Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 132 (2025) 50–55

www.elsevier.com/locate/procedia

12th CIRP Global Web Conference (CIRPe 2024)

Image processing with deep-learning and transfer learning for cutting tool degradation monitoring

Lorenzo Colantonio*,[a], Lucas Equeter[a], Hugo Giovannelli[a], Pierre Dehombreux[a], Saïd Mahmoudi[b], François Ducobu[a]

[a]*Machine Design and Production Engineering Lab, University of Mons, Place du parc, 20, 7000, Mons, Belgium*
[b]*Computer Science Unit, University of Mons, Place du parc, 20, 7000, Mons, Belgium*

* Corresponding author. *E-mail address:* lorenzo.colantonio@umons.ac.be

Abstract

Monitoring the degradation of cutting tools is of utmost importance in the manufacturing world. Tools with substantial wear fail to produce high-quality parts in terms of geometry, residual stress, and surface finish. Furthermore, replacing tools in a non-optimal manner can lead to increased production costs and downtime. Therefore, monitoring the condition of the tool is essential to avoid these additional costs and ensure good production quality. This article explores various classification models, specifically VGG19, EfficientNetV2, and Vision Transformers. These models classify the state of tools using their images. Using transfer learning, a comparison of the best-performing artificial intelligence-based image analysis models is conducted to identify those most suitable for monitoring cutting tools. A comparative analysis of their generalizability, performance and explainability is realized. The model with the best performance is VGG19 with an accuracy of 94%, followed by EfficientNetV2 and ViT with an accuracy of 87%. A full comparison of these results is carried out.

Keywords: Cutting Tool; Degradation; Monitoring; Deep-Learning; Transfer Learning

1. Introduction

During machining operations, the interaction between tool and workpiece leads to tool wear. This wear results in imperfect cutting, which can reduce the quality of parts produced. Factors such as the geometry of the workpiece, its residual stresses and its surface finish are all dependent on the condition of the tool. Wear is a major problem in machining operations, and it is estimated that the cost of cutting tools can represent between 3% and 12% of production costs. Tool failure, due to wear, can account for up to 20% of production stoppages [1]. To avoid excessive tool wear, tools need to be replaced regularly. Industrial practice is to carry out tests at the start of production and to take safety margins to determine the time to replace the tool. Also, sometimes, replacement is based on the judgement of the machining operator to decide whether or not the tool needs to be replaced. The solutions currently in use clearly lack objective criteria for replacing the tool at the optimal moment, leading to waste and costs. In order to determine the optimum tool replacement, it is useful to use decision support methods to determine whether the tool is worn or still usable.

Artificial intelligence (AI) is increasingly being used to monitor cutting tools as a decision support methods. There are several AI techniques for monitoring the degradation of cutting tools, typically divided into two categories: indirect and direct. Indirect approaches relies on sensors installed in the machine to estimate the state of the tool from cutting signals. Extensive review present the strategies that have been developed in this field [2]. These indirect approaches are often specific to a given experimental condition or set-up, and are difficult to apply to more complex industrial contexts.

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 precise, 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 [3]. 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

10.1016/j.procir.2025.01.009

intelligence methods overcome these issues, by their ability to reliably identifying wear even in such conditions [3]. Numerous AI methods have been employed to identify or categorize the region of tool wear using images. For example, Pagani et al utilized the color of the chips to infer the tool's condition [4]. In 2019, Wu et al compared their own model with VGG16 to distinguish between different types of wear. Their model was quicker, but the performances were comparable between the two models [5]. Some researchers have also used images of the vibration spectrogram along with transfer learning models like VGG16, LeNet, etc., to classify the state of the tool [6]. Other researchers have employed U-Nets to directly segment the wear zone on tool images, both in milling [7] and turning [8]. In the field of image classification or segmentation, 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. This is why transfer learning is often the preferred approach in this type of application. 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.

The novelty of this article therefore lies in the use and comparison of 3 AI methods for image classification using transfer learning. The 3 baseline architecture are: VGG19 [9], EfficientNetV2 [10] and Visual Transformers [11]. 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. The effectiveness of these methods in response to these alterations is examined. Explainability analysis is also employed to highlight the challenges experienced by these networks. Finally a discussion about the most suitable to industrial scenario is carried out. In contrast to the manual wear analyses currently carried out in industry, this approach automatically detects and takes advantage of deep learning capabilities to robustly and automatically recognise the wear zone.

2. Database

In order to train the different approaches presented in this article, a turning database coming from experimental turning tests on C45 steel bars is used. These bars were machined with a CNMG120404-M3 TP40 tool in a tool holder DCLN R 2020K12-M on a Weiler E35 lathe. The tool is one of the lowest grades in order to favour the appearance of wear and at the same time to reduce the quantity of material used during the tests. A total of 30 tools were used during the straight turning testing campaign. 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. These tools were used under similar cutting conditions but with varying cutting speeds (Table 1). The tool was inspected every 2 minutes 40 seconds using a Byameyee 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 Fig. 1.

Table 1. Cutting conditions during experimental turning tests

Test	Cutting Speed [m/min]	Feed [mm/rev]	Depth of cut [mm]
1 to 10	260	0.2	1
11 to 15	250	0.2	1
16	240	0.2	1
17 to 20	265	0.2	1
21 to 30	Variable during life: 240 to 260	0.2	1

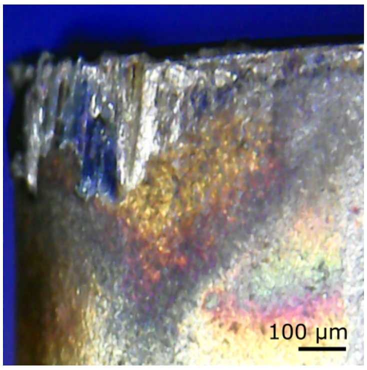


Fig. 1. Image of an insert. VB: 280 μm

In this article, 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 (Fig. 2). A tool with wear between 0 and 150 microns is deemed new, one with wear between 150 and 300 microns is considered with moderate wear, and a tool with wear exceeding 300 microns 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 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 microns. This threshold is established by the ISO 3685 standard [12]. The database, however, lacks homogeneity: there are more images of new tools than worn tools, which can be a limitation when training AI. To counteract this imbalance, the data are 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 [13]. Two types of augmentation are taken into account: image manipulation and lighting modification During training, each image is randomly augmented using a combination of the following modifications:

- 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 Fig 1 is shown in Fig. 3(a)
  - **Rotation**. This technique rotates the image. The rotation angle is randomly chosen between -20 to 20 degrees. This modification applied on Fig 1 is shown in Fig. 3(b)
- 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 Fig 1 is shown in Fig. 3(c)
  - **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 Fig 1 is shown in Fig. 3(d)

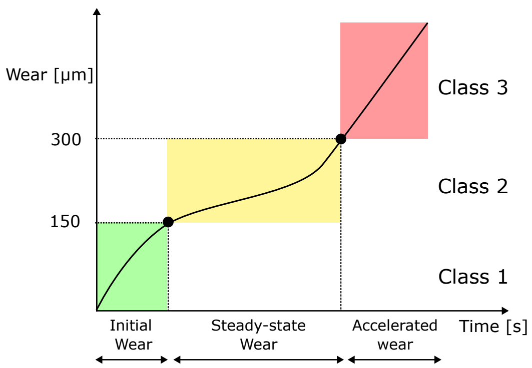


Fig. 2. Classification of the tool degradation into 3 classes: green - New, yellow - Moderate wear, red - Worn tool

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.

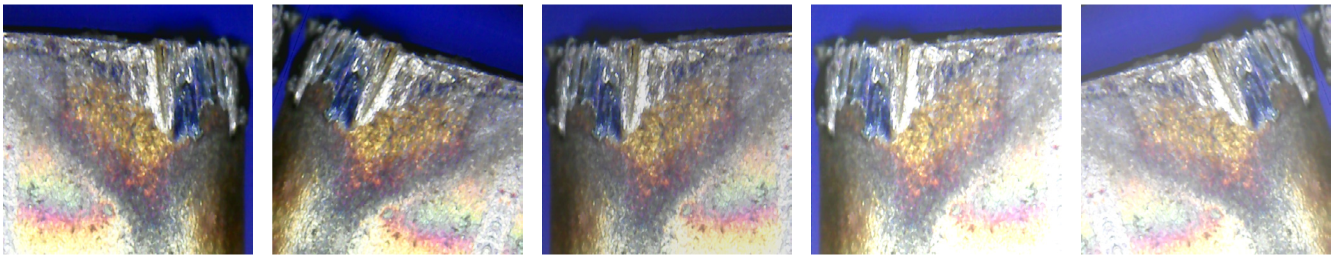
In order to test the ability of an AI approach to learn this training domain, a testing database needs to be defined. This database must represent all existing cases and it consist of 5 images of each class. Each of this image is tested six times under distinct configurations: once in its original form, once flipped, once rotated, once after adjusting its contrast, once after altering its brightness and finally once with all the previous modifications (Fig. 3(e)). This process ensures that each test image is thoroughly evaluated under various conditions, enhancing the robustness of the testing phase.

3. Models

Artificial intelligence offers multiple strategies for image classification. The most straightforward approach involves building and training a deep neural network from scratch on the database. While this method is certainly viable, it's often more advantageous to utilize a pre-existing, pre-trained architecture that has been trained on extensive databases. This makes it possible to take advantage of an architecture that is already capable of extracting valuable features from images thanks to its training. This approach, known as transfer learning (Fig. 4), involves using an existing network and modifying its output to adapt it to a new classification task [14].

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** [9]. Developed in 2014, VGG19 is a deep convolutional neural network that aims at simplicity. It is composed of 16 3x3 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** [10]. 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)** [11]. 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. 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



(a) (b) (c) (d) (e)

Fig. 3. Image transformations applied on image from Fig. 1 (**a**) Horizontal flip (**b**) Image rotation (**c**) Modification of contrast (**d**) Modification of brightness (**e**) All modifications applied

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.

To adapt these models to classify the state of the tool, the last layers are modified to classify the tool state (Fig. 4). 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.

All the approaches are trained under comparable conditions with an AMD Ryzen 9 7950 X3D CPU. The input for each model is a color image with dimensions of 224 by 224 pixels. The Adam optimizer is employed during the training process. The loss function "Categorical crossentropy" is used. The metrics for evaluation are "precision", "recall", "F1-score" and "accuracy". A maximum of 300 epochs is set. To avoid overfitting and eliminate unnecessary calculations, an early stopping mechanism is implemented. This mechanism stops the learning process if there is no improvement in network performance over a period of 60 epochs. To obtain optimal results, the learning rate is progressively reduced as the model approaches convergence.

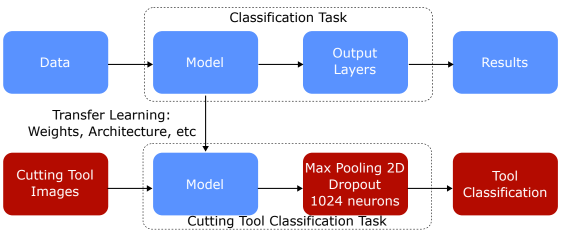


Fig. 4. Transfer learning principle: A given architecture is reused for a different tasks.

4. Results

Table 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 article. 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, taking approximately 22 minutes. Once trained, this model is capable of making a prediction, known as inference time, in 40 ms. The data presented in the table indicates that the architecture retains its ability to accurately identify tool wear, regardless of the changes made to the image.
- 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. However, even with the original image (referred to as 'Initial' in Table 2), it makes mistakes in identifying class 2 and 3. This model is quicker to train than VGG19, taking only 7 minutes. It's worth noting that this model reached its best performance quite fast, in just 65 cycles of training. 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 inference time, is similar to VGG19, at 80 milliseconds.
- 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 a total training time of around 3.5 hours for 300 epochs. The inference time is also the longest with 2 s per image.

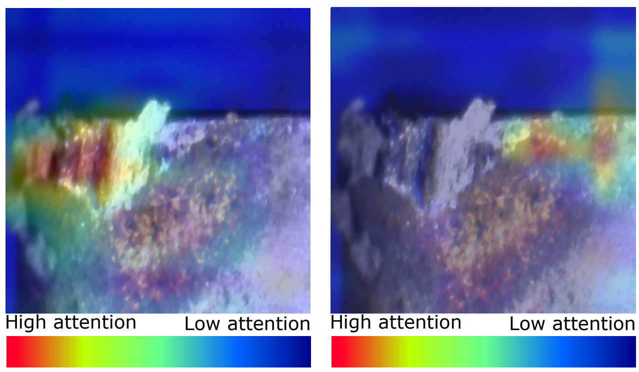
A straightforward comparison of results serves as a useful performance indicator, but it does not fully determine whether one method outperforms another. This is particularly true when

Table 2. Comparison of results for the different methods segmented by class and the type of image modification applied

		Initial	Flipped	Rotated	Contrasted	Brightness	All	Precision	Recall	F1 Score	Accuracy
VGG19	Class 1	100%	80%	100%	100%	100%	100%	94%	97%	0.95	94%
	Class 2	100%	100%	80%	80%	100%	100%	90%	93%	0.92	
	Class 3	100%	100%	80%	100%	100%	80%	100%	93%	0.97	
EfficientNetV2	Class 1	100%	100%	100%	100%	100%	100%	83%	100%	0.91	87%
	Class 2	80%	80%	60%	80%	80%	100%	80%	80%	0.8	
	Class 3	80%	80%	80%	80%	80%	80%	100%	80%	0.89	
ViT	Class 1	100%	100%	100%	100%	100%	100%	86%	100%	0.92	87%
	Class 2	100%	80%	60%	100%	80%	100%	79%	87%	0.83	
	Class 3	80%	80%	60%	80%	60%	80%	100%	73%	0.85	

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 microns, so in a strict case, a tool with 299 microns of wear would still be considered as moderate wear being only 1 micron 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 3 shows the position of errors made by the different architectures in different classes. A buffer zone of 50 microns, 25 for each class, is added between the classes, creating a transition zone between classes 1 and 2, and between classes 2 and 3. The first zone covers wear from 125 to 175 microns (transition from class 1 to 2), and the second covers wear from 275 to 325 microns. Analysis of the results in this zone reveals where the boundary between a usable tool and a worn one becomes unclear. Table 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 microns 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.

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 [15]. 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. Fig. 5(a) 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 methods have identified that the distinction between a new tool and a worn tool is attributed to the amount of flank wear. Fig. 5(b) shows the VGG19 attention map for



(a) (b)

Fig. 5. Attention map obtain with a Grad-CAM analysis of an image classified with VGG19. (**a**) Attention map of a correctly classified images, the attention map is located on the wear. (**b**) Attention map of an incorrectly classified image with VGG19. The image is the same as Fig. 5(a) but with a variation of contrast. This change of contrast modify the attention zone that is not on the correct part of the wear of the tool.

the classification of the same image as Fig. 5(a) 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.

5. Conclusion

This article 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

Table 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: 0 to 125 μm	Transition 1- 2: 125 to 175 μm	Class 2: 175 to 275 μm	Transition 2-3: 275 to 325 μm	Class 3: 325+ μm
VGG19	1	0	2	1	1
EfficientNetV2-M	0	0	6	6	0
ViT	0	1	3	1	7

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 model are derived from experimental tests conducted during turning operations. 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.

In terms of speed, EfficientNetV2 was the fastest model to train and query, with a training time of just 7 minutes and an inference time of 80 ms. VGG19, while slightly faster in querying (70 ms), took more than three times longer to train, with a time of 22 minutes. The ViT approach was the slowest of all, with a training time of 3 hours and an inference time of 2 seconds.

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.

In conclusion, for databases similar to the one presented in this article, we recommend using CNN approaches such as VGG19 and EfficientNetV2 to classify the state of cutting tools from their images. Thanks to transfer learning, it is possible to detect excessive tool wear and therefore replace it at the most optimal time.

Future studies could explore detecting and classifying tool defects during machining, like tool's plastic deformation, unusual damage, etc. Another direction of research is to automatically identify the wear zone and damages and measure the wear based on this identification. The attention map obtained in this article indicates that the networks can automatically identify the wear zone. An image segmentation network would make it possible to measure the wear zone and thus predict a remaining useful life of the tool. In this study, the database is augmented and limited to a single type of tool. An analysis on a more diversified database could also help industries to better understand the implementation and limitation of these techniques.

References

[1] R. U. Baig, S. Javed, M. Khaisar, M. Shakoor, and P. Raja. 2021. Development of an ANN model for prediction of tool wear in turning EN9 and EN24 steel alloy. Adv. Mech. Eng., vol. 13, no. 6. p.16878140211026720.

[2] Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (2021). A systematic literature review of cutting tool wear monitoring in turning by using artificial intelligence techniques. Machines, 9(12), 351.

[3] P. Muthuswamy and K. Shunmugesh 2023. Artificial intelligence based tool condition monitoring for digital twins and industry 4.0 applications. Int. J. Interact. Des. Manuf. IJIDeM, vol. 17, no. 3, pp. 1067–1087.

[4] Pagani, L.; Parenti, P.; Cataldo, S.; Scott, P.J.; Annoni, M. Indirect cutting tool wear classification using deep learning and chip colour analysis. *Int. J. Adv. Manuf. Technol.* **2020**, *111*, 1099–1114.

[5] Wu, X., Liu, Y., Zhou, X., & Mou, A. (2019). Automatic identification of tool wear based on convolutional neural network in face milling process. Sensors, 19(18), 3817.

[6] Patil, S. S., Pardeshi, S. S., & Patange, A. D. (2023). Health Monitoring of Milling Tool Inserts Using CNN Architectures Trained by Vibration Spectrograms. CMES-Computer Modeling in Engineering & Sciences, 136(1).

[7] Bergs, T., Holst, C., Gupta, P., & Augspurger, T. (2020). Digital image processing with deep learning for automated cutting tool wear detection. Procedia Manufacturing, 48, 947-958.

[8] Miao, H., Zhao, Z., Sun, C., Li, B., & Yan, R. (2020). A U-Net-based approach for tool wear area detection and identification. IEEE Transactions on Instrumentation and Measurement, 70, 1-10.

[9] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[10] Tan, M., & Le, Q. (2021, July). Efficientnetv2: Smaller models and faster training. In International conference on machine learning (pp. 10096-10106). PMLR.

[11] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

[12] ISO 3685—Tool Life Testing with Single-Point Turning Tools. 1993. Available online: https://www.iso.org/fr/standard/9151.html (accessed on 27 may 2024).

[13] Maharana, K., Mondal, S., & Nemade, B. (2022). A review: Data preprocessing and data augmentation techniques. Global Transitions Proceedings, 3(1), 91-99.

[14] Iman, M., Arabnia, H. R., & Rasheed, K. (2023). A review of deep transfer learning and recent advancements. Technologies, 11(2), 40.

[15] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE international conference on computer vision (pp. 618-626).