, L. N. L. de Lacalle, R. Santana, and A. Wretland, "Feature extraction-based prediction of tool wear of Inconel 718 in face turning," *Insight - Non-Destructive Testing and Condition Monitoring*, vol. 60, no. 8, pp. 443–450, 2018, ISSN: 1354-2575. DOI: 10.1784/insi.2018.60.8.443.

[237] H.-C. Yang, Y.-Y. Li, M.-H. Hung, and F.-T. Cheng, "A cyber-physical scheme for predicting tool wear based on a hybrid dynamic neural network," *Journal of the Chinese Institute of Engineers*, vol. 40, no. 7, pp. 614–625, 2017, ISSN: 0253-3839. DOI: 10.1080/02533839.2017.1372223.

[238] Y. Shaban, S. Yacout, M. Balazinski, and K. Jemielniak, "Cutting tool wear detection using multiclass logical analysis of data," *Machining Science and Technology*, vol. 21, no. 4, pp. 526–541, 2017, ISSN: 1091-0344. DOI: 10.1080/10910344.2017.1336177.

[239] A. F. A. Azmi, A. A. Habaibeh, and J. Redgate, "Rapid design of tool-wear condition monitoring systems for turning processes using novelty detection," *International Journal of Manufacturing Technology and Management*, vol. 17, no. 3, p. 232, 2009, ISSN: 1368-2148. DOI: 10.1504/IJMTM.2009.023931.

[240] X. Wang, Y. Huang, N. Nguyen, and K. Krishnakumar, "CBN tool flank wear modelling using Hybrid Neural Network," *International Journal of Mechatronics and Manufacturing Systems*, vol. 1, no. 1, p. 83, 2008, ISSN: 1753-1039. DOI: 10.1504/IJMMS.2008.018280.

[241] B. Sick, "Fusion of hard and soft computing techniques in indirect, online tool wear monitoring," *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 32, no. 2, pp. 80–91, 2002, ISSN: 1094-6977. DOI: 10.1109/TSMCC.2002.801347.

[242] S. S. Aralikatti, K. N. Ravikumar, H. Kumar, H. Shivananda Nayaka, and V. Sugumaran, "Comparative Study on Tool Fault Diagnosis Methods Using Vibration Signals and Cutting Force Signals by Machine Learning Technique," *Structural Durability & Health Monitoring*, vol. 14, no. 2, pp. 127–145, 2020, ISSN: 1930-2991. DOI: 10.32604/sdhm.2020.07595.

[243] N. K. Vuong, Y. Xue, S. Liu, Y. Zhou, and M. Wu, "Predicting Surface Roughness and Flank Wear in Turning Processes," in *2020 IEEE International Conference on Prognostics and Health Management (ICPHM)*, vol. 2020-June, 2020, pp. 1–6, ISBN: 978-1-7281-6286-7. DOI: 10.1109/ICPHM49022.2020.9187056.

[244] B. H. Aghdam, E. Cigeroglu, and M. H. Sadeghi, "Output Only Functional Series Time Dependent AutoRegressive Moving Average (FS-TARMA) Modelling of Tool Acceleration Signals for Wear Estimation," in *Conference Proceedings of the Society for Experimental Mechanics Series*, ser. Conference Proceedings of the Society for Experimental Mechanics Series, C. Niezrecki, Ed., vol. 7, Springer International Publishing, 2015, pp. 111–122, ISBN: 978-3-319-15229-5. DOI: 10.1007/978-3-319-15230-1_11.

[245] M. Pratama, E. Dimla, C. Y. Lai, and E. Lughofer, "Metacognitive learning approach for online tool condition monitoring," *Journal of Intelligent Manufacturing*, vol. 30, no. 4, pp. 1717–1737, 2019, ISSN: 0956-5515. DOI: 10.1007/s10845-017-1348-9.

[246] K. Jemielniak, T. Urbański, J. Kossakowska, and S. Bombiński, "Tool condition monitoring based on numerous signal features," *The International Journal of Advanced Manufacturing Technology*, vol. 59, no. 1-4, pp. 73–81, 2012, ISSN: 0268-3768. DOI: 10.1007/s00170-011-3504-2.

[247] H. Gao, H. Gao, C. Chen, Y. Su, and M. Xu, "The Investigation of A Self-adjusting Tool Wear Monitoring System," in *2006 International Conference on Mechatronics and Automation*, 2006, pp. 1690–1694, ISBN: 1-4244-0465-7. DOI: 10.1109/ICMA.2006.257451.

Appendix A

Scientific contributions

This appendix list of publications made during the thesis in journal, conference or poster presentation. They are sorted by type and year.

A.1 Journal Papers

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (10 December 2021). A Systematic Literature Review of Cutting Tool Wear Monitoring in Turning by Using Artificial Intelligence Techniques. "Machines, 9" (351), 1-54. doi:10.3390/machines9120351

Colantonio, L., Dehombreux, P., Hajzman, M., & Verlinden, O. (2022). 3D projection of the LuGre friction model adapted to varying normal forces. "Multibody System Dynamics 55(3)", 267-291.

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (26 May 2024). Confidence Interval Estimation for Cutting Tool Wear Prediction in Turning Using Bootstrap-Based Artificial Neural Networks. "Sensors, 24" (11), 3432. doi:10.3390/s24113432

Equeter, L., Do, P., Colantonio, L., Tiberi, L., Dehombreux, P., & Iung, B. (14 October 2025). Hypotheses in Opportunistic Maintenance Modeling: A Critical and Systematic Literature Review. "Machines, 13" (10), 947. doi:10.3390/machines13100947

Colantonio, L., Moroncini, G., Sénéchal, O., Dehombreux, P., Ducobu, F., & Equeter, L., Cost and Environmental Assessment of Tool Replacement Strategies Under Imperfect Wear Monitoring in Ti6Al4V Milling. "International Journal of Prognostics and Health Management, 16" (4). doi: 10.36001/ijphm.2025.v16i4.4269

A.2 Conference Papers

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (19 April 2023). Comparison of cutting tool wear classification performance with artificial intelligence techniques. "Materials Research Proceedings, 28", 1265-1274. doi: 10.21741/9781644902479-137

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (2024). Comparison of artificial intelligence techniques for cutting tool condition monitoring. "Materials Research Proceedings". doi:10.21741/9781644903131-217

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (03 April 2025). Explainable AI for tool condition monitoring using Explainable Boosting Machine. "Procedia CIRP, 133", 138-143. doi:10.1016/j.procir.2025.02.025

Colantonio, L., Equeter, L., Giovannelli, H., Dehombreux, P., Mahmoudi, S., & Ducobu, F. (2025). Image processing with deep-learning and transfer learning for cutting tool degradation monitoring. "Procedia CIRP, 132", 50-55. doi: 10.1016/j.procir.2025.01.009

A.3 Oral Presentations with Abstract Published or Not

Colantonio, L., Dehombreux, P., Hajzman, M., & Verlinden, O. (2021). "3D LuGre Model for Multibody Systems Involving Contacts with Stick-slip". Paper presented at ECCOMAS Thematic Conference on Multibody Dynamics, Budapest, Hungary.

Colantonio, L. (26 January 2021). "Prédiction de la Dégradation des outils de coupe par des méthodes d'intelligence artificielle". Paper presented at 29èmes Journées Manufacturing'21, Online.

Colantonio, L. (13 May 2022). "Prédiction de la dégradation des outils de coupe avec des méthodes d'intelligence artificielle". Paper presented at 30 èmes journée Manufacturing'21, Besançon, France.

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (24 November 2022). "Cutting Tool Health Monitoring in Turning with Artificial Intelligence Technique". Paper presented at 6th International Conference on System Reliability and Safety, Venise, Italy.

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (October 2023). "Suivi de la Dégradation des Outils de Coupe avec des Méthodes d'Intelligence Artificielle". Paper presented at 31èmes Journées Manufacturing'21, Cluny, France.

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (19 April 2023). "Comparison of cutting tool wear classification performance with artificial intelligence techniques". Paper presented at 26th International ESAFORM Conference on

Material Forming, ESAFORM 2023.

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (03 July 2023). "Cutting Tool Health Monitoring in Turning with AI". Paper presented at 12th IMA International Conference on Modelling in Industrial Maintenance and Reliability (MIMAR), Nottingham, United Kingdom.

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (2023). Cutting Tool Degradation Monitoring in Turning with Artificial Neural Network. In "Proceedings of the 33rd European Safety and Reliability Conference (ESREL 2023)". Singapore, Singapore: Research Publishing, Singapore. doi:10.3850/978-981-18-8071-1_P287-cd

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (25 April 2024). "Comparison of artificial intelligence techniques for cutting tool condition monitoring" [Paper presentation]. 27th International ESAFORM Conference on Material Forming, ESAFORM 2024, Toulouse, France.

Colantonio, L., Equeter, L., Giovannelli, H., Mahmoudi, S., Dehombreux, P., & Ducobu, F. (2024). "Image Processing With Deep-Learning And Transfer Learning For Cutting Tool Degradation Monitoring" [Paper presentation]. 12th CIRP Global Web Conference (CIRPe 2024).

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (2024). Industrial Tool Condition Monitoring: Integrated Direct And Indirect Approach For Automated Database Creation. In Krzysztof Kolowrocki & Ewa Dabrowska (Eds.), "Advances in Reliability, Safety and Security - ESREL 2024 Collection of Extended Abstracts Part 1". Gdynia, Poland: Polish Safety and Reliability Association.

Colantonio, L., Moroncini, G., Sénéchal, O., Dehombreux, P., Ducobu, F., & Equeter, L. (09 July 2025). "Joint environmental and economic assessment of cutting tool replacement policies in Ti6Al4V milling under uncertain condition estimate" [Paper presentation]. 13th IMA International Conference on Modelling in Industrial Maintenance and Reliability MIMAR2025, Nancy, France.

A.4 Posters

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (06 September 2022). "Predicting the Health of Cutting Tools with Artificial Intelligence". Poster session presented at Mardi des Chercheurs 2022 (MdC2022).

Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (26 March 2024). "Predicting the Health of Cutting Tool with Artificial Intelligence". Poster presentation. Mardi des chercheurs 2024 (MdC 2024).

Appendix B

Surface Roughness of the Cutting Tools

This chapter presents the results of the roughness measurements performed on the cutting tools used in the experimental framework described in Chapter 13. These measurements were carried out at Mondragon Unibertsitatea by Mikel Berrioategortua, Gorka Ortiz de Zárate, and Pedro Arrazola.

B.1 Introduction

During the milling tests described in Chapter 13, variations in tool wear were observed on cutting tools operating under identical cutting conditions. It is therefore important to characterize the possible geometric variations of the cutting tools that could explain these differences in wear. In Chapter 13, this analysis was carried out for the cutting-edge radius. Another parameter that must be monitored is the surface roughness of the cutting tool. This appendix therefore presents the roughness measurements performed on the cutting tools. Appendix B.2 describes the roughness on the side cutting edge, whereas Appendix B.3 focuses on the roughness of the face cutting edge, as defined in Figure B.1.

B.2 Roughness in Side Cutting Edge

Figure B.2 presents the arithmetic roughness of the side cutting edge for two new cutting tools compared with the worn tool 0021. The error bars represent the measurement variability between the three teeth of the tool. This figure shows that the arithmetic roughness of the clearance face and the rake face differs significantly between a new tool and a worn tool. The roughness magnitude of the clearance face ranges between 0.2 µm and 0.3 µm, whereas that of the rake face is around 0.85 µm.

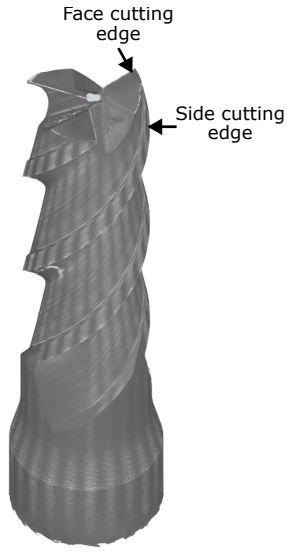


Figure B.1: Definition of the face and side cutting edges.

New tools exhibit a lower clearance-face roughness compared with the worn tool, indicating that this surface tends to increase in roughness as wear progresses.

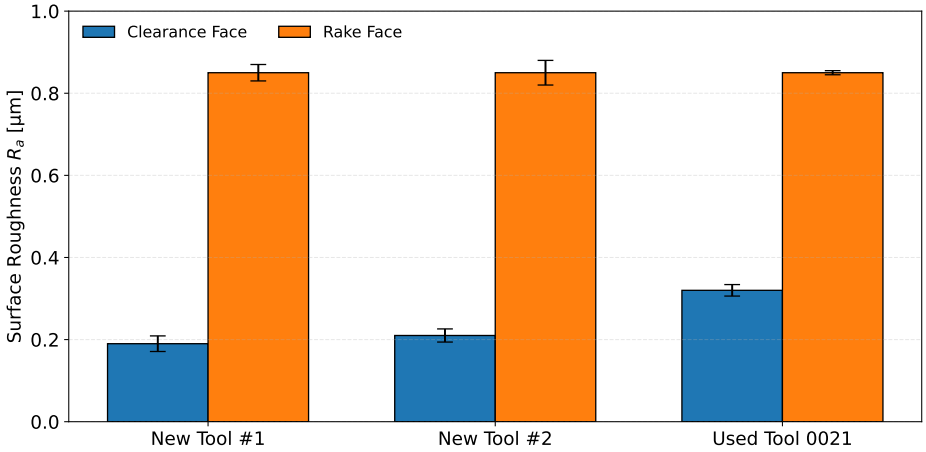


Figure B.2: Arithmetic roughness (R_a) for the side cutting edges of the cutting tools.

B.3 Roughness in Face Cutting Edge

Figure B.3 presents the same type of plot as that shown for the side cutting edge, but this time for the face cutting edge. As observed on the side cutting edge, the clearance face exhibits a lower roughness than the rake face, with values around 0.15 µm and 0.7 µm respectively. No variation between the new and the worn tool

is observed, indicating that on this face the roughness does not evolve with tool wear. It should be noted, however, that this face experienced very limited wear during the tests, which explains these results. If this face had been subjected to more significant wear, a difference in roughness might have been observed.

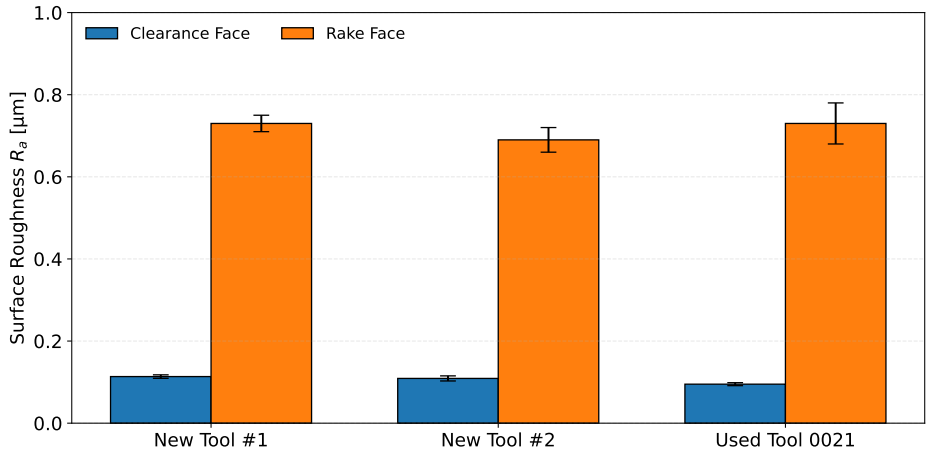


Figure B.3: Arithmetic roughness (R_a) for the face cutting edges of the cutting tools.

B.4 Conclusions

The roughness measurements performed on both the side and face cutting edges show that the clearance face consistently exhibits a lower roughness than the rake face, regardless of the tool's wear state. On the side cutting edge, the clearance-face roughness increases with tool wear, indicating that this surface is sensitive to wear progression. In contrast, no roughness evolution is observed on the face cutting edge, which is consistent with the very limited wear observed on this surface during the tests. Overall, these results suggest that surface roughness evolve with wear on the side cutting edge, but not on the face cutting edge under the experimental conditions described in Chapter 13.

Appendix C

Continued State of the Art

C.1 Continued In-depth analysis

Chapter 4 focused its in-depth analysis on Artificial Neural Network (ANN) approaches, as this technique is the most extensively utilized throughout the thesis. The remainder of this chapter presents the other AI approaches that were not previously covered. Sec C.1.1 to C.1.6 presents the Fuzzy, SVM, SOM, HMM, CNN, and other approaches. Sec. C.2 presents the results of the articles analysed in the SLR in table form.

C.1.1 Fuzzy Inference System and ANFIS

This section presents the results for the ANFIS and FIS approaches. It was chosen to group these two approaches because the observations made in the following are identical for both. The 26 articles are presented in Table C.2 for the FIS and Table C.3 for the neuro-fuzzy approach.

C.1.1.1 Presentation of the Articles

There are 8 articles about FIS: [95, 107, 185–190] and 18 articles for the neuro-fuzzy approach: [86, 91, 95, 191–205]. For the inference system, it is important to note that, out of the 8 articles, five articles have the author Balazinski Marek (and Baron Luc for four of these articles) in the author list which explains the similarity in the approaches presented in these articles ([95, 186, 188–190]).

C.1.1.2 Input

The inputs used for the fuzzy models are presented in Figure C.1. It is observed that the most used input is the cutting forces followed by the AE and feed rate signals. With 17 out of 26 articles using it, the cutting force is the most used input

signal. A particular approach using temperature as well as the power and voltage of the machine as input signal is presented in [197].

C.1.1.3 Pre-Processing

The pre-processing technique use is presented in Figure C.2. The most used pre-processing techniques are: None and Statistic. It indicates that the fuzzy logic approaches do not require elaborated pre-processing technique to obtain a good accuracy.

C.1.1.4 Output

The output of the fuzzy models is presented in Figure C.3. The vast majority of articles use the fuzzy logic to monitor the flank wear V_b.

C.1.1.5 Comparison

For the FIS, the papers [188, 189] perform a comparison of their approach that uses a Tagaki–Sugeno–Kang Fuzzy Approach Based on Subtractive Clustering Method with a neural network, neuro-fuzzy and Mamdani fuzzy logic. The approaches proposed in these papers have a lower root mean squared error on the prediction of V_b.

C.1.1.6 Conclusions

The fuzzy approaches occupied a large amount of the literature around the 2010s; today, they are much less represented in the literature. There is nothing that explain this decline in interest. It seems that the interest by the researcher has evolved such that these approaches have been replaced. The interest of fuzzy approaches lies mainly in their explicable aspect. Unlike neural networks, a fuzzy approach is not a black box. This is a great feature, especially in shop floor applications where the fuzzy approach can be of interest by providing more information than a simple prediction. In [91], the low processor time is also cited as a key feature of the fuzzy approach compared with neural networks.

C.1.2 SVM

As mentioned in Section 4.3.2, SVM received great interest over the past five years and is one of the most used techniques to monitor the tool wear with 18 articles. A presentation of these articles is provided in Table C.4.

C.1.2.1 Input

The input features used for the SVM approach are presented in Figure C.4. It is observed that the most used input is the cutting force. It also appears that the use of images to generate features is more commonly used in this AI technique than with other approaches (except CNN).

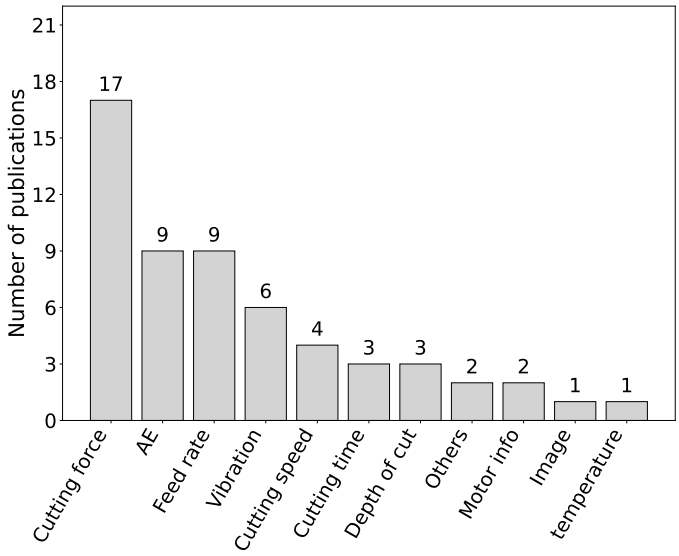


Figure C.1: Repartition of inputs for fuzzy techniques.

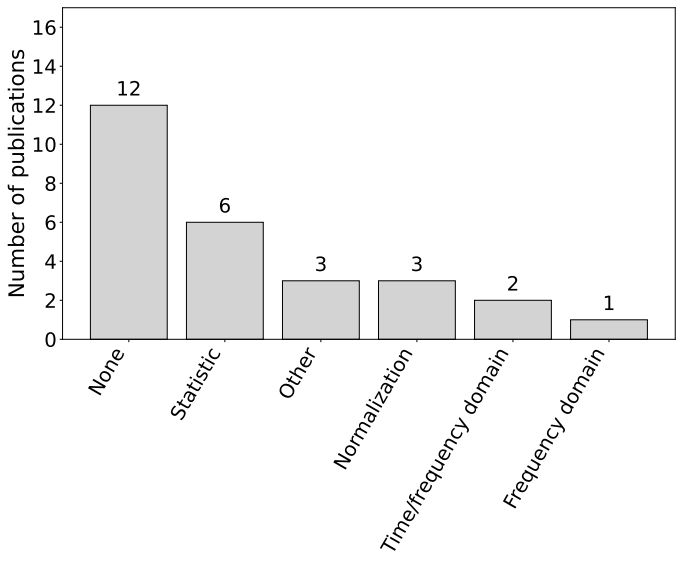


Figure C.2: Repartition of pre-processing for fuzzy techniques.

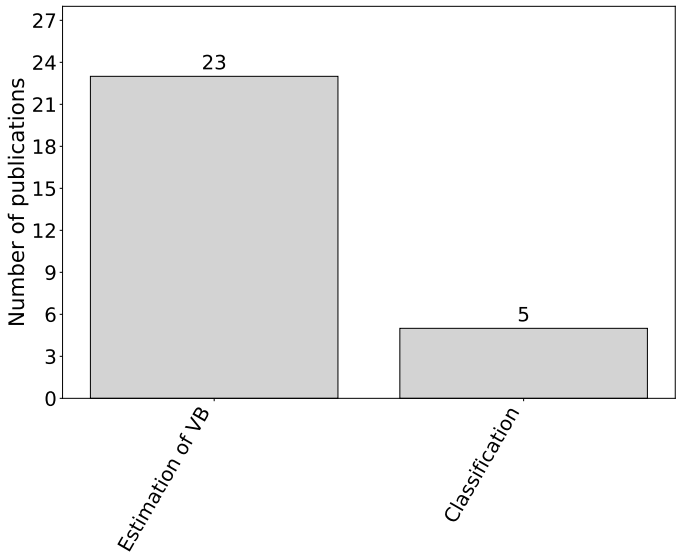


Figure C.3: Repartition of outputs for fuzzy techniques.

C.1.2.2 Pre-Processing

Due to the sensibility of SVM to noise and outliers, it appears that this approach requires more pre-processing on the data than ANN. Indeed as Figure C.5 shows, the vast majority of the approaches show the use of a pre-processing technique. This pre-processing is performed in the time-frequency domain and some statistical features are computed.

C.1.2.3 Output

The output of the SVM is presented in Figure C.6. As described, SVM was initially only used in classification purpose. Here, the number of articles that use SVM for regression and classification are almost the same.

C.1.2.4 Comparison and Particular Approaches

In a particular article [206], the authors use the SVM approach to make a short-term prediction of V_b at time t and also at time $t+1$ ("prediction of V_b"). This approach uses SVM coupled with a genetic algorithm to perform this task. A comparison is realized with an AutoRegressive Integrated Moving Average (ARIMA) model and the SVM approach developed in that paper. The authors demonstrate that the SVM approach performs better than the ARIMA model.

In another instance, it is proposed to use the cutting forces to classify the state of the tool into three states ("initial", "normal" and "severe" wear) [207]. The cutting forces are pre-processed with several techniques: normalization, statistical features and time-frequency domain. This is followed by a correlation analysis to find the best

features. These features are used as input of a Gravitational Search Algorithm–Least Square Support Vector Machine model (GSA-LSSVM). In this entry, the authors compare the results obtained with this technique with other related methods such as: K-Nearest Neighbour(k-NN), Feed Forward Neural Network (FFNN), Classification And Regression Tree (CART) and Linear Discriminant Analysis (LDA). The classification obtained with the GSA-LSSVM model outperforms the others. However, most other authors using SVM in this context compare SVM with ANN, and they show that SVM is more accurate (in classification and regression) than ANN [208–211]. Some authors also compare the results obtained with SVM and other AI techniques: [207–211].

C.1.2.5 Conclusions

Applications of SVM appeared later in the literature and remain a relatively modern approach. As this method is more dependent on the quality of the input data, some feature extraction is required to improve the quality of the inference. Fitting the correct parameter in the SVM approach is a complex task, but it appears that the quality of the inference made with this approach is at least at the same level as with an ANN approach. In practice, this method may be more difficult to implement in the industrial sector due to its sensibility to noisy data, and it needs a pre-processing approach. However, SVM has the advantage of a good generalization performance which could make this approach particularly interesting in the future.

C.1.3 SOM

Only 7 articles described this approach and are presented in Table C.5.

C.1.3.1 Input

The inputs of the SOM are presented in Figure C.7. Due to the nature of this approach, only inputs that are highly correlated with the tool wear are used. Indeed, the cutting parameters such as cutting speed, depth of cut, feed rate, etc. are not used as input of the SOM contrary to other approaches.

C.1.3.2 Pre-Processing

The pre-processing techniques used are presented in Figure C.8. It is observed that the statistical features and frequency domain features are the most used pre-processing approaches. This method is generally robust against noisy data but computing the kurtosis, skewness, average value, etc. is an easily accessible way of improving the results obtained with this approach and reducing the learning time.

C.1.3.3 Output

This approach is only able to perform classification as observed in Figure C.9.

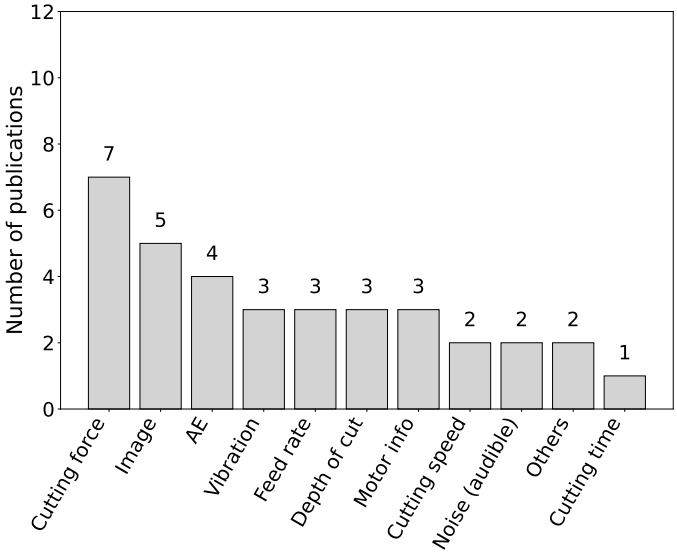


Figure C.4: Repartition of SVM inputs.

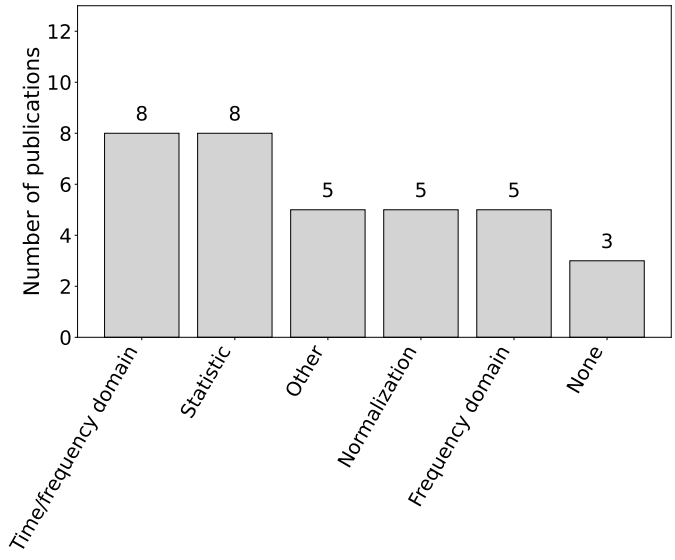


Figure C.5: Repartition of SVM pre-processing.

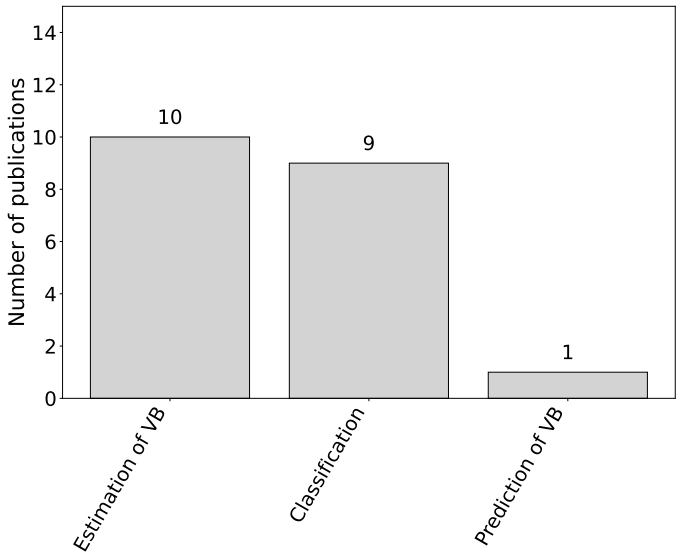


Figure C.6: Repartition of SVM outputs.

C.1.3.4 Comparison

Some articles present the SOM with another method. In [212], a comparison between SOM, SVM and k-nearest neighbour approach is realized. In [91], the use of SOM, neuro-fuzzy and ANN is discussed and software is presented for shop-floor application. In general, the SOM approach does not outperform the others but proposes a simple and accessible classification solution. However, the lack of generalization capabilities of this approach requires training under all cutting conditions, which is not always necessary with other methods.

C.1.3.5 Conclusions

The SOM approach is quite different from the others by its unsupervised learning aspect. Some authors use this aspect to try to select the best features independently of classical correlation approaches [213]. Other uses this aspect to work with imbalanced data and uni-sensor approach [212]. In an industrial application, imbalanced data often constitute the only available data, and this technique can therefore be used in this context. The advantage of low computation time is also highlighted [91].

C.1.4 HMM

The HMM approach is not strongly represented in the literature with only 3 articles: [94, 214, 215]. These articles are presented in Table C.6. They are all used to perform a classification on the tool state.

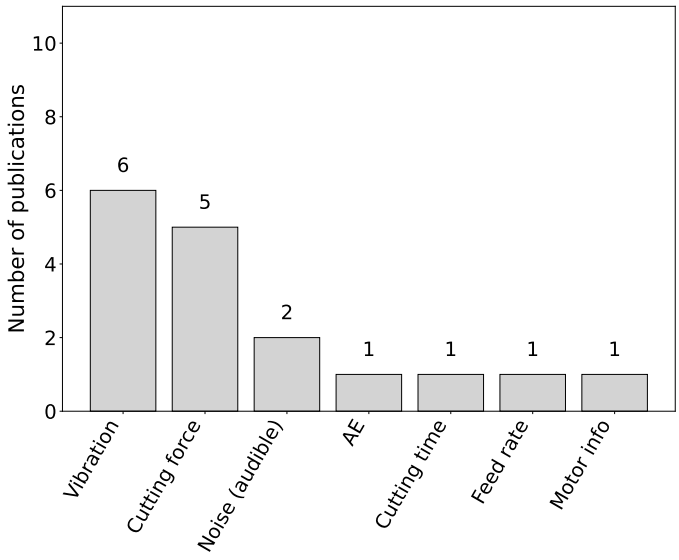


Figure C.7: Repartition of SOM inputs.

C.1.4.1 Presentation of the Articles

In Ref. [215], a discrete hidden model uses the wavelet transformation on the vibration signal in the feed direction to classify the tool state within two states ("sharp" or "worn" tool). This approach achieves a hit rate up to 97%. The authors try this approach under different training conditions (e.g., length of training data and variation of observation sequence length), and the HMM performs well in all conditions.

In Ref. [214], a Discrete Hidden Markov Model (DHMM) is used in combination with an SOM. They use the cutting force and the acceleration to monitor the state of the tool and these features are pre-processed with a FFT and coded with a SOM. Their approach performed well, and a processing time of around 0.2 s is reported, which makes this approach usable in online applications.

In Ref. [94], the use of ANN is compared to an HMM to monitor the tool wear. The cutting force signal is used as input, pre-processed with multiple techniques: statistical (mean, skewness, etc.), frequency (FFT, PSD) and time-frequency (wavelet and spectrogram). The best features are chosen by the researcher following a correlation analysis. It is concluded that both approaches perform well with the advantage of HMM to be extremely easy to train but limited to a certain amount of output values (classification).

C.1.4.2 Conclusions

Even if this approach presents only 3 papers in the literature, the results reported in these publications show that this technique is capable of realizing a good quality classification. In theory, there are Markov models capable of performing a regression, but these have not been tested yet in the literature. As the classification of the

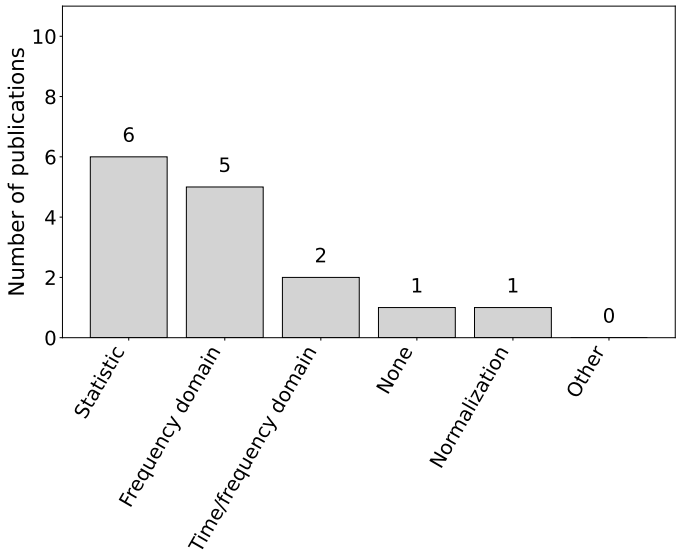


Figure C.8: Repartition of SOM pre-processing.

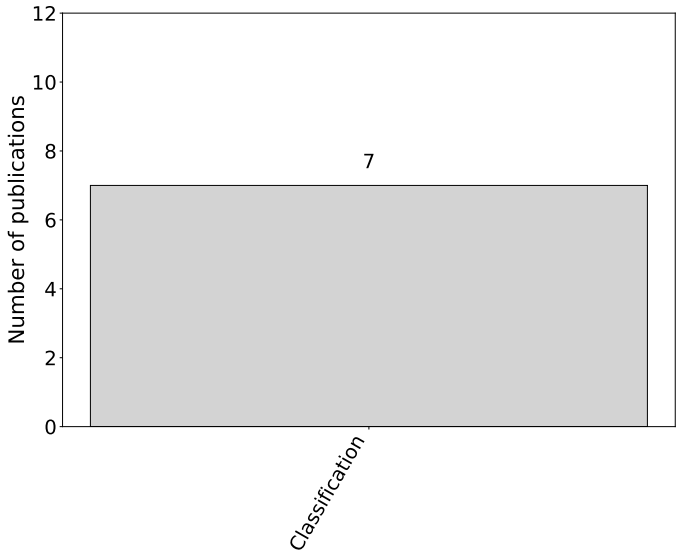


Figure C.9: Repartition of SOM outputs.

tool state is not the most common approach, the low interest in this technique can be explained by this consideration. Moreover, as stated in one of these articles, in comparison, a neural network is able to have the same precision as an HMM. However, the neural network needs a little more trial and error to determine the best architecture for learning [94]. HMM therefore does not seem to present any major advantages, which leads researchers to not favour this approach over another.

C.1.5 CNN

The use of CNN has recently emerged. Only 3 articles discuss the use of CNN to monitor the tool wear: [113–115]. They are presented in Table C.7. All of these articles have been published in 2020 showing a new interest in this techniques (see Section 4.3.2). These articles almost use the same approach.

C.1.5.1 Presentation of the Articles

In [113], the authors used the image of the insert to compute the value of V_b. This is achieved by identifying the wear area on the tool. The identification of this zone makes it possible to count the number of pixels which allows finding the dimensions of the wear zone. The authors perform a data augmentation on the initial dataset to improve the robustness of the CNN. In [115], a similar approach is followed, but the images of the insert are used directly to classify the state of the tool. In this study, the classification is made within two classes: "GO/NO GO". Both articles report accuracy greater than 90%.

Based on the dependency of chip colour on the cutting tool temperature, and the influence of tool wear on tool temperature, it has been proposed to use the chip colour to monitor the tool wear [114]. Images of the chip are pre-processed to remove the background and a kernel density estimation is performed. They show that this density estimation is more efficient on the hue channel from the Hue Saturation Value (HSV) channels of the images. This is used as an input of the CNN which performs a classification around three classes depending on the class of the wear: "New", "Medium" and "High". They achieve accuracy of above 90%. A comparison is made with a functional data analysis (FDA) classifier, and it fails to beat the CNN.

C.1.5.2 Conclusions

The late interest in this technique can be explained by the complexity to obtain a significant number of images for the datasets. Indeed, this technique uses images of the tool which can be impractical to obtain in industrial applications. These approaches work greatly in laboratory conditions but still require human intervention to achieve them. Comparing this approach with the others presented in this paper, automating this approach seems more difficult. In addition, in real production applications, turning is often performed with cutting fluids which can further complicate image capture. Indeed, the conditions for taking images must be consistent and the light must be controlled so as not to influence the neural networks [114]. Today, to our knowledge, there is no automated machine allowing the introduction of a capture system of this type which may explain the low interest

raised by this technique. This lack of interest can also be explained by the false belief that CNN is more difficult to implement than ANN [42]. Furthermore, this approach requires stopping the machining process to allow the picture to be taken which removes all the interest of monitoring in real time. However, the advantage of this approach is that it makes it possible to directly measure the wear on the tool rather than going through correlated indicators such as the cutting forces, for example. Automating this approach would be equivalent to automating the measurement of V_b, which would provide important data for other forecasting techniques. By using the chip colour instead, it could then be possible to carry out the measurement without interrupting the process [114].

C.1.6 Other

This section covers all approaches that are not related to the other categories. The articles are presented in Table C.8. It can be noted that, among these methods, there is a majority of decision trees and classifiers. These approaches are generally compared in pairs with others, hence they do not have a dedicated section in this SLR.

C.2 Systematic Literature Review - Tables

Table C.1: General summary of ANN-related publications.

Article	Method	Material	Input	Output	Pre-Processing	Comments
[81]	ANN	Inconel 718	Cutting forces AE Vibration	Estimation of VB	Time/frequency	3 layers: 3 to7 neurons in the hidden layer
[102]	Recurrent neural network (RNN)	mild steel	Cutting forces Vibration Acoustic (audible)	Estimation of VB	Statistic	Best performance with 6 to 9 neurons in hidden layers
[82]	ANN	steel EN8	Cutting speed Cutting forces Feed Depth of cut	Estimation of VB	None	3 layers: 5 inputs ; 1hidden (5 to 10 neurons); 1 output
[83]	ANN	hardened bearing steel (61HRC)	Cutting forces Vibration	Estimation of VB	Statistic	Multi layer perceptron
[216]	Regression neural networks	high manganese cast steel	Cutting speed Cutting forces Vibration Feed Depth of cut Temperature	Estimation of VB	Normalization	

Table C.1 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[101]	Generalized regression neural network with improved fruit fly optimzation	high manganese cast steel	Cutting speed Feed Depth of cut	Estimation of VB	Normalization	93% accuracy
[80]	Back Propagation NN	aluminum alloy 7050	Vibration Acoustic (audible)	Classification	Time/frequency	3 layers: 8-11-3; 3 states classification up to 92.5% accuracy
[103]	ANN	steel	Other	RUL	None	
[84]	ANN	Ti6Al4V	AE Vibration	Estimation of VB	Statistic	Best results with 3-3-1 and 3-6-1
[107]	ANN, fuzzy logic and regression (not AI)	AISI 4140	Cutting speed Feed Depth of cut	Estimation of VB	None	
[97]	ANN	AISI 4140	Cutting speed Cutting forces Vibration Feed Depth of cut	Estimation of VB	None	shape: 5-14-6-1-1

Table C.1 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[85]	ANN	AISI 316	Cutting forces AE Vibration	Other	Time/frequency Statistic	3 layers: 6-x-1. Output: percentage of tool life
[105]	ANN and case based reasoning	steel AISI 304; AISI D2; AISI 52100	Cutting speed Feed Depth of cut Other	RUL Other		Output: Tool life and tool-shim interface temperature
[217]	ANN	steel (AISI 4340)	Cutting speed Feed Depth of cut			
[106]	ANN and regression model (not AI)	3 different steel	Cutting speed Feed Depth of cut Other	Estimation of VB Other	None	Comparison with multiple linear regression model. Output: surface roughness and flank wear

Table C.1 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[99]	ANN and SVM	steel (AISI 4340 AND 52100 hardenend)	Surface texture	Estimation of VB	None	Reflectiveness of the surface; 28 ANN: 6 Multi Layer Perceptron (MLP), 3 MLP with Principal Component Analysis (MLPPCA), 3 Generalized FeedForward (GFF), 3 Radial Basis Function, 3 Time Delay Neural Network, 3 time-lag recurrent network (TLRN), 3 Recurrent Network, 2 linear regression (LR), 1 probabilistic neural network (PNN) and 1 SVM

Table C.1 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[98]	ANN and regression (not AI)	steel (AISI 4340)	Cutting speed Cutting forces Feed Depth of cut Other Temperature	Estimation of VB	Other	Architecture: 6-4-5-4-1. Comparison with regression
[86]	ANN and CANFIS	7075 alloy	Cutting speed Feed Depth of cut	Estimation of VB	None	Co-Active Neuro Fuzzy Inference System (CANFIS)
[218]	ANN	grey cast iron	Cutting speed Cutting time Feed Depth of cut	Other	None	Output: Cutting forces (estimation of flank wear)
[219]	ANN and polynomial classifiers	mild steel	Cutting speed Cutting forces AE Cutting time Feed	Estimation of VB	Other Statistic	ANN 90% accuracy and classifier 85%

Table C.1 ...*cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[220]	ANN	AISI 52100 bearing steel	Cutting speed Cutting time Feed Depth of cut Other	Estimation of VB	Normalization	3 layers: 5-7-1
[87]	ANN and regression model (not AI)	steel AISI H13	Cutting speed Feed Depth of cut	Estimation of VB Other	None	3 layers and 5 neurons in the hidden layer. Output: Flank wear and surface roughness
[88]	ANN	EN-8 steel	Cutting speed Cutting time Feed Depth of cut Other Temperature		None	3 layers and 5 neurons in the hidden layers. There is a mathematical relationship to predict the tool wear. The AI does not predict directly the wear

Table C.1 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[221]	ANN	hardened (55HRC) H-13 steel tube	Cutting speed Cutting forces Cutting time Other	Estimation of VB	Other	
[108]	Neural networks	Experiment AISI 1045 steels	Cutting speed Feed Depth of cut Motor info	Estimation of VB	Other	Machine learning process optimal linear associative memory (MLP OLAM)
[222]	Random Forest; gradient boosting regression, neural network and Support vector regression	hardened steel (58 HRC)	Cutting speed Cutting forces AE Vibration Feed Depth of cut Other	Estimation of VB	Time/frequency Statistic Frequency domain	

Table C.2: General summary of fuzzy logic related publications.

Article	Method	Material	Input	Output	Pre-Processing	Comments
[185]	Decision tree (j48) and fuzzy logic	steel	Vibration	Classification	Statistic	95% accuracy
[107]	ANN, fuzzy logic and regression (not IA)	AISI 4140	Cutting speed Feed Depth of cut	Estimation of VB	None	
[186]	Type 2 fuzzy model	Titanium Metal Matrix Composite	AE	Estimation of VB	Other	
[187]	Hybrid: ANN and fuzzy modelling	SAE 6150 chromim-vanadium alloy steel	Cutting forces AE Vibration	Estimation of VB	Statistic Frequency domain	
[188]	Takagi–Sugeno–Kang (TSK) Fuzzy Approach Based on Subtractive Clustering Method	steel	Cutting forces Feed	Estimation of VB	None	

Table C.2 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[189]	Takagi–Sugeno–Kang (TSK) Fuzzy Approach Based on Subtractive Clustering Method	steel	Cutting forces	Estimation of VB	None	
[190]	Fuzzy method	/	Cutting forces Feed	Estimation of VB	None	2 approaches: one with the knwoledge given by the researcher and one automatic genetic algorithm

Table C.2 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[95]	ANN; Fuzzy decisions support logic (rules set by the researcher); Artificial neural network based fuzzy inference system (ANNBFIS)	/	Cutting forces Feed	Estimation of VB	None	For ANN 3 layers: 5 neurons in the hidden layers

Table C.3: General summary of neuro-fuzzy related publications.

Article	Method	Material	Input	Output	Pre-Processing	Comments
[191]	ANFIS	steel 1215.	Cutting forces	Estimation of VB	Time/frequency	5 layers; 4 different activation function: sigmoid, triangular, gaussian, bell-shaped
[192]	Type 2 basis function network	steel/ inconel/ Ti6Al4V	AE Vibration Motor info	Estimation of VB	Normalization	
[193]	ANFIS	steel	Cutting forces Feed	Estimation of VB	Statistic	
[194]	ANFIS	hardened JIS S45C carbon steel	Cutting speed Feed Depth of cut Other	Estimation of VB	Other	Features extraction with Z-filter
[195]	ANFIS	From another study	Cutting forces AE Vibration Cutting time	Estimation of VB	Normalization	Inductive, transductive and evolving neuro-fuzzy systems

Table C.3 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[196]	ANFIS	cryogenically treated AISI M2 high speed steel	Cutting speed Cutting time Other	Estimation of VB	None	
[197]	ANFIS	D2 X115Cr VMo121 steel	Surface texture Temperature Motor info	Estimation of VB	None	Accuracy up to 98%
[198]	Fuzzy with a new Takagi–Sugeno–Kang (TSK) modelling	steel	Cutting forces Feed	Estimation of VB	None	
[86]	ANN and CANFIS	7075 alloy	Cutting speed Feed Depth of cut	Estimation of VB	None	Co-Active Neuro Fuzzy Inference System
[199]	Transductive-Weighted Neuro-Fuzzy Inference System (TWNFIS)	From another study	Cutting forces AE Vibration	Estimation of VB	None	More than 90% accuracy

Table C.3 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[200]	ANFIS compared with Hybrid approach	steel EN 24 and FG15	Cutting forces AE Vibration Cutting time	Estimation of VB	Normalization	comparison between Transductive Weighted Neuro-Fuzzy Inference System (TWNFIS), ANFIS and Hybrid Incremental Modeling (HIM). HIM outperform the other
[91]	SOM, neuro-fuzzy, ANN	EN-8 Steel	Cutting forces	Classification Estimation of VB	None	ANN: 3 layers: 2-3-1; SOM: classification
[201]	Evolving extended Takagi–Sugeno based neuro-fuzzy algorithm	Inconel 718	Cutting forces AE	Estimation of VB	Statistic	

Table C.3 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[202]	Localized fuzzy neural network (LFNN)	45 middle carbide steel workpiece	Cutting forces AE	Classification	Time/frequency	
[95]	ANN; Fuzzy decisions support logic (rules set by the researcher); Artificial neural network based fuzzy inference system (ANNBFIS)	/	Cutting forces Feed	Estimation of VB	None	For ANN 3 layers: 5 neurons in the hidden layers
[203]	ANFIS	steel	Cutting forces AE	Classification	Statistic	Best with triangular and bell function
[204]	ANFIS	FG15 cast iron	Cutting forces AE Vibration Cutting time	Estimation of VB	Other	Around 82% accuracy

Table C.3 *...cont.*

Article	**Method**	**Material**	**Input**	**Output**	**Pre-Processing**	**Comments**
[205]	ANFIS and Back Propagation NN	Stainless steel 304 L	Cutting forces	Classification Estimation of VB	Statistic	2 states classification

Table C.4: General summary of SVM-related publications.

Article	Method	Material	Input	Output	Pre-Processing	Comments
[78]	SVM	high strength steel (45CrNiMoVA)	Cutting forces Vibration Surface texture	Estimation of VB	Time/frequency Statistic Frequency domain	Grid search-SVR, Genetic algorithm SVR, Particle swarm optimizer-SVR and Grid-search SVR
[207]	Gravitational search algorithm coupled with least square support vector machine model	steel	Cutting forces	Classification	Time/frequency Normalization Statistic	3 states classification
[212]	SOM; SVM; k-nearest neighbour	steel AISI D6	Vibration	Classification	Normalization Statistic Frequency domain	Accuracy up to 92%; 2 states classification
[206]	SVR modified	/	Other	Prediction of Vb	Normalization	Forecasting at $t+1$

Table C.4 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[223]	SVM and decision tree	steel (1.4542)	Motor info	Classification	Time/frequency Normalization Frequency domain	2 states classification; above 90% accuracy
[224]	SVM	ZMn13 high manganese cast steel	Cutting speed Cutting time Feed Depth of cut	Estimation of VB	None	Nelder–Mead Leave One Out-Least squares Support Vector Machines (NM-LOO-LSSVM), Particle Swarm Optimization-LSSVM (PSO-LSSVM) and LSSVM. Best performances with NM-LOO-LSSVM

Table C.4 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[225]	SVM	superalloy	AE Acoustic (audible) Feed Depth of cut Motor info	Classification Estimation of VB	Time/frequency Other	Optimization with genetic alogrithms

Table C.4 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[99]	ANN and SVM	steel (AISI 4340 AND 52100 hardened)	Surface texture	Estimation of VB	None	Reflectiveness of the surface; 28 ANN: 6 Multi Layer Perceptron (MLP), 3 MLP with Principal Component Analysis (MLPPCA), 3 Generalized FeedForward (GFF), 3 Radial Basis Function, 3 Time Delay Neural Network, 3 time-lag recurrent network (TLRN), 3 Recurrent Network, 2 linear regression (LR), 1 probabilistic neural network (PNN) and 1 SVM

Table C.4 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[222]	Random Forest; gradient boosting regression, neural network and Support vector regression	hardened steel (58 HRC)	Cutting speed Cutting forces AE Vibration Feed Depth of cut Other	Estimation of VB	Time/frequency Statistic Frequency domain	
[208]	Classifier: Decision tree, K-neighbour, Random forest; Regressor: MLP, Random Forest, SVM, Naive Bayes, gradient boosting	Generation on Python (turning case)	Cutting forces Motor info	Classification Estimation of VB	None	2 states classification
[209]	v-SVR	steel #50	Cutting forces	Estimation of VB	Time/frequency Statistic	Comparison with BPNN
[226]	SVM	C50 steel	Surface texture	Classification	Other	3 states classification
[227]	SVM	C50 steel	Surface texture	Estimation of VB	Other	

Table C.4 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[228]	SVM	GH4169	AE	Classification	Time/frequency Other Frequency domain	4 states classification
[210]	LSSVM	steel AISI 1040	Cutting forces Acoustic (audible)	Estimation of VB	Statistic	
[229]	SVM	ASSAB705/760 steel	AE	Classification	Time/frequency Statistic	80-85% accuracy
[211]	v-SVR	#50 steel	Cutting forces	Classification	Normalization Statistic	2 states classification. 96% accuracy
[230]	SVM	AISI 01	Surface texture	Estimation of VB	Other	Genetic algorithm

Table C.5: General summary of SOM-related publications.

Article	Method	Material	Input	Output	Pre-Processing	Comments
[212]	SOM; SVM; k-nearest neighbour	steel AISI D6	Vibration	Classification	Normalization Statistic Frequency domain	Accuracy up to 92%; 2 states classification
[213]	SOM	steel EN1A1	Cutting forces Vibration Acoustic (audible) Feed Motor info	Classification	Statistic Frequency domain	
[231]	SOM	mild steel	AE Vibration Cutting time	Classification	Statistic	
[232]	Self organizing spiking neural network map	mild steel	Cutting forces Vibration Acoustic (audible)	Classification	Statistic Frequency domain	
[91]	SOM, neuro-fuzzy,ANN	EN-8 Steel	Cutting forces	Classification Estimation of VB	None	ANN: 3 layers: 2-3-1; SOM: classification
[79]	SOM	Industrial in South Africa (piston manufacturer)	Cutting forces Vibration	Classification	Time/frequency Statistic Frequency domain	3X3 output

Table C.5 *...cont.*

Article	**Method**	**Material**	**Input**	**Output**	**Pre-Processing**	**Comments**
[233]	Self organizing map (SOM)	cast iron	Cutting forces Vibration	Classification	Time/frequency Statistic Frequency domain	

Table C.6: General summary of HMM-related publications.

Article	Method	Material	Input	Output	Pre-Processing	Comments
[214]	HMM	45# steel	Cutting forces Vibration	Classification	Other Normalization Frequency domain	
[94]	ANN and HMM	From another study	Cutting forces	Classification Estimation of VB	Time/frequency Statistic Frequency domain	Comparison between the 2 approaches: similar results
[215]	Discrete Hidden Markov Model (DHMM)	AISI 8620 steel	Vibration	Classification	Time/frequency	2 states classification; more than 97% accuracy

Table C.7: General summary of CNN-related publications.

Article	Method	Material	Input	Output	Pre-Processing	Comments
[113]	convolutional neural network (U-Net)	45# steel	Surface texture	Estimation of VB	Other	
[114]	CNN and functional data analysis	low carbon steel (25CrMo4)	Surface texture	Classification	Other	3 states classification; around 90% accuracy;
[115]	CNN	steel	Surface texture	Classification	Other	2 states classification

Table C.8: General summary of publications related to various other methods.

Article	Method	Material	Input	Output	Pre-Processing	Comments
[234]	Logical Analysis of data (AI classifier); Logical analysis of survival curves (LASC)	titanium metal matrix composite	Cutting forces	Remaining Useful Life (RUL)	None	Non AI outperform AI
[212]	SOM; SVM; k-nearest neighbour	steel AISI D6	Vibration	Classification	Normalization Statistic Frequency domain	Accuracy up to 92%; 2 states classification
[235]	Nearest-Neighbour-like algorithm using non-nested Generalized Exemplar (NNGE)	steel	Vibration	Classification	Statistic	4 states classification up to 100% accuracy
[185]	Decision tree (j48) and fuzzy logic	steel	Vibration	Classification	Statistic	95% accuracy

Table C.8 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[223]	SVM and decision tree	steel (1.4542)	Motor info	Classification	Time/frequency Normalization Frequency domain	2 states classification; above 90% accuracy
[236]	linear regression, decision trees, random forests, k-nearest neighbours, adaptive boosting, bootstrap aggregating and gradient boosting	inconel 718	Cutting forces	Estimation of VB	Statistic	
[237]	Hybrid dynamic neural network (HDNN)	steel (different alloy)	Motor info	Estimation of VB	Statistic	
[238]	Logical analysis of data (LAD)	steel (AISI 1045)	Cutting forces Feed	Classification	None	

Table C.8 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[105]	ANN and case based reasoning	steel AISI 304; AISI D2; AISI 52100	Cutting speed Feed Depth of cut Other	RUL Other		Output: Tool life and tool-shim interface temperature
[239]	Novelty detection	steel	Cutting forces	Estimation of VB	Statistic Frequency domain	Up to 100% accuracy
[219]	ANN and polynomial classifiers	mild steel	Cutting speed Cutting forces AE Cutting time Feed	Estimation of VB	Other Statistic	ANN 90% accuracy and classifier 85%
[240]	Hybrid neural network	AISI 52100 bearing steel	Cutting speed Cutting time Feed Depth of cut Other	Estimation of VB Other	Other	Training with an extended Kalman filter. Activation function: Sigmoid and optimization of the network

Table C.8 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[241]	Hybrid approach	/	Cutting forces	Estimation of VB	Other	Physical model use to convert the features
[114]	CNN and functional data analysis	low carbon steel (25CrMo4)	Surface texture	Classification	Other	3 states classification; around 90% accuracy;
[242]	Naive Bayes classifier	Oil hardened Nickel steel	Cutting forces Vibration	Classification	Time/frequency	4 states classification. 96% accuracy with the cutting forces and 70% with the vibration signal
[243]	Random forest; Gradient boosting method (GBM); regression model	steel	Cutting forces Vibration	Estimation of VB Other	Frequency domain	Output: tool wear and surface roughness

Table C.8 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[222]	Random Forest; gradient boosting regression, neural network and Support vector regression	hardened steel (58 HRC)	Cutting speed Cutting forces AE Vibration Feed Depth of cut Other	Estimation of VB	Time/frequency Statistic Frequency domain	
[208]	Classifier: Decision tree, K-neighbour, Random forest; Regressor: MLP, Random Forest, SVM, Naive Bayes, gradient boosting	Generation on Python (turning case)	Cutting forces Motor info	Classification Estimation of VB	None	2 states classification

Table C.8 *...cont.*

Article	Method	Material	Input	Output	Pre-Processing	Comments
[244]	Functional Series Time dependent AutoRegressive Moving Average (FSTARMA)	AISI 1045 steels	Vibration	Other		Output: Vibration signal
[245]	Metacognitive learning approach (Rclass)	low carbon alloy steel	Cutting speed Cutting forces Vibration Feed Depth of cut	Classification	None	
[246]	Hierarchical algorithm	Inconel 625	Cutting forces AE Vibration	Other	Time/frequency Statistic Frequency domain	Output: percentage of tool life
[247]	Self adjusting monitoring system	45 middle carbide steel workpiece	Cutting forces AE	Estimation of VB	Time/frequency Statistic Frequency domain	Regroups multiple approach