Large Model Behaviour on Affective Use Cases.

Hugo Bohy

hugo.bohy@umons.ac.be

Tuesday 9th September, 2025

A dissertation submitted to the Faculty of Engineering
of the University of Mons, for the degree of Doctor of Philosophy in Engineering Science

Supervisor: Prof. T. Dutoit
Co-Supervisor: Prof. F. Moiny

Jury members

Prof. **Sidi Mahmoudi** - University of Mons, President
Prof. **Stéphane Dupont** - University of Mons, Secretary
Prof. **Thierry Dutoit** - University of Mons, Supervisor
Prof. **Francis Moiny** - University of Mons, Co-Supervisor
Prof. **Mohammad Soleymani** - University of Southern California
Prof. **Jean Vanderdonckt** - UC Louvain
Dr. **Kevin El Haddad** - University of Mons

"It is easy to make things hard, but hard to make them easy."
Rutger Bregman

Abstract

The development of AI has created sophisticated applications, but they often lack the ability to understand non-verbal cues essential for human communication. This thesis addresses that gap by exploring audio-visual deep learning methods for Affective Computing.

We believe that analysing deep learning model behavior can improve performance and build trust by making their decision-making processes more transparent. Our work focuses on processing voice signals and facial expressions, specifically smiles and laughter, as key indicators of emotional states.

The main contributions of this research are fourfold. First, we enhanced three existing datasets by adding annotations for speaker / listener roles and the intensity of smiles and laughter. Then, using LSN-TCN, a deep learning-based neural network, we analyzed how fusing audio and visual feature representations impacts the detection of smiles and laughter. We also implemented Social-MAE, an advanced multimodal system that effectively encodes facial and vocal information for tasks like emotion recognition. Finally, we explored a novel method to separate affective information from existing deep learning systems without compromising their performance by using an auxiliary network.

This thesis provides open-source methods to leverage non-verbal cues, paving the way for more sophisticated and empathetic AI systems with potential applications in social and clinical settings.

Acknowledgements

My four years as a PhD candidate have been marked by the people that supported me through it. I can not express how grateful I am to Kevin for the advices, the guidance and the vast amount of bad jokes that kept me going. You are a incredible mentor, collegue and friend. I'm glad I have been able to work with you since my 3rd year bachelor.

I am also grateful to Thierry, whose kindness and great leadership bring a peaceful and joyful atmosphere in the lab. This environment made such a difference in many ways to help me fulfill the work I was doing. Thank you for the supervision you provided to help me succeed.

I would also like to extend my thanks to Francis for his hospitality and support during the many Physics labs. Looking back at my scores from those labs, it's clear that with hard work, anything is possible. Being an assistant for your classes was an enriching experience.

I want to express my deepest thanks to all the members of my committee and the defense jury. Your feedback and guidance were crucial, and I truly appreciate the time and effort you invested in my work. The defense was a challenging but rewarding experience, and I am grateful for your support.

A huge thank you to all my collegues from both Physics and ISIA Lab, with whom I share memories that will stick with me for years. I am certain your presence either distract me enough to get through the tough times or brightened my days even when the sun was hidden.

Finally, I would like to take a moment to thank my family and friends. These four years of thesis work, which could have been a lonely period, were anything but thanks to you. Your constant support through the trials and tribulations and your encouragement helped me see this work through to the end, even in moments of doubt. Thank you.

Contents

List of acronyms

AC	Affective Computing	4
AE	Auto-Encoder	23
AI	Artificial Intelligence	4
ASD	Autism Spectrum Disorders	5
ASR	Automatic Speech Recognition	100
AST	Audio Spectrogram Transformer	101
AU	Action Unit	49
CNN	Convolutional Neural Network	12
DCT	discrete cosine transform	34
DL	Deep Learning	4
DNN	Deep Neural Network	6
ECG	Electrocardiogram	4
EEG	Electroencephalogram	51
EMG	Electromyography	51
FACS	Facial Action Coding System	49
FC	fully connected	21
FFN	Feed-Forward Network	26
FFT	Fast Fourier Transform	103
FN	False Negative	29
FP	False Positive	29
fps	frames per second	36
GAP	Global Average Pooling	20
GDPR	General Data Protection Regulation	47
GSR	Galvanic Skin Response	51
HRI	Human-Robot Interaction	47
HOG	Histogram of Oriented Gradients	36
ML	Machine Learning	6
MLP	Multi-Layer Perceptron	11

Chapter 1

Scope and Contributions

Contents

1.1 Introduction & Motivations

The development of Artificial Intelligence (AI) depends on numerous factors, such as the emergence of new technologies and new computational powers, and has spread in various themes such as:

- Research focusing on detection and preventing danger for humans in situations like railroad tracks or warehouses;
- Development of autonomous cars and the ethical choices behind unforeseeable events like accidents or road works;
- Chatbots conversing with humans on diverse subjects.

The specific domain of Affective Computing (AC) is a branch that aims at interpreting, understanding and generating signals that takes emotions into account. One common application is the conversational agents, which have widely improved during the last few years with tools from OpenAI or Google such as ChatGPT[1] or Gemini[2]. These tools not only process text, but can also accept audio and visual requests, enabling them to perform affective tasks such as Speech Emotion Recognition. They provide an impressive combination of quality and performance but they lack clarity on the training and inference methodologies.

Deep Learning (DL) methods, commonly used for complex tasks for a decade now, have been able to increase the state of the art in almost all domains. Although impressive for detection, recognition and data generation, their decision making relies on complex features that are hardly understandable for humans, in opposition to engineered features previously used.

The objective of the present work is to explore audio-visual DL approaches to perform affective tasks. They have a specific behaviour to process audio and visual data. Our assumption is that analysing their behaviour would allow humans to better understand the decision process or some hidden data characteristics. Intuitively, understanding a decision process enhances the trust in the content decoded from a user affective state.

In AC, one of the goal is to allow virtual agents to better understand human non-verbal communication. This is achieved in several ways:

- in text, with sentiment analysis where the "emotional" content of a phrase, a paragraph or a whole document is inferred;
- in physiological signals like Electrocardiogram (ECG), by understand the relationship between affective states and body behaviours;
- in visual data, either static images or dynamic videos, as the body and the face communicates countless pieces of information per second;

[1] https://chatgpt.com/
[2] https://gemini.google.com/

- in speech prosody, tone, or even the absence of speech.

All these modalities provide important resources to understand human communication. In their day to day lives, people mix and fuse the cues from these different channels to engage in social activities. Based on such observation, the development of virutal agents should use not only verbal information provided by the user, but non-verbal content as well.

The complexity of fusing modalities led to lot of research in different domains. There is countless methods to extract information from multimodal input. The topic of this work in AC is limited to audio and visual modalities, more specifically to the processing of voices and faces.

The motivation behind this work was to provide open-source methods that leverage non-verbal cues to detect the emotional content (short-term) or the mood (long-term) in an interaction. The importance of such context is key for an external observer to understand the dynamics between individuals. For example people with Autism Spectrum Disorders (ASD) or other social difficulties would benefit from tools that give affective information and help them in their social interaction. Naive by nature, as they start from randomly initialised weights, DL models have been compared to a person with ASD that could learn from large amount of social examples [1].

Recent DL models require large datasets to reach satisfactory performance. Said performance depends on both the quality of the datasets (recording conditions and annotations) and their availability for training. While some systems show impressive results (e.g. ChatGPT on Natural Language Processing (NLP) tasks), they do not disclose the datasets used for training. Other systems fail to reach optimal performance due to lack of data in the downstream domain. They either use data augmentation to artificially increase the size of the dataset or spend resources creating a new dataset. When it comes to data collection, even more for audio-visual content, the ethical aspect is an important matter. In Europe, the AI Act [2] provides a legal framework to, among other things, regulate the use of personal data in AI systems.

1.2 Contributions

The original contributions of this thesis are listed below:

- IB, an extension of three datasets available for research is performed to provide new information. It follows an annotation protocol that specifies the turning role of participants in dyadic interactions (speaker or listener), as well as annotations on two non-verbal expressions (smiles and laughter). In addition to the expressions classes, the work conducted also contains subclasses for each one about their intensity. The extension is validated on multiple applications and the importance of providing intensity as a subclass is also discussed;

- Since the nature of laughter and smiles as distinct classes is not universal, we try to understand how a DL system would interpret them. To achieve this:
  - we use the voice and the area around the lips as input of LSN-TCN, a DL architecture, and set the goal to detect which expression appears;
  - we study the impact of fusing modalities on the recognition rate;
  - we analyse and discuss the distribution of each expression intensity during inference.
- Confident with the observations made on lips and voice, we extend our visual input on the whole face. We implement Social-MAE, a multimodal system based on attention and masked input to encode information about face and voice. The system is then used to perform downstream tasks, including smile and laugh detection and emotion recognition;
- Finally, we explore the possibility to conserve and disentangle affective information from existing DL systems without modifying their performance. This is achieved by connecting an auxiliary network and aiming to reconstruct the original input.

1.3 Organisation of the Dissertation

- Chapter 2 presents the theoretical notions about DL that are important for the reader to understand. It covers the cornerstone of most recent architectures that rely on Deep Neural Networks (DNNs).
- Chapter 3 presents how audio and visual data are processed in Machine Learning (ML) systems, either as single modality or joint together using fusion techniques.
- Chapter 4 introduces the reader to core concepts of Affective Computing (AC), including challenges, non-verbal communication and emotion models.
- Chapter 5 describes the datasets available for research in AC domain and their characteristics. It also presents the IB dataset, a annotation extension of three existing audio-visual collections of dyadic interactions.
- Chapter 6 presents the implementation and analysis of the detection of laugh and smile expressions by a multimodal system. It includes the discussion on the importance of modality fusion, the use of pre-trained models and how expression intensity influences the detection rate.
- Chapter 7 describes an architecture upgrade based on masking and multimodal attention. An analysis on downstream tasks is performed to study its encoding efficiency.
- Finally, Chapter 8 shows a proof of concept of a system that retains lost information from pre-trained models based on methods described in previous chapters. More

specifically, it discusses the affective content saved by an auxiliary branch plugged to the pre-trained model.

Part I

Theoretical Background

Chapter 2

Deep Learning

Contents

The last decade has seen the rapid development of DL, a branch of ML. DL groups all complex systems that are able to extract features from all sort of input based on DNN. These models are designed to perform tasks of varying degrees of complexity. DNN models vary in terms of applications and number of parameters. It originates from the design of Multi-Layer Perceptron (MLP), and has grown to complex models of billions of parameters for the most advanced (Computer Vision: ResNet18 [3]: 11.7M (2015),

Speech Recognition: Whisper-Large [4]: 1.55B (2022), NLP: DeepSeek-V3 [5]: 671B (2025)).

In this chapter, we present the basics of DL. First, we will explain the learning scheme that underlies most models (Section 2.1). Next, we will discuss the architecture of Convolutional Neural Network (CNN) (Section 2.2). Then, we describe the Transformers architecture, an important breakthrough in DL (Section 2.3). Finally, we present evaluation metrics, for domains like classification (Section 2.4) before giving a brief summary (Section 2.5).

2.1 Learning Scheme

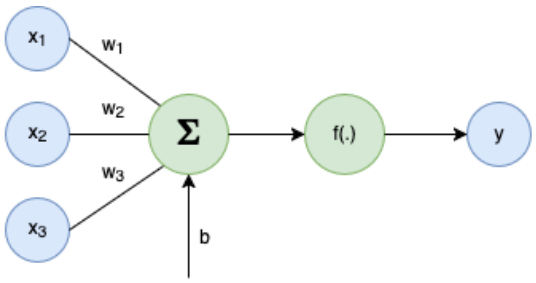
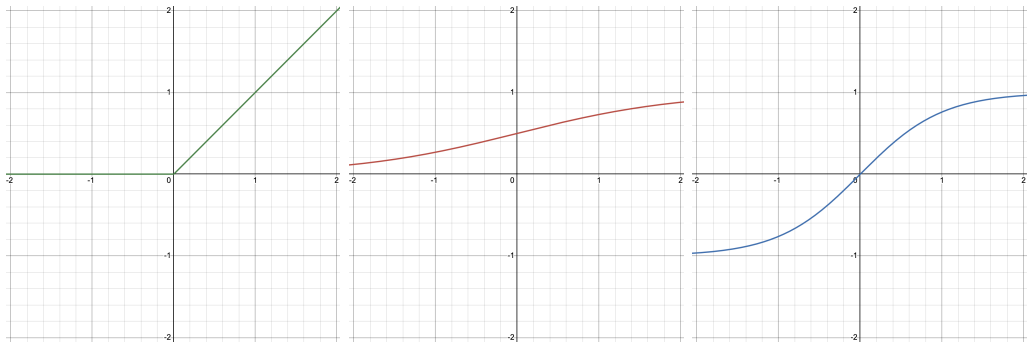


Figure 2.1. Representation of a perceptron. The node y is the result of the weighted sum of each node x_i passed through an activation function $f(.)$.

The main advance of DNN compared to previous ML systems is the combination of layered perceptrons to extract a complex representation of the data rather than relying on complex statistical modeling. A perceptron (Figure 2.1) is equivalent to a single value y, calculated as the weighted combination of x_i values from previous layers according to the equation 2.1.

$$y = f(W^T x) = f(\sum_{i=1}^{n} w_i x_i + b) \tag{2.1}$$

where f(.) is an activation function which introduces non-linearity, w_i the weight applied to the i^{th} value and b the bias that shifts the output of the activation function by adding a constant. Several activation functions can be considered depending on the application: Rectified Linear Unit (ReLU), Sigmoid, Tanh (Figure 2.2). The introduction of non-linearity is crucial for extracting abstract features from the output of the previous layer.



(a) ReLU: $\max(0, x)$ (b) Sigmoid: $\frac{1}{(1+e^{-x})}$ (c) Tanh: $\frac{(e^{x}-e^{-x})}{(e^{x}+e^{-x})}$

Figure 2.2. Activation functions. (a) ReLU is efficient but limits some neuron activations. (b) Sigmoid converts input into output ranging between 0 and 1. (c) Tanh is similar to Sigmoid with an output range between -1 and 1.

MLP is the most basic model for DNN methods and consists of several layers of perceptrons as depicted in Figure 2.3. It is defined with a set of parameters (W, b) as:

$$(W, b) = (W^{(1)}, b^{(1)}, W^{(2)}, b^{(2)}, \ldots, W^{(l)}, b^{(l)}, \ldots, W^{(L),b^{(L)}}) \tag{2.2}$$

where $W^{(l)}$ contains the weight matrix of layer l and $b^{(l)}$ the bias. The abstract multi-dimensional space between layers is referred to as the latent space, it contains a representation of the data where each dimension is a separate feature.

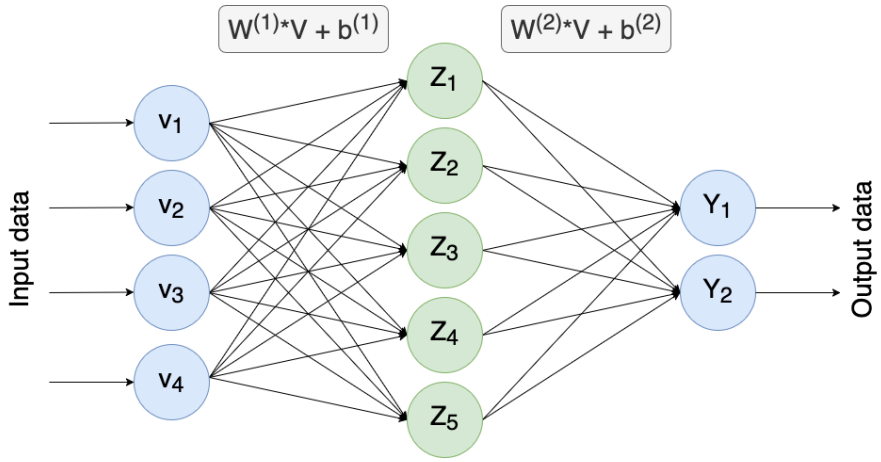


Figure 2.3. A multi-layer perceptron. Each circle is a perceptron, a value that embeds a representation of the input data. The more distant the perceptron layer, the more complex the representation it contains. In this example, the latent space corresponds to the green circles.

The weights and biases are adjusted during training by trial and error at every step. One step corresponds to the processing of a batch (a fixed quantity of input data) by the model. The amount of batch usually matches the dataset size divided by the batch size. An epoch refers to the relative period taken to process all batches once. The goal of adjusting weights is to reduce a cost function (or loss function) $\mathcal{L}$. The cost function depends on the task at hand: some are specific to classification (determine a discrete value in a set of possible targets), others to regression (predict a value in a continuous domain). We discuss cost functions later in Section 2.4.

The learning scheme is separated in three parts:

- **Training Phase:** The model learns by adjusting weights based on the content of the training subset.
- **Validation Phase:** At the end of each epoch, the model is evaluated on the validation subset.
- **Test Phase:** The final model is evaluated on the test subset to assess generalisation.

While a trained model can be used as is for inference in applications, especially if the results are satisfactory, it can also serve as backbone for transfer learning, which will be further discussed in Section 2.1.3.

2.1.1 Training

The training phase aims at tuning the parameters (W, b) so that the network outputs values as close as possible to the target for a given dataset. The objective is to learn generalisable features applicable to unseen data. Training is an iterative process where the network computes an output for a given input and updates its parameters to minimise the loss function $\mathcal{L}$, defined as:

$$\mathcal{L} = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(\hat{y}_i, y_i) \tag{2.3}$$

where $\hat{y}_i$ represents the model prediction and y_i the ground truth target. The choice of loss function depends on the task: e.g. cross-entropy for classification, mean squared error for regression, etc.

The loss function provides a feedback signal to adjust the model parameters. Updates are performed using gradient descent-based algorithms, such as Stochastic Gradient Descent (SGD) or Adam [6], which impact training speed, stability, and generalisation [7]. The

weights and biases are adjusted according to their gradient values:

$$w_{i,j}^{(l)} = w_{i,j}^{(l)} - \alpha \frac{\partial \mathcal{L}}{\partial w_{i,j}^{(l)}} \tag{2.4}$$

$$b_{i}^{(l)} = b_{i}^{(l)} - \alpha \frac{\partial \mathcal{L}}{\partial b_{i}^{(l)}} \tag{2.5}$$

where α is the learning rate, controlling step size in weight updates. A poor learning rate choice may lead to slow *convergence* (if too low) or *divergence* (if too high), affecting model performance. To efficiently compute gradients, models use *backpropagation*. It first calculates gradients for the last layer and propagates them backward using the chain rule:

$$\frac{\partial \mathcal{L}}{\partial w_{i,j}^{(l)}} = \frac{\partial \mathcal{L}}{\partial z_{j}^{(l)}} \cdot \frac{\partial z_{j}^{(l)}}{\partial w_{i,j}^{(l)}} \tag{2.6}$$

where $z_j^{(l)}$ is the output of neuron j. The computed gradients are then applied to update parameters using Equations 2.4 and 2.5.

Training consists of:

- A **forward pass**, where input data propagates through the network to produce an output and compute the loss.
- A **backward pass**, where gradients are computed and weights updated accordingly.

Models typically require multiple epochs to converge. Training can be done using either batch gradient descent that processes the entire dataset at once, but is computationally expensive, mini-batch gradient descent that updates weights using smaller batches, balancing efficiency and stability or SGD that updates weights per sample, introducing noise but improving generalisation.

A major challenge is *overfitting*, where the model memorises the training data instead of generalising. Regularisation techniques like weight decay, dropout, and data augmentation mitigate this. The opposite is *underfitting* which occurs when the model lacks sufficient capacity to learn patterns and can be addressed by increasing model complexity or extending training duration.

Determining when to stop training is crucial. *Early stopping* monitors validation loss and halts training when no further improvement is observed, preventing overfitting. Learning rate schedules also help refine convergence.

2.1.2 Validation and Test

The validation and test phases assess the model performance on unseen data, but they serve distinct purposes:

- **Validation Set:** Used during training to fine-tune hyperparameters (e.g., learning rate, dropout, batch size). It helps monitor overfitting and guides optimisation strategies.
- **Test Set:** Evaluated only once after training is complete to assess the model's final generalisation ability.

Validation and Overfitting Detection Validation is performed at the end of each epoch using a separate subset of data. The model's loss and other evaluation metrics are computed, and the best-performing parameters are typically stored as checkpoints. As mentioned, one key purpose of validation is to detect *overfitting*, where the model performs well on training data but generalises poorly. *Overfitting* is often identified when the training loss decreases while the validation loss increases.

While deep learning models typically use a single validation set, K-fold cross-validation is useful for smaller datasets. It involves splitting the dataset into K subsets, training on $K-1$ folds, and validating on the remaining fold, repeating this K times to ensure robustness.

Test Set and Fair Evaluation The test set serves as a final benchmark for model performance and must remain completely separate from training and validation to prevent data leakage. Leakage can occur if test data influences training, leading to misleadingly high performance.

Test set evaluation is crucial for model comparison in research, and statistical significance should be considered when comparing results. Techniques like **t-tests**, further discussed in Section 2.4, ensure that improvements are not due to random chance. Additionally, a well-trained model should generalise to new, unseen data. Robustness can be tested by evaluating performance across different datasets (domain adaptation) or by introducing small perturbations (adversarial attacks, noise) to assess stability.

2.1.3 Transfer Learning

Transfer learning is a technique that improves the learning process by leveraging knowledge acquired from pre-trained models rather than training from scratch. This approach significantly enhances efficiency, reduces the need for large labeled datasets, and accelerates training convergence.

Instead of initialising model weights randomly, transfer learning repurposes models that have been trained on large-scale datasets and fine-tunes them to perform well on new tasks with different but related data distributions.

Types of Transfer Learning

There are different strategies for transfer learning, depending on how much of the pre-trained model is reused:

- **Feature Extraction (Frozen Weights):** The pre-trained model's layers act as a fixed feature extractor, and only a new classifier or task-specific layers are trained on top of the extracted representations. This is common when the new dataset is small.
- **Fine-tuning (Full Model Adaptation):** Some or all layers of the pre-trained model are updated alongside the newly added task-specific layers. Fine-tuning requires more computational resources but allows the model to adapt better to the new task.
- **Hybrid Approach:** The lower layers (which learn generic features) remain frozen while the higher layers (task-specific features) are fine-tuned. This balances efficiency and task adaptation.

Few-shot and Zero-shot Learning

Modern transfer learning approaches, like few-shot and zero shot learning, aim to minimise the reliance on labeled data. In Few-shot Learning, the model is fine-tuned using only a small number of labeled examples per class. This is beneficial when data collection is expensive or limited. Zero-shot Learning refers to the model generalising to new tasks without explicit retraining, relying on high-level semantic embeddings learned from vast datasets.

Benefits and Challenges

Transfer learning offers multiple advantages. While training from scratch requires extensive data and computational resources, fine-tuning pre-trained models accelerates convergence. It also improves generalisation by leveraging knowledge from large datasets to help models perform better on smaller, domain-specific datasets. It also enables high performance even with scarce labeled data.

However fine-tuning includes challenges. If the new dataset is too different from the pre-trained model's dataset, performance may degrade. This is known as as domain shift. Fine-tuning can also overwrite learned knowledge and can harm generalisation. While pre-trained models save training time, fine-tuning large models still requires significant resources.

Figure 2.4 illustrates how a pre-trained image recognition model (e.g., ResNet) can be adapted to a new task, such as classifying ducks and dogs.

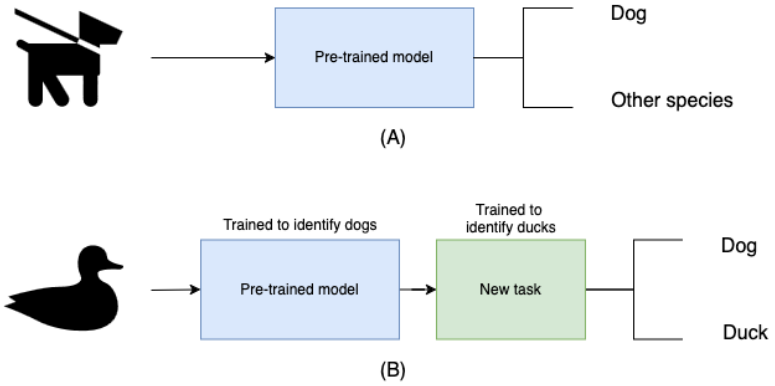


Figure 2.4. Illustration of transfer learning in neural networks for image classification. (A) Pre-trained Model: A model trained on a dataset (e.g., to classify dogs vs. other species). (B) Transfer Learning: A pre-trained model is adapted to a new task (e.g., identifying ducks).

2.2 Convolutional Neural Network

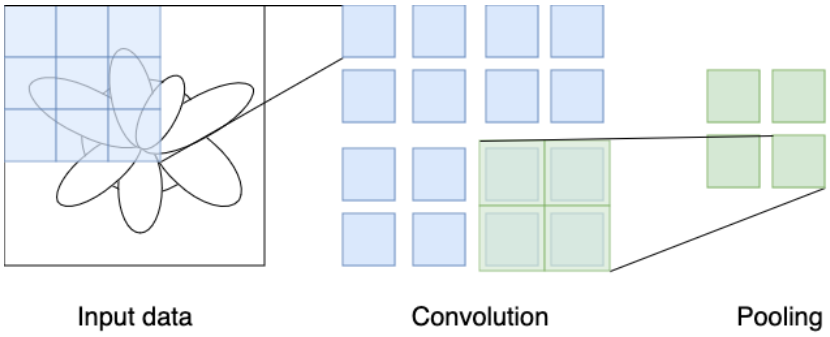


Figure 2.5. Illustration of a CNN processing pipeline. The left section represents the input data, where a filter (blue grid) slides over the raw image to extract local features. The convolution operation (middle section) transforms the input into feature maps by applying a kernel that captures spatial patterns. Pooling (right section) down-samples the feature map by aggregating values from neighboring regions, reducing dimensionality while keeping essential information.

2.2.1 Definition

A CNN [8] is an extension of MLP specifically designed to process structured signals such as images, audio, and video. Unlike MLPs, CNNs leverage spatial hierarchies and local connectivity patterns to efficiently learn meaningful representations.

Three key principles define the effectiveness of CNNs:

- **Locality:** Neighboring values in an input (e.g., pixels in an image or samples in an audio signal) exhibit strong correlations, and CNNs exploit this spatial dependency.
- **Stationarity:** The same patterns frequently appear in different parts of the input, allowing filters to learn shared features across the entire dataset.
- **Compositionality:** Higher-level representations are constructed by hierarchically combining low-level features, enabling abstraction and complex pattern recognition.

A typical CNN consists of two fundamental types of layers: convolutional layers and pooling layers, as depicted in Figure 2.5. Convolutional layers apply learnable filters to detect spatial and temporal features, while pooling layers reduce dimensionality by summarising information in local regions. These layers are often stacked in deep architectures, allowing models to progressively extract more complex patterns.

CNNs automatically learn hierarchical feature representations, making them particularly suited for computer vision. The next sections will detail the role of convolutional and pooling layers, along with the training strategies used to optimise CNN-based models.

2.2.2 Convolution Layer

The convolution layer extracts features from an input by applying a set of filters (or kernels) that slide across the signal. Each filter consists of learnable weights, which interact with input values to emphasise important patterns. For an input image I of dimension $w \times h \times d$, where w and h are the width and height, and d represents the number of channels (e.g., $d = 3$ for RGB images), a convolution operation applies d learnable kernels K of size $K_h \times K_w$. The result is a feature map computed as follows:

$$z_{ij} = \sum_{a=\frac{-K_h}{2}}^{\frac{K_h}{2}} \sum_{b=\frac{-K_w}{2}}^{\frac{K_w}{2}} (W_{ab} \times x_{(i+a)(j+b)}) \tag{2.7}$$

$$z = W * x \tag{2.8}$$

where W represents the filter weights and x the input values.

To control the spatial dimensions of the output, padding is used to adjust the receptive field. There is common types of padding. The first is valid padding (or no padding) where no extra pixels are added resulting in an output size smaller than the input. The second is same value padding where extra pixels (typically zeros) are added to maintain the same spatial dimensions between input and output. Padding helps preserve border information, ensuring that feature extraction applies uniformly across the image.

The way filters slide across the input is describe by the stride. Stride defines the step size at which the filter moves across the input. A stride of 1 results in maximum coverage, whereas a stride greater than 1 skips positions, reducing the feature map's spatial size.

Higher strides introduce downsampling effects, reducing computation but potentially losing fine-grained details.

For multi-channel convolutions (as stated above RGB images have 3 channels) CNNs apply filters with the same depth as the number of channels leading to aggregated responses that create a more complex representation. The output of a multi-channel convolution is computed as the sum of filtered responses from all channels.

The convolution operation enables hierarchical feature learning. Early layers detect low-level features like edges and textures, while deeper layers recognise complex patterns such as object parts or high-level abstractions. The ability of CNNs to automatically learn these representations makes them powerful for computer vision [9, 10].

2.2.3 Pooling Layer

The pooling layer is a non-trainable layer designed to reduce the dimensionality of feature maps while preserving essential information. Pooling is typically applied independently to each channel of a feature map. The two most common types are Max Pooling and Average Pooling. With the Max Pooling method the model selects the maximum value within a given window, emphasising dominant features and edges. Average Pooling computes the average value within a given window, preserving more contextual information.

In addition to standard pooling operations, Global Average Pooling (GAP) is widely used in modern architectures. Instead of applying a fixed-size kernel, GAP computes the average of all values across the entire feature map, effectively reducing each channel to a single value. This technique is commonly used before fully connected layers in classification models, enabling parameter-efficient architectures [11, 12].

Pooling serves several key functions:

- **Dimensionality Reduction:** Decreases the spatial size of feature maps, leading to fewer parameters and lower computational cost.
- **Translation Invariance:** Helps models focus on patterns rather than specific pixel locations.
- **Overfitting Reduction:** Downsampling forces the network to learn more robust features by discarding non-essential details.

Despite its benefits, pooling removes fine-grained spatial information, which may be undesirable in tasks requiring precise localisation, such as object detection or segmentation. To address this, some architectures use attention mechanisms as alternatives to pooling, as discussed in Section 2.3.4.

2.2.4 Fully Connected Layers and Output

After extracting hierarchical features through convolution and pooling layers, a CNN typically ends with one or more fully connected (FC) layers to produce a final output. These layers function similarly to those in MLP networks. The role of FC layers is to transform the high-dimensional feature maps into a structured representation suitable for classification, regression, or other downstream tasks.

For classification tasks, the final FC layer typically outputs a vector representing class probabilities. When handling multi-class classification, a *softmax* activation function is applied:

$$\hat{y}_i = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}}, \tag{2.9}$$

where $\hat{y}_i$ is the probability of class i, and z represents the raw scores produced by the FC. The sum of all $\hat{y}_i$ equals to 1. For binary classification, the *sigmoid* activation function (defined in Section 2.1) is often used instead.

Beyond classification, CNNs can be adapted for regression tasks by replacing the softmax layer with a linear activation function. In object detection and segmentation, FC layers may be replaced with more specialised architectures such as bounding box regression heads or segmentation masks [13].

2.2.5 CNN Applications and Challenges

CNNs have demonstrated state-of-the-art performance across various domains due to their ability to automatically learn hierarchical representations [14, 15]. Despite their widespread success, CNNs face several challenges. One key limitation is their difficulty in capturing long-range dependencies, as they primarily focus on local features. Another challenge is the computational cost associated with deep CNNs, which require substantial processing power and memory, making them expensive for real-time applications. Additionally, CNNs heavily depend on large labeled datasets for training, limiting their effectiveness in low-data scenarios. To mitigate this, techniques such as transfer learning and few-shot learning are often employed, as presented above. Finally, CNNs are vulnerable to adversarial attacks, where small, imperceptible perturbations in input images can lead to incorrect predictions, posing significant security risks in applications like facial recognition and autonomous driving.

2.3 Transformer

Transformers are a groundbreaking innovation in the field of AI, introduced in the paper "Attention Is All You Need" [16]. Since then, they have become a foundational archi-

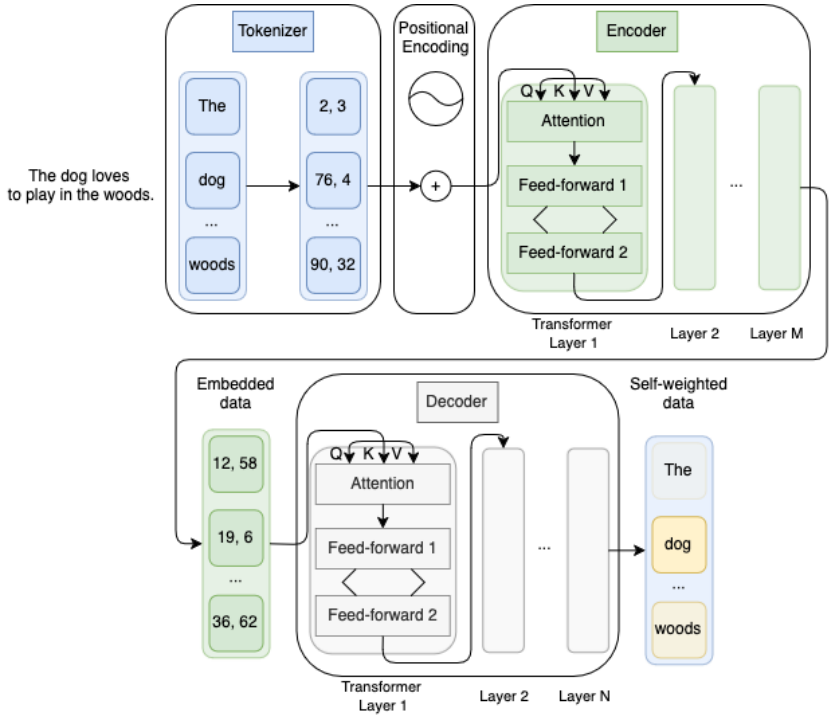


Figure 2.6. Overview of the Transformer architecture, illustrating the tokenization process, positional encoding, and the flow of data through the encoder and decoder. The input text is first converted into numerical tokens with additional positional encodings, and processed through self-attention and feed-forward layers. The decoder takes the encoded latent representations and applies self-attention to generate contextualised outputs.

tecture, playing a crucial role in the development of state-of-the-art models for various natural language processing tasks, including machine translation and sentiment analysis, outperforming CNN-only models.

This section examines the fundamental principles of the Transformer architecture, covering key components: the Tokenizer, the Encoder-Decoder, the Attention Mechanism, the Feed-Forward Networks, and the Positional Encoding. A simplified diagram of the Transformer framework is presented in Figure 2.6.

2.3.1 Tokens

Input sequences are divided into fixed-size sets of tokens, which serve as the fundamental units for the Transformer. The model processes input by splitting it into smaller parts and converting them into embeddings of a fixed but higher-dimensional representation. This tokenization, combined with the attention mechanism, enables the model to handle

sequences of varying lengths efficiently through matrix multiplications. This advancement significantly enhances the ability to generate or translate sequences such as images, text, and audio. For instance, in NLP, a text tokenizer assigns an index to each word based on a predefined dictionary [17, 18]. In the case of Vision Transformer (ViT) [19], CNNs act as tokenizers, converting image patches into meaningful representations.

2.3.2 Positional Encoding

Positional Encoding is a mechanism designed to incorporate information about the order of elements within an input sequence. Since Transformers do not inherently capture positional dependencies, positional encoding is introduced to help the model distinguish between elements based on their position. It consists of a set of parameters that are added element-wise to the input tokens, ensuring that positional relationships are retained. These parameters can be either engineered values, such as a combination of sine and cosine functions [20], or learnable parameters which lead to similar performance [16].

Our positional encoding operates as an additive vector P that remains constant across all input sequences. For a simple example, consider the i^{th} input sequence $x_i = [x_{i1}, x_{i2}, \ldots, x_{in}]$ in a batch, with the corresponding positional encoding $P = [p_1, p_2, \ldots, p_n]$. The final representation s_i before passing through the first layer is computed as:

$$s_i = x_{ij} + p_j \tag{2.10}$$

This ensures that each token retains positional information while being processed by the Transformer model.

2.3.3 Encoder and Decoder

The Transformer architecture consists of two main components: an encoder that processes and encodes the input information into a latent representation, and a decoder that reconstructs or generates an output from this latent space. The encoder-decoder approach can be categorised as an Auto-Encoder (AE) when the output objective is derived exclusively from the input data, in any form. For instance, an autoencoder can be used for denoising tasks, where noise is artificially added during data augmentation and the model learns to reconstruct the clean signal.

Beyond generative tasks, the encoder can also be fine-tuned as a feature extractor for classification or regression problems. Encoding information is fundamental in ML, as model performance is directly influenced by its ability to capture patterns in the training data. The final output of the encoder, the latent space, serves as an intermediate representation of the input data. In supervised learning tasks such as regression and classification, this latent representation is typically passed through decision layers to

produce class predictions or continuous values. In contrast, generative models leverage the decoder to transform the latent space into meaningful outputs.

The Transformer encoder consists of multiple stacked layers, where each layer comprises a self-attention mechanism and a feed-forward network, detailed in Sections 2.3.4 and 2.3.6, respectively. The decoder, as introduced in [16], differs from the encoder. First, it employs masked attention to enforce causality, ensuring that predictions at a given position i do not depend on future tokens $j > i$. Second, it introduces an additional cross-attention mechanism, which allows the decoder to attend to both the previously generated sequence and the encoded input representation.

Transformers follow a residual learning paradigm, meaning that each layer refines the representation by adding newly extracted information to its input, improving gradient flow and stabilising training.

2.3.4 Attention

The attention mechanism is the core component of the Transformer. It enables the model to perform what is known as 'self-attention', a dynamic process through which it evaluates and assigns different levels of importance to various elements within an input sequence. Each element can be a word in a sentence, a pixel in an image, or any other discrete unit in the data.

Self-attention allows the Transformer to capture contextual information, as it considers how each element relates to all other elements in the sequence. This comparison generates a set of attention scores that reflect the importance of each element with respect to the others. These scores are then passed through a softmax function that transforms them into probability distributions that sum to 1. These distributions determine how much focus each element should place on other elements. More specifically, the Attention mechanism is defined as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2.11}$$

with Q the query vectors, K the key vectors, V the value vectors and d_k the vector dimension. In the example sentence 'The dog loves to play in the woods', self-attention allows the word 'loves' to focus on the word 'dog' more than 'woods' (Figure 2.6). We can see those vectors as a user with a query that looks for a match in all keys, returning a value for its request.

QK^T, the dot product between queries and keys, gives a measure of the similarity of each pair $q_i k_j$. If d_k increases to large values, the dot product magnitude pushes the softmax function to reach extremely small gradient. To balance this effect and avoid exploding gradients, the dot product is scaled by a factor $\frac{1}{\sqrt{d_k}}$. The matrix multiplication with the values gives a weighted representation based solely on available data.

2.3.5 Multi-head attention

The *multi-head attention* mechanism is a crucial component of the Transformer architecture, enhancing the model's ability to capture complex dependencies within an input sequence. It extends the concept of *self-attention*, where every token in the sequence attends to every other token, by introducing multiple parallel attention mechanisms, referred to as heads (Figure 2.7 - left).

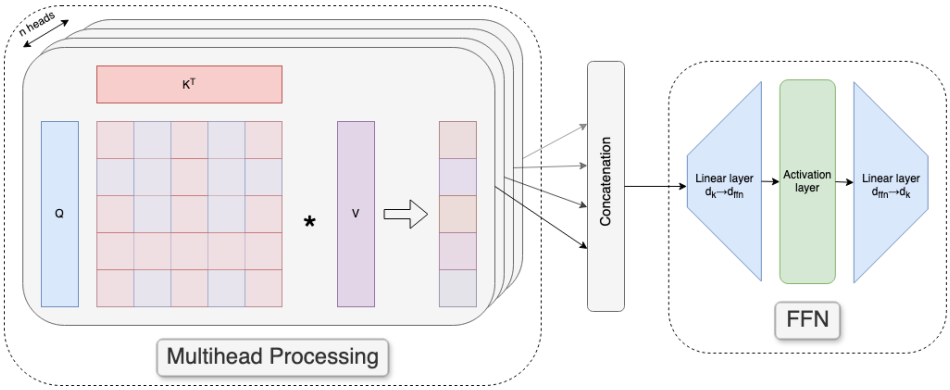


Figure 2.7. Illustration of the multi-head attention mechanism followed by a feed-forward network in a Transformer model. On the left, the attention mechanism computes the attention scores. This operation is repeated across multiple heads, each extracting different relationships within the input sequence. The outputs from all attention heads are concatenated and fed to feed-forward network (right). The feed-forward network consists of two FC layers with an activation function in between, allowing for additional feature transformation before propagating to the next layer in the model.

A single self-attention layer may struggle to capture different aspects of the input data, such as syntax and semantics in NLP or spatial and temporal features in vision tasks. To address this, the Transformer applies multiple self-attention operations in parallel. Each attention head learns a distinct representation of relationships within the sequence, enabling the model to process information more effectively.

Mathematically, given an input sequence X, the Transformer first projects it into three different spaces to form Q, K, and V matrices:

$$Q = XW_Q, \quad K = XW_K, \quad V = XW_V \tag{2.12}$$

where W_Q, W_K, W_V are learnable projection matrices that reduce the input dimensionality. This reduction is analogous to 1D convolutional layers. Each head independently performs scaled dot-product attention as in Equation 2.11.

Once the attention computation is completed across all h heads, the resulting outputs are concatenated and projected back to the original dimension using a final weight matrix:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \text{head}_2, ..., \text{head}_h)W_O \tag{2.13}$$

where W_O is a trainable weight matrix that integrates the different attention perspectives.

The introduction of multiple heads enables the Transformer to:

- Capture various types of dependencies simultaneously, such as short-term and long-term relationships.
- Learn different representations of meaning, such as word order and contextual importance.
- Improve generalisation by allowing multiple attention patterns rather than relying on a single perspective.

The combination of multiple attention heads diversify the extraction process of the model, contributing to its state-of-the-art performance across various domains, including text, images, and speech processing.

2.3.6 Feed-Forward Networks

Beyond the attention mechanism, the Feed-Forward Network (FFN) plays a crucial role in Transformers by refining and transforming latent representations before passing them to the next layer (Figure 2.7 - right). Each latent embedding, after undergoing multi-head attention, is processed independently by an FFN. Unlike CNNs, which extract spatial patterns, which process sequences step by step, the FFN applies the same transformation to each embedding without considering positional dependencies. This ensures that information flows effectively while maintaining computational efficiency.

The FFN consists of two linear transformations with a non-linear activation function in between:

$$\text{FFN}(x) = \text{activation}(x \times W_1 + b_1) \times W_2 + b_2 \tag{2.14}$$

where W_1, W_2 are weight matrices, b_1, b_2 are biases, and ReLU is typically used as the activation function. The first linear transformation expands the representation into a higher-dimensional space, while the second one projects it back to its original size. This dimensional expansion, often with a factor of 4, allows the network to learn richer representations.

The inclusion of an FFN after attention serves multiple purposes. It enhances feature extraction by introducing non-linearity, allowing for more complex transformations beyond simple weighted sums from attention. It processes each token independently, ensuring that the model can learn transformations without forcing unnecessary sequen-

tial dependencies. The projection into a higher-dimensional space enhances the learning capabilities of the model, helping it capture intricate relationships within the data.

2.3.7 Comparison with CNNs

While CNNs process local spatial features using convolution kernels, Transformers leverage FFNs to transform embeddings after attention. The advantage of this approach is that the FFN learns a more general transformation rather than focusing on local patterns. However, this comes with a computational cost, as each token must be processed separately in a high-dimensional space.

2.4 Evaluation metrics

This section introduces several evaluation metrics. The choice of metric depends on the problem type, whether it is classification, regression, speech recognition, or another deep-learning application. Below is a non-exhaustive list of commonly used metrics, categorised by their usage.

2.4.1 Loss Functions

Loss functions output a value that the model attempts to optimise during training. For classification tasks, the most commonly used loss function is the cross-entropy loss, which measures the difference between the predicted probability distribution and the true class labels. Common regression losses are Mean Squared Error (MSE) and Mean Absolute Error (MAE), but custom loss functions are also often used.

Cross-entropy Cross-entropy loss is defined as:

$$\mathcal{L} = -\sum_{i=1}^{n} y_i \log(\hat{y}_i) \tag{2.15}$$

where y_i is the true label (1 for the correct class, 0 otherwise) and $\hat{y}_i$ is the predicted probability for that class.

For binary classification, it simplifies to:

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] \tag{2.16}$$

For multi-class classification, where the output is a probability distribution over C classes using a softmax function, cross-entropy is:

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{c=1}^{C} y_{ic}\log(\hat{y}_{ic}) \tag{2.17}$$

Mean Squared Error (MSE) MSE calculates the squared differences between predicted and actual values (Lower values indicate better model performance):

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2.18}$$

Benefits: MSE strongly penalises large errors due to the squared term, making it useful for applications where large deviations should be minimized.

Drawbacks: The squared nature of MSE gives disproportionate weight to large errors, which can make the model more sensitive to outliers. Additionally, since MSE is in squared units of the target variable, it may be less interpretable compared to absolute error metrics.

Mean Absolute Error (MAE) MAE computes the absolute differences between predicted and actual values:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{2.19}$$

Benefits: Unlike MSE, MAE treats all errors equally, making it more robust to outliers. It is also easier to interpret because it reflects the average deviation in the same unit as the target variable.

Drawbacks: MAE does not penalise large errors as heavily as MSE, which might be undesirable in scenarios where minimizing large deviations is crucial.

2.4.2 Performance Metrics

Accuracy measures the overall correctness of the model:

$$\text{Accuracy} = \frac{\sum_{i=1}^{N} TP_i}{\sum_{i=1}^{N}(TP_i + \sum_{j\neq i} FP_{i,j})} \tag{2.20}$$

where the numerator represents the total correctly classified samples across all classes, and the denominator is the total number of samples. While accuracy is useful for balanced datasets, it can be misleading in the presence of class imbalance. In such cases, class-specific metrics such as Precision, Recall, and F1-score provide a more informative evaluation:

$$P = \frac{TP}{TP + FP} \tag{2.21}$$

$$R = \frac{TP}{TP + FN} \tag{2.22}$$

$$\text{F1-score} = 2 \times \frac{P \times R}{P + R} = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{2.23}$$

where True Positive (TP) indicates correctly classified positive samples, False Positive (FP) represents incorrect positive predictions, and False Negative (FN) denotes actual positives that were misclassified as negatives. Precision measures how often the model is correct when predicting a positive class. Recall (also called Sensitivity) quantifies how well the model captures actual positive instances. F1-score is the harmonic mean of Precision and Recall, balancing both aspects. These metrics can be further divided into "micro-averaged", which aggregate TP, FP, and FN across all classes before computing the final score, and "macro-averaged", which compute individual scores per class and then take their unweighted average.

A confusion matrix provides a structured way to evaluate classification performance by summarizing correct and incorrect predictions across multiple classes. For a multi-class classification task with N classes, the confusion matrix is represented as follows:

	Predicted Class 1	Predicted Class 2	...	Predicted Class N
Actual Class 1	TP_1	$FP_{1,2}$	...	$FP_{1,N}$
Actual Class 2	$FP_{2,1}$	TP_2	...	$FP_{2,N}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
Actual Class N	$FP_{N,1}$	$FP_{N,2}$	...	TP_N

Table 2.1. Confusion Matrix for Multi-Class Classification

Each diagonal element TP_i represents the number of correctly classified instances for class i. Off-diagonal elements $FP_{i,j}$ represent misclassifications, where samples belonging to class i were incorrectly classified as class j.

2.4.3 Statistical Significance and Fair Evaluation

Beyond model performance metrics, it is essential to ensure that reported performance are statistically significant. One common method is hypothesis testing using the p-value.

The p-value represents the probability of observing the obtained results if the null hypothesis was true. The null hypothesis is a statistical hypothesis that states that no statistical significance exists in a set of given observations. It assesses the credibility of a hypothesis by using sample data. The p-value provides a level of significance at which the null hypothesis would be rejected. A lower p-value (typically $p < 0.05$) indicates stronger evidence against the null hypothesis, suggesting that performance differences are statistically significant.

A good model should generalise well to unseen data. Generalisation can be tested by evaluating domain shifts, where the model is tested on data from a different distribution or adversarial robustness, assessing how small input perturbations affect predictions. Another test is the out-of-distribution (OOD) detection, identifying whether the model confidently classifies unknown inputs.

2.5 In Brief

Summary for Chapter 2

- This chapter provided an overview of deep learning, highlighting its fundamental concepts, architectures, and evaluation metrics. We introduce the learning scheme and transfer learning as a technique that leverages pre-trained models to improve performance on new tasks.
- Convolutional Neural Networks (CNNs) were introduced as a key architecture for structured data processing. Transformers were presented as an alternative architecture, excelling in capturing long-range dependencies through self-attention mechanisms.
- The chapter also covered evaluation metrics for classification tasks, like F1-score, and cross-entropy loss, and regression problems, like Mean Squared Error (MSE) and Mean Absolute Error (MAE). The importance of statistical significance tests and considerations for fair model evaluation were discussed.

Perspectives for Chapter 2

- DL presents several ethical and practical challenges. High power consumption raises environmental concerns, while data privacy and security risks emerge from models trained on vast user data.
- Additionally, DL models often lack interpretability, making it difficult to understand their decision-making process. Explainable AI (XAI) research aims to improve transparency through feature attribution and interpretability methods.

Chapter 3

Audio-Visual Processing

Contents

This chapter is based on the following publications :

- Hugo Bohy, Ahmad Hammoudeh, Antoine Maiorca, Stéphane Dupont, and Thierry Dutoit. "Analysis of Co-Laughter Gesture Relationship on RGB Videos in Dyadic Conversation Context". In *Proceedings of the Workshop on Smiling and Laughter across Contexts and the Life-span within the 13th Language Resources and Evaluation Conference*, 2022.
- Antoine Maiorca, Hugo Bohy, Youngwoo Yoon, and Thierry Dutoit. "Objective Evaluation Metric for Motion Generative Models: Validating Fréchet Motion Distance on Foot Skating and Over-smoothing Artifacts". In *Proceedings of the 16th ACM SIGGRAPH Conference on Motion, Interaction and Games*, 2023.

Our work focuses on processing images, videos and audio signals and the information resulting from their fusion. In this chapter we will first outline several representations for audio (Section 3.1) and visual data (Section 3.2). Then we will discuss the various methods to fuse modalities together (Section 3.3). We conclude the chapter with a summary and perspectives (Section 3.4).

3.1 Audio representation

Audio data can be represented in various forms, ranging from raw waveforms to structured feature embeddings extracted through deep learning models. The choice of representation depends on the computational efficiency required and the level of abstraction needed for downstream tasks.

3.1.1 Waveform Representation

Raw audio is typically stored as a time-series waveform, where each sample represents the amplitude of a sound signal at a given time step. The sampling rate determines the resolution of the signal, with common values including 16 kHz for speech and 44.1 kHz for music.

Although raw waveforms preserve all information from the original signal, they are high-dimensional and computationally expensive to process. Directly feeding raw audio into deep learning models requires important computational resources and large amounts of training data. As a result, most audio processing pipelines leverage feature extraction methods to transform waveforms into more compact and informative representations.

3.1.2 Handcrafted Audio Features

To reduce dimensionality while preserving relevant information, various handcrafted audio features have been engineered. These features extract spectral and temporal characteristics that are useful for speech and audio recognition tasks.

MFCCs are one of the most widely used audio features, particularly in speech processing. They simulate the human auditory system by applying a Mel-scale filterbank to the power spectrum of the signal, followed by a discrete cosine transform (DCT) to decorrelate frequency components. The resulting coefficients capture the timbre and phonetic characteristics of the audio. The computation of MFCCs involves:

- Applying the Short-Time Fourier Transform (STFT) to extract frequency information over time.
- Passing the spectrum through Mel-scale filterbanks (a handcrafted collection of filters), emphasising perceptually important frequencies.

- Applying the logarithm to approximate human loudness perception.
- Performing a DCT transformation to obtain decorrelated coefficients.

A spectrogram is a time-frequency representation of an audio signal that visualises how frequency content evolves over time. Spectrograms can be further processed into Mel-Spectrograms, where frequency bins are mapped to the Mel scale, mimicking human auditory perception. These representations serve as powerful inputs for CNNs and other deep-learning models.

While other handcrafted features exist, their interest is limited in our work and are therefore not mentioned. Engineered features have the main advantage of being more easily interpretable than learned features from DL feature extractors.

3.1.3 Learned Audio Representations

Recent advances in deep learning have enabled models to learn feature representations directly from raw waveforms, avoiding the need for handcrafted features. These deep features capture higher-level abstractions that are useful for complex tasks like speech recognition.

Several self-supervised learning architectures extract deep audio embeddings from raw waveforms:

- Wav2Vec 2.0 [21]: Uses a Transformer-based model to learn speech representations from unlabelled audio, improving performance on speech recognition tasks.
- HuBERT [22]: Uses masked prediction training to learn hierarchical speech representations. It generates pseudo-labels for each unmasked segment of the speech and then tries to predict the pseudo-label of the masked segments.
- Audio2Vec [23]: Builds up a missing audio section based on surrounding sections.

These models provide robust embeddings that outperform traditional features in various speech and audio understanding tasks.

Deep-learning-based approaches learn feature transformations directly from raw waveforms:

- CNN-Based Feature Extraction: Some models apply 1D or 2D CNNs to raw waveforms or spectrograms to extract localised spectral patterns.
- Transformer-Based Representations: Inspired by ViT [19] (Section 3.2.3), some architectures process spectrogram patches as input tokens, leveraging self-attention to model long-range dependencies.

These approaches eliminate the need for manual feature engineering. They also achieve state-of-the-art performance in speech recognition, sound event detection, and music classification.

3.2 Visual representation

Visual data, including both images and videos, presents unique challenges due to its high dimensionality and complex spatial-temporal dependencies. While raw pixel values can be directly processed, most deep-learning approaches extract meaningful features through engineered or learned representations.

3.2.1 Raw Image and Video Representations

Images are represented as 3D tensors, where each pixel is associated with a value across multiple colour channels:

$$I(x, y, c) \in \mathbb{R}^{H \times W \times C} \tag{3.1}$$

where H and W denote the height and width of the image, and C represents the number of channels (e.g., RGB images have $C = 3$ channels). Videos introduce an additional temporal dimension, effectively forming a 4D tensor:

$$V(t, x, y, c) \in \mathbb{R}^{T \times H \times W \times C} \tag{3.2}$$

where T represents the number of frames per second (fps). Due to this extra dimension, video processing is computationally expensive and often requires efficient feature extraction techniques. Processing raw visual data directly also needs significant memory and computational power. As a result, handcrafted features and deep-learning-based embeddings are commonly used to extract relevant information.

3.2.2 Traditional Feature Extraction

Before deep learning became dominant, various handcrafted features were designed to capture specific aspects of visual data. These features reduce dimensionality while preserving meaningful representations.

Optical flow measures motion between consecutive video frames by estimating pixel displacement. It is commonly used in gesture recognition, action recognition, and motion tracking.

Histogram of Oriented Gradients (HOG) is a feature descriptor that captures local object shape and appearance by computing the distribution of edge orientations. It has been extensively used in early object detection algorithms, including pedestrian detection.

3.2.3 Deep Feature Extraction

Modern deep-learning approaches extract high-level visual representations through hierarchical feature learning. These methods replace handcrafted features with learned embeddings that are optimised for specific tasks.

CNNs have revolutionised visual feature extraction by automatically learning hierarchical representations. Popular architectures include:

- ResNet [3]: Introduces skip connections to improve deep network training.
- EfficientNet [24]: Optimises model scaling for better efficiency.

Many modern vision tasks leverage pretrained models to extract feature representations, reducing computational costs and training time. Some commonly used backbones include:

- ViT [19]: Trained to classify images on a large-scale dataset to capture global relationships. It processes images by breaking them into small patches, projecting each patch into a high-dimensional token, and then using a self-attention mechanism to understand the relationships between these tokens.
- DINO [25]: A self-supervised ViT model that learns high-quality embeddings without labeled data.
- CLIP [26]: Jointly trained on image-text pairs to enable zero-shot learning. One modality is used as the Query the self-attention mechanism and the other is used as the Key and the Value.

Deep-learning-based approaches, particularly Transformer models like ViTs, have surpassed traditional CNNs in certain computer vision applications, highlighting the ongoing shift towards more flexible architectures.

3.3 Fusion levels

This section presents the different fusion methods. In this context, fusion refers to all approaches that mix different modality together (concatenation, addition, multiplication, ...). It can either be early fusion, late fusion and mid fusion (Figure 3.1).

When combining modalities such as speech and video, maintaining temporal alignment is crucial. For example speech and lip movements must be synchronised to ensure meaningful correlations. Misalignment can negatively impact performance, particularly in tasks like speech-driven facial animation or audiovisual emotion recognition.

Audio and visual data often carry different levels of importance for a given task. In some cases, one modality may dominate the learning process, leading to an imbalance in feature utilisation. Adaptive weighting techniques or attention mechanisms can help dynamically adjust the contribution of each modality during training.

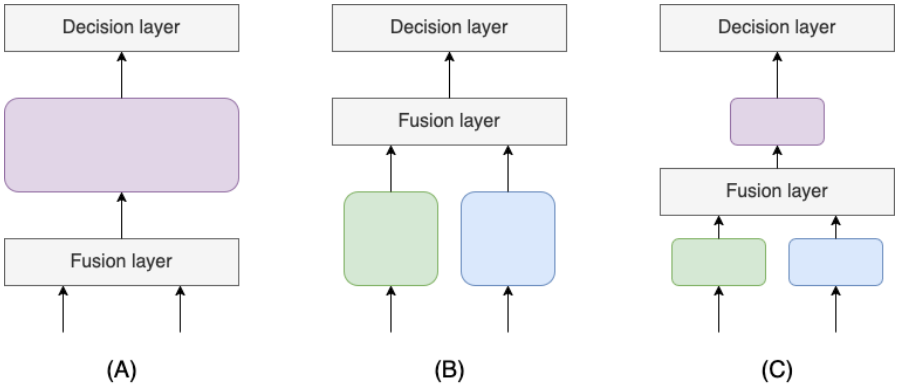


Figure 3.1. Illustration of different multimodal fusion strategies. (A) Early Fusion: Input modalities are fused at the feature level before being processed, leading to a single latent representation. (B) Late Fusion: Each modality is processed independently through its own feature extraction network, and only the decision-level representations are combined before reaching the final decision layer. (C) Mid Fusion: Separate modality-specific networks extract features independently before fusion, allowing each modality to retain distinct feature representations.

3.3.1 Early Fusion

Early fusion refers to methods that combine input data or features from multiple modalities before they are fed into a model (Figure 3.1.A). This integration can occur at different levels, including raw data concatenation, feature-level fusion, or tokenized fusion within a shared model architecture.

Types of Early Fusion

One of the simplest approaches to early fusion is the direct concatenation of raw data representations. For example, audio waveforms can be combined with image pixel values to create a multimodal input representation. However, this approach often results in extremely high-dimensional data, making it computationally expensive and difficult to train.

Instead of fusing raw inputs, feature-level fusion extracts meaningful representations from each modality before merging them. For instance, handcrafted features such as MFCCs from audio can be concatenated with CNN-extracted features from images or videos. This approach reduces dimensionality while maintaining important characteristics from each modality.

Modern multimodal architectures often employ Transformer-based models to fuse different modalities at the feature level. In this approach, both audio and visual features are tokenized and projected into a shared embedding space, allowing the Transformer to

model cross-modal interactions. This method benefits from self-attention mechanisms, which help capture dependencies across modalities more effectively.

Challenges of Early Fusion

Despite its advantages, early fusion presents several challenges. Concatenating multimodal features, especially at the raw data level, can lead to a dramatic increase in input size. This not only increases computational costs but also makes training more difficult due to the curse of dimensionality. Careful feature selection or dimensionality reduction techniques (e.g., PCA, autoencoders) can mitigate this issue. While early fusion offers a powerful way to integrate multimodal data by preserving cross-modal relationships, addressing these challenges is essential to achieving optimal model performance.

3.3.2 Late Fusion

Late fusion refers to methods where decisions from multiple independently trained models, each processing a different modality, are combined at the output level (Figure 3.1.B). This approach is particularly useful in scenarios where individual modalities require specialised architectures or when cross-modal alignment is challenging.

Common Late Fusion Methods

A simple yet effective approach to late fusion is applying predefined rules to aggregate predictions. These include:

- **Majority Voting**: In classification tasks, each modality-specific model votes for a class, and the majority prediction is selected.
- **Max-Pooling**: The model outputs are compared, and the class with the highest confidence score across modalities is chosen.
- **Weighted Averaging**: Instead of treating all modalities equally, confidence scores from different models are weighted based on prior knowledge or validation performance.

Instead of relying on fixed rules, a neural network can be trained to combine modality-specific outputs. A common approach is to use a MLP that takes the probability scores or feature representations from each model as input and learns an optimal fusion strategy. Transformer-based fusion models can be employed to model relationships between modality-specific predictions.

Advantages and Disadvantages

One major benefit of late fusion is that each modality can be processed independently, allowing for specialised architectures tailored to the unique characteristics of each data type. This independence also enables more modular and flexible system designs, where new modalities can be added without retraining the entire model. Additionally, late fusion is particularly useful when modalities have vastly different feature distributions, making joint representation learning (as in early fusion) difficult.

However a significant drawback of late fusion is that it does not inherently capture cross-modal interactions, as each modality is processed. Late fusion relies solely on high-level decisions, which may overlook meaningful relationships between modalities. Moreover, training separate models for each modality can be computationally expensive, especially when deep architectures are involved. In cases where cross-modal dependencies are crucial for accurate predictions, late fusion may be suboptimal.

3.3.3 Mid Fusion

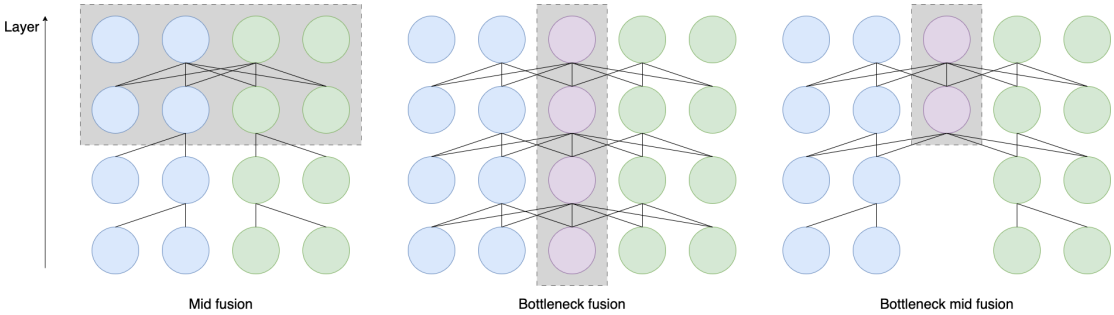


Figure 3.2. Illustration of mid fusion schemes. Mid fusion allows the exchange of cross-modal information. In traditional mid fusion (left), information flows directly between modalities. The bottleneck (centre) acts as a boundary to limit the influence of one modality on the other, while still learning certain intermodal dependencies. The bottleneck mid fusion (right) scheme combines the advantages of letting the first layers learn the low-level characteristics and managing the transfer of information between modalities.

Mid fusion is a compromise between early and late fusion, where modality-specific features are first extracted separately and then integrated at an intermediate stage in the model as illustrated in Figure 3.2. This approach enables each modality to be processed independently in the early layers while still allowing for cross-modal interactions at later stages. By merging latent representations instead of raw inputs or final decisions, mid fusion retains the advantages of both early and late fusion while mitigating their respective drawbacks.

Types of Mid Fusion

A common mid fusion strategy is learning a shared feature space where both audio and visual embeddings are projected. Instead of processing each modality separately throughout the network, a joint embedding space ensures that features from different modalities can be aligned and leveraged for better downstream performance. Attention mechanisms have also proven highly effective in multimodal learning. Cross-modal attention is commonly employed to dynamically adjust the importance of each modality by learning attention weights. This ensures that the model focuses on relevant features from each modality depending on the input context. In Chapter 2, we presented the architecture of Transformers, in particular the attention composed of the Query, Key and Values tensors. In the case of single-modality processing, these tensors are generally projections of the same input. For multimodality, and mid fusion in particular, we can use one modality for the Query tensor, and another for the Key and Value tensors.

Bottleneck fusion [27] introduces a technique to regulate the amount of information transferred between modalities, ensuring that one modality does not dominate the fusion process. This is typically achieved through:

- Cross-Attention Mechanisms, which selectively control how much information each modality contributes to the final representation.
- Gating Mechanisms, which dynamically adjust the contribution of each modality, allowing the model to learn optimal fusion strategies.

Bottleneck fusion helps prevent redundancy and enables efficient information sharing while preserving modality-specific features.

Advantages and Challenges of Mid Fusion

Mid fusion provides a balance between modality independence and cross-modal interaction making it a flexible choice for multimodal learning. By allowing early layers to specialise in modality-specific feature extraction while enabling later layers to integrate information, mid fusion captures correlations across different data types without excessive computational costs.

However, mid fusion also introduces challenges. The model must effectively learn to align and merge features from heterogeneous modalities, which may have different temporal resolutions or feature distributions. Additionally, selecting the optimal fusion layer and tuning the fusion strategy (e.g., attention weights, bottleneck constraints) requires careful experimentation.

3.4 In Brief

Summary for Chapter 3

- This chapter introduced different representations for audio and visual data, ranging from raw signals to engineered and deep-learning-based features. Each approach offers trade-offs between information retention, computational cost, and interpretability.
- We then explored fusion techniques that integrate multimodal information at different stages of processing. These approaches are categorised into early fusion (feature-level integration), late fusion (decision-level combination), and mid fusion (intermediate-level integration), each with its own benefits and limitations.

Perspectives for Chapter 3

- The choice of data representation remains an open research question, as different approaches balance efficiency, interpretability, and task-specific effectiveness. Deep features have demonstrated superior performance but often lack explainability.
- Future research may explore adaptive and dynamic fusion mechanisms that can adjust based on task requirements, data availability, and computational constraints.

Part II

Affective Computing

Chapter 4

Core Concepts in Affective Computing

Contents

This chapter is based on the following publication:

- H. Bohy, "Sensing the mood of a conversation using non-verbal cues with Deep Learning," In *2022 10th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, Nara, Japan, 2022.

This chapter introduces several core concepts of Affective Computing. First we define the terms, present key challenges in the domain and some applications (Section 4.1). Second we describe low-level affect descriptors for non-verbal communication (Section 4.2) and high-level descriptors such as emotions (Section 4.3). Finally we summarise the chapter and present future perspectives (Section 4.4).

4.1 Introduction

4.1.1 Definition

AC is a multidisciplinary field that integrates computer science, psychology, and cognitive science to develop systems capable of recognizing, interpreting, and responding to human affects. The term was first introduced by Rosalind W. Picard in her groundbreaking work Affective Computing [28], where she outlined how computing can relate to, arise from, or influence human emotions.

The importance of emotions in human life is significant, shaping our perceptions, decisions, and interactions. One of Picard's key arguments reinforces this idea: emotions influence human behaviour more strongly than logic or law. For instance, despite the existence of strict laws and severe punishments, emotional and social factors often have a greater impact on human actions—highlighting the importance of studying emotions in both human psychology and artificial intelligence. Izard in a treatise on emotion theory [29] describes emotions as a motivating force in perception and attention, while Leidelmeijer [30] emphasizes their strong connection with cognitive processes: "Once the emotion process is initiated, delibarate cognitive processing and physiological activity may influence the emotional experience, but the generation of emotion itself is hypothesized to be a perceptual process".

Affective states occur during various scenarios, either widely studied (e.g., detecting stress through various modailties [31–33], sentiment analysis on social media [34,35]) or intuitively logical (e.g., personalizing learning content based on a student's emotional responses , designing robots that can adapt to human moods for better interaction).

4.1.2 Key Challenges

Affective Computing remains a highly complex task due to the subjective, cultural, and multimodal nature of affects. Several challenges arise in accurately detecting and interpreting emotional states, spanning issues related to data variability and ethical considerations.

Affect Recognition Complexity Affects are highly subjective and vary significantly between individuals. Unlike objective data such as speech transcripts or physiological

measurements, affective expressions are influenced by personal experiences, social norms, and cultural backgrounds. The same facial expression or vocal tone may convey different affects depending on the context and individual differences.

Moreover, affects are expressed through multiple channels, including facial expressions, vocal characteristics, gestures, body movements and physiological signals. Because of this multimodal nature, affect recognition systems must integrate diverse data sources.

Ethical Concerns in Affective Computing The deployment of emotion-tracking technologies raises significant ethical and privacy concerns, especially as these systems are increasingly integrated into workplaces, healthcare, and surveillance applications.

- Many emotion recognition datasets are biased toward certain demographics, cultures, or emotional expressions. For instance, datasets containing mostly Western facial expressions may misinterpret emotions from individuals of different cultural backgrounds [36]. This can lead to misclassification and unfair decision-making, particularly in applications such as hiring processes, mental health assessments, and security surveillance.
- Affect recognition often requires continuous monitoring through cameras, microphones, and physiological sensors. This raises concerns about informed consent and data security. Users may not always be aware that their emotional data is being collected or how it is being used. Ethical frameworks such as General Data Protection Regulation (GDPR) emphasize the need for explicit user consent and data anonymization in affective applications [37].
- DL-based emotion analysis could potentially be used for manipulative purposes, such as targeted advertising, political influence, or workplace surveillance. For instance, models trained to detect stress levels might be exploited by employers to monitor employees' affects without their consent, leading to ethical dilemmas in workplace environments.

4.1.3 Applications of Affective Computing

In the healthcare domain, Affective Computing plays an important role in mental health diagnosis and support. Advanced algorithms analyze speech patterns to detect conditions such as depression, while systems adapt interactions with individuals on the ASD to provide tailored therapeutic assistance, as discussed in [1] and implemented in [38]. These technologies ensure that patient care is not only clinical but also empathetic [39].

The education sector witnesses the rise of learning systems designed to adapt to students emotional needs. These systems use affective computing to adjust teaching strategies based on real-time feedback about student engagement and emotions, creating a more inclusive and supportive learning environment [40].

Human-Robot Interaction (HRI) is another area where Affective Computing makes significant impacts. Social robots are increasingly deployed in customer service roles, offering empathetic responses that enhance user satisfaction. Additionally, these social robots provide therapeutic companionship to individuals needing emotional support, fostering connections that improve mental well-being.

Entertainment and gaming experiences are redefined through adaptive systems that respond to player emotions and stress levels. These technologies create immersive environments where content is personalized based on audience engagement and preferences, making entertainment more impactful and resonant with users [41].

4.2 Low-Level Affect Descriptors

Human communication is a complex interplay of verbal and non-verbal elements [42]. Verbal communication, such as speaking, writing, or singing, operates within structured rules such as grammar and semantic meaning. In contrast, non-verbal communication conveys information through tone, gestures, body posture, and positioning. Studies indicate that between 60 and 90% of communication is non-verbal [43]. This form of communication can be both intentional and unintentional. Intentional cues include actions like pointing to indicate something, while unintentional cues come from physical characteristics that hint at age or other attributes. The way these cues are interpreted does not follow specific rules but are instead shaped by the context in which they occur, whether geographical, educational, or temporal, and can evolve dynamically within interactions.

The environment also influences non-verbal communication, serving as a source of numerous signals that impact interactions. Each element shapes interactions, often without conscious awareness on the part of the individual. For instance, an individual engaged in conversation within a messy space may find their interaction quality reduced, showing how external factors can influence human communication.

Verbal and non-verbal communication operate simultaneously, making disentanglement between the two challenging. Words can transmit emotional messages, while non-verbal cues extend beyond mere emotional expression. As highlighted in [44], Ekman examines how verbal and non-verbal behaviors interrelate, whether by repeating a message through different channels or complementing it. This interlacing of verbal/non-verbal communication and how DL models manage it are studied and discussed in Chapter 8.

Information from various modalities can either be congruent, reinforcing each other for more effective communication (e.g. a smiling face paired with a cheerful tone strengthens the expression of happiness). Alternatively, information from different modalities can complement one another, enhancing clarity and reducing uncertainty—such as a confident tone matched with a smiling face. There's also potential for interaction between

modalities to generate new meanings through contradictory information—for instance, expressing irony by conflicting facial and vocal behaviors.

4.2.1 Types of Non-Verbal Communication

As noted in section 4.1, non-verbal communication covers a wide range of behaviours and signals that convey meaning without words. This section shows the different types of non-verbal communication, exploring their importance in human interaction and emotional computing.

Facial Expressions

Facial expressions play a crucial role in non-verbal communication. They serve as a key component in understanding human emotions and reactions without relying on verbal cues. Multiple models aim at representing facial expressions, based on handcrafted features like Facial Action Coding System (FACS) or based on DL methods.

The FACS, developed by Paul Ekman, is a key tool in this field [45]. It breaks down facial movements into specific components known as Action Unit (AU), each representing distinct muscle movements (Figure 4.1). This method has played a key role in research into emotion recognition and behavioural analysis, providing a systematic approach to understanding facial expressions.

More recently, modern automated facial recognition technologies leverage deep learning models such as CNNs and Transformers. As presented in Chapter 2 these systems offer greater accuracy and scalability than traditional methods, enabling complex scenarios to be analysed with greater precision. They represent a significant advance in the ability to interpret facial expressions computationally.

Vocal Cues

Speech cues play a central role in human interaction, as an essential component of non-verbal communication. These cues encompass aspects such as pitch, volume, tone and speech patterns, all of which contribute significantly to conveying emotions, intentions and attitudes beyond the simple words spoken. Their importance lies in their ability to provide context and depth to conversations, allowing individuals to assess the emotional state and authenticity of others.

In terms of representation, as discussed in Chapter 3, traditional features focus on extracting specific acoustic properties from speech signals. These include pitch variations, intensity, formants and other measurable attributes that can be analysed using signal processing techniques. By identifying patterns in these characteristics, researchers can infer emotional states or communication dynamics, offering valuable insights into how speech signals influence interactions.

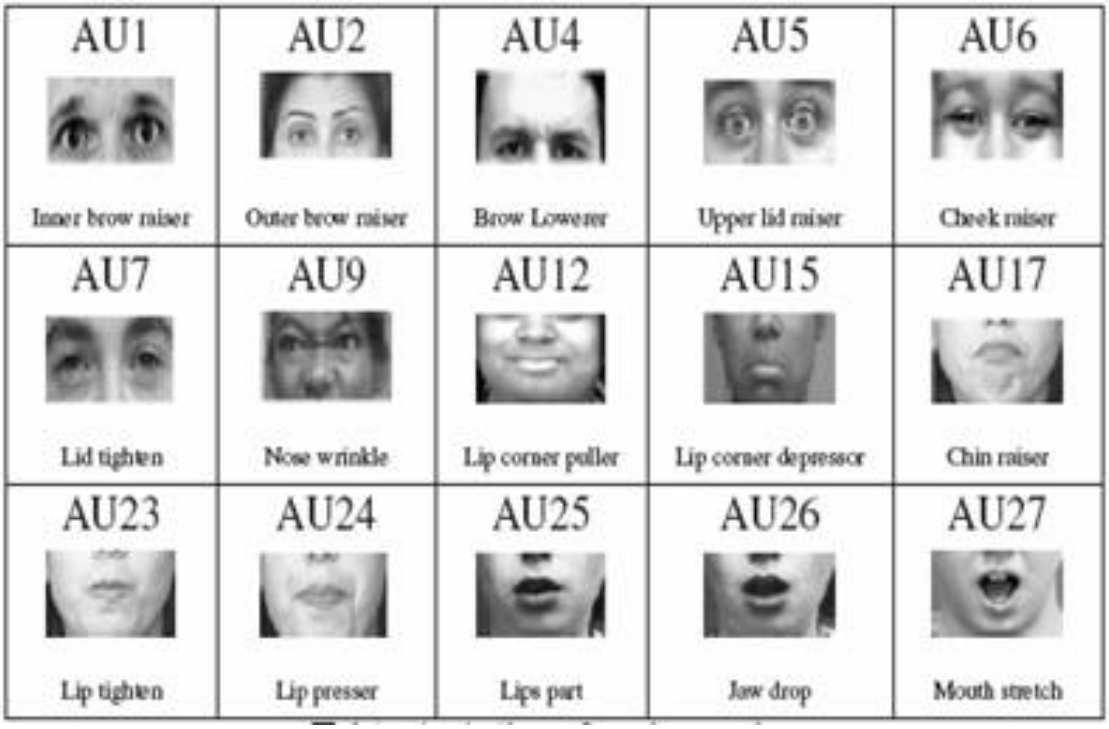


Figure 4.1. Examples of some Action Units extracted from [46].

On the other hand, deep learning approaches have brought about a breakthrough in the representation of speech signals by employing neural networks to automatically learn complex patterns from raw audio data. Unlike traditional methods, deep models capture the complex and subtle aspects of speech through layers of abstraction. This capability improves the accuracy and adaptability of systems for recognising nuanced emotional expressions.

Body Language and Gesture Recognition

Our bodies communicate information whether in a static way (posture) or dynamicaly (gestures). Hand gestures, microexpressions, the direction of the gaze, the dilation of the pupils, and even subtle movements like nodding or shrugging are all factors that allow us to interpret a person emotional state and intentions. These non-verbal cues play a crucial role in human interaction, often conveying messages that words alone cannot express.

Body language analysis extends beyond gestures to include aspects such as proxemics (the use of personal space) and kinesics (the study of body movements). Understanding these elements helps in interpreting emotions, detecting deception, or assessing comfort

levels during interactions. For instance, crossed arms might indicate defensiveness, while open postures can signify approachability.

Pose estimation is a key technological advancement that enables the detection and tracking of human body keypoints from images and videos. This involves identifying specific points on the body, such as joints, to analyze posture and movement accurately. One of the most prominent frameworks in this field is OpenPose [47]. It employs advanced computer vision techniques to detect human body, face, and hand keypoints, providing a comprehensive understanding of posture and gestures.

Physiological Signals

Non-verbal communication based on physiological signals involves analyzing data collected from sensors like Electroencephalogram (EEG) or ECG. These devices measure biological processes within the body, providing insights into emotional reactions that are not always explicitly expressed. For instance, an EEG can capture electrical activity in the brain associated with different emotional states, while an ECG measures heart rate variability to reflect stress levels.

In addition to EEG and ECG, other physiological signals such as Galvanic Skin Response (GSR), which measures changes in skin reddening, and Electromyography (EMG), tracking muscle activity, are also utilized. These sensors detect subtle changes in the body that correspond to emotional experiences, offering a more comprehensive understanding of human emotions. One of the main drawback of using physiological signals is their invasive recording conditions [48].

4.2.2 Encoding and Decoding Processes

The general definition of non-verbal communication as stated in [42] expresses communication by means other than verbal elements. It does not indicate whether it refers to the type of signal produced (called *encoding*) or to the interpretation made by the person perceiving it (called *decoding*). Humans encode and decode non-verbal behaviour daily with varying degrees of awareness and control. There are times when the responses are carefully planned and we are fully aware of what we are doing; and sometimes our responses are more automatic and have little planning or awareness. For example, pose for photographs implies a high level of awareness and control while nervous mannerisms are often enacted outside of our control [49].

Although non-verbal communication is universally used, encoding and decoding depend on several contexts such as culture, transmission channels and even the mood of the participant. Interpretation errors occur when the actors in the interaction do not have the same judgements for similar non-verbal cues, which reduces the accuracy of the communication.

4.3 High-Level Affect Descriptors

4.3.1 Models of Emotions

In the paper "What is Emotion?" [50], authors argue that physical reactions precede emotional feelings, such as a racing heart preceding fear. This theory, part of the James-Lange framework of emotion, suggests that physiological signals can be used to infer emotion. However, this theory is limited if we consider that emotions can influence physiology or occur without any perceptible physical change. Modern definitions of emotions are rather broad, but share the common characteristic that they are complex states involving both physiological and mental changes [51]. Several modeling approaches exist to describe emotional states. The next sections present two widespread approaches: discrete and dimensional.

Discrete Classes Ekman's Theory of Emotions [52] identifies specific, distinct emotions such as happiness, sadness, anger, disgust, surprise, fear, and contempt. These emotions are considered universally recognizable across cultures. Plutchik expanded on this with his Wheel of Emotions [53], which illustrates how basic emotions can combine to form more complex ones (Figure 4.2).

Continuous Dimensions Dimensional models represent emotions along continuous scales rather than as separate categories. Russell's Circumplex Model [54] uses two dimensions: valence (positive to negative) and arousal (calm to excited) (Figure 4.3). The Pleasure-Arousal-Dominance Model [55] incorporates three dimensions: pleasure, arousal, and dominance. While these models capture a broader spectrum of emotional states, they may be less intuitive for humans to interpret.

4.3.2 Personality traits

Personality traits refer to long term characteristics that shape human personality. They play a significant role in shaping how individuals express emotions and interact with their surroundings. In this section we present the Big Five Theory Personality Model [56] (OCEAN) framework for understanding personality and its use for human-agent interaction.

The Big Five Personality Model (OCEAN) One of the most widely recognized frameworks for understanding personality is the Big Five Personality Model, often referred to by the acronym OCEAN. This model identifies five core dimensions of personality: Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism.

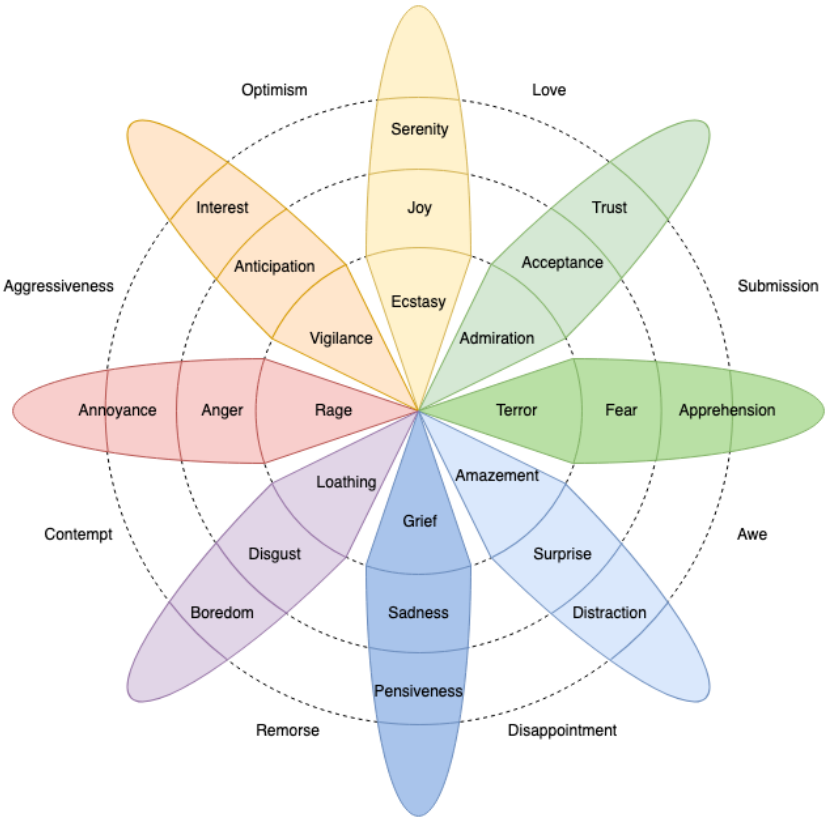


Figure 4.2. Plutchik's Wheel of Emotions. A visual representation of basic and complex emotions organized into eight primary bipolar categories: Joy–Sadness, Trust–Disgust, Fear–Anger, and Surprise–Anticipation. Emotions intensify toward the center (e.g., Serenity → Joy → Ecstasy) and merge into more complex emotions (e.g., Joy + Trust = Love).

- High score in openness refers to individuals that tend to be curious, creative, and open to new experiences. They are more likely to engage with novel ideas and environments.
- Conscientiousness is characterized by organization, responsibility, and goal-directed behavior. Conscientious individuals are typically reliable and methodical.
- Extraverts are sociable, energetic, and enjoy interacting with others. They often seek stimulation and thrive in dynamic social settings.
- People high in agreeableness are empathetic, cooperative, and trusting. They value harmonious relationships and are less likely to engage in conflict.

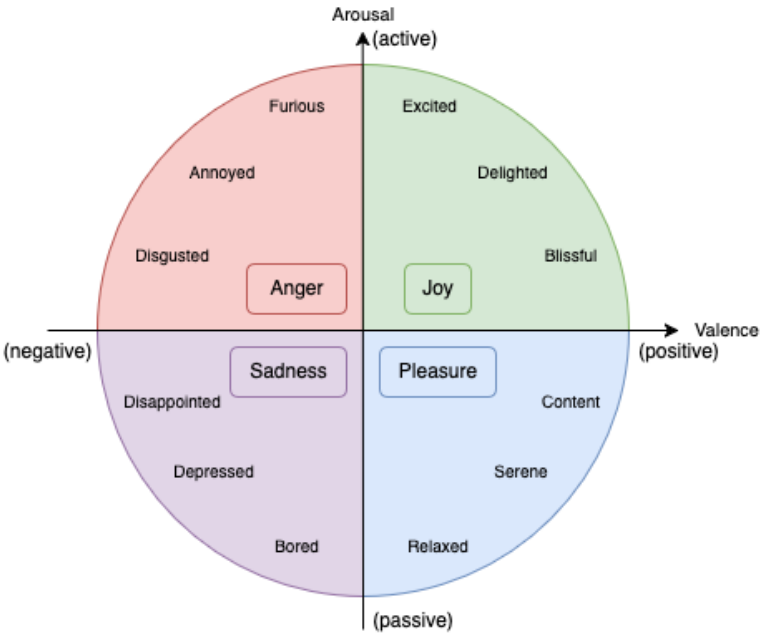


Figure 4.3. Russell's Circumplex Model.

- Neuroticism is associated with emotional instability, anxiety, and a tendency to experience negative emotions. Neurotic individuals may be more sensitive to stress and challenges.

These dimensions of personality significantly influence how individuals express their emotions and interact with others. For instance, openness might affect how someone engages with new technologies, while conscientiousness could impact their adherence to structured interactions. Similarly, extraversion and agreeableness can shape communication styles and social engagement, which are critical factors in human-agent interaction.

Personality in Human-Agent Interaction One key application of personality-aware agent is in conversational interfaces such as chatbots and virtual assistants. By analyzing user inputs and adapting their responses based on inferred personality traits, these systems can provide a more personalised and engaging experience. For example, an extroverted user might receive more dynamic and interactive responses, while an introverted user could be offered quieter, more reflective interactions.

4.4 In Brief

Summary for Chapter 4

- AC aims to build systems that can recognize, interpret, and respond to human emotions, bridging computer science and psychology.
- Challenges include emotion subjectivity and cultural variability, multimodal nature of affect (voice, face, gestures, physiology) and ethical concerns (data privacy, bias, and responsible use).
- Non-verbal communication (facial expressions, vocal tone, gestures, physiological signals) is critical for affect recognition.
- We introduced emotion models (discrete and dimensional) and the OCEAN describing personality trait.

Perspectives for Chapter 4

- Current systems lack customisation for a specific user. We could consider moving towards emotion recognition systems that adapt to individual differences.

Chapter 5

Datasets

Contents

This chapter is based on the following publication:

- Kevin El Haddad, Hugo Bohy, and Thierry Dutoit. "The Interaction Behavior Dataset: A Dataset of Smiles and Laughs in Dyadic Interaction". In *2025 IEEE International Conference on Artificial Intelligence and eXtended and Virtual Reality (AIxVR)*, 2025.

Affective Computing (AC) covers a huge number of fields of application in many different modalities. Recent AI systems require large quantities of data in order to be the most effective in terms of recognition or detection. These quantities of data are at the heart of a major problem: their availability. Although there is more and more data available for research, it is still possible to contribute by providing new content or increasing the number or quality of the annotations.

This chapter first introduces datasets available for research, focusing on the face and/or voice community (Section 5.1). Then, we present our proposition for additional annotations to existing databases to include laughing and smiling expressions (Section 5.2).

5.1 Existing Affective Datasets

In this section we discuss datasets that contain either voice, face or both. We present them as either unlabeled or labeled affective-wise. Unlabeled datasets are mostly used for pre-training models in unsupervised tasks such as de-noising [57]. Labeled datasets allow us to evaluate performance of models in affective use-cases. While some datasets were not directly used, they inform on the available resources in the field.

5.1.1 Unlabeled datasets

Librispeech [58] is a dataset that contains 1000 hours of audiobook English speech content. It is distributed across more than 2,400 speakers with a relatively balanced gender distribution (51.5 % male speakers). Half of the dataset is considered as "clean" recording quality based solely on word error rate (WER) results. 20 female and 20 male speakers have been extracted from the "clean" set to form a clean validation, and the same process was applied for the test partition, leading to 5 hours of content for each partition.

PodcastFillers [59] contains 199 podcast episodes in English annotated for filler words. It was curated to maintain a gender-balanced distribution of 350 speakers leading to 145 hours of content. Since it was extracted from podcast, it is considered as recorded in naturalistic conditions.

VoxPopuli [60] is a huge dataset that contains over 400,000 hours of unlabeled audio recording from European Parliament events. For comparison purposes, we report here the transcribed portion of the dataset, which is limited to 1791 hours and 16 languages (without equal distribution between languages). With over 4300 identified speakers, it represents one of the most diverse set available, although its gender balance is 34 % of female and 66 % of male.

VGGFace2 [61] is an image-only dataset that contains over 3.31 million images of 9131 speakers. The content comes from Google Image and can be considered as naturalistic

record conditions. The gender distribution across subject is 59.3 % of male and 40.7 % of female.

The Edinburgh International Accents of English Corpus (EdAcc) [62] is a dataset that aims to represent the variety in English accents. The content is video recording of video calls between friends. It contains almost 40 hours from 120 speakers, in various dyadic settings.Female and male speakers account for 51.2 % and 48.8 % of the population, respectively.

The VoxCeleb2 [63] dataset is a large-scale audio-visual dataset primarily focused on speaker recognition. It serves as a valuable resource for research in self-supervised training as it contains more than a million utterances with 6,000 speakers of 145 different nationalities. It provides a wide range of languages, accents, ethnicities and ages from real-world recordings.

Table 5.1. Comparison of unlabeled datasets according to several criteria: number of speakers, gender distribution, modality type, duration, language. *gender-balanced is specified but no value is provided.

Name	Modality	# speakers	Duration (h)	Languages	Gender Distr. Male | Female
Librispeech	audio	2400+	1000	En	51 % | 49 %
PodcastFillers	audio	350	145	Eng	n.a. | n.a.*
VoxPopuli	audio	4300+	1791	En, De, Fr + 13 others	66 % | 34 %
VGGFace2	visual	9131	n.a.	-	59 % | 41 %
EdAcc	audiovisual	120	40	En	49 % | 51 %
VoxCeleb2	audiovisual	6000+	2442	En	61 % | 39 %

5.1.2 Labeled datasets

The Emotional Voices Database (EmoV-DB) [64] contains audio recordings in English and French from 5 individuals (En: 2 males, 2 females; Fr: 1 male). Speakers were asked to read sentences while expressing discrete emotions (Angry, Sleepy, Amused, Neutral, Disgust). The total number of sentences is 7590.

Emotional Voice Message (EMOVOME) [65] is a dataset of real-life phone conversations. It includes 999 voice messages from 100 Spanish speakers (50% males, 50% females), labeled with continuous (valence/arousal) and discrete (Angry, Sad, Happy, Surprise, Fear, Disgust, Neutral) emotions.

MSP-Podcast [66] is one of the largest naturalistic speech emotional dataset. It contains over 237 hours of podcast recordings annotated by crowdsourcing in both continuous (valence/arousal/dominance) and discrete emotions (Angry, Sad, Happy, Surprise, Fear, Disgust, Contempt, Neutral). Speech samples are distributed between Train, Validation and three Test partitions from more than 1800 English speakers.

AffectNet [67] is an image-only dataset that contains 450,000 manually annotated facial images. It includes thousands of individuals (49% males, 51% females) and labels cover continuous (valence/arousal) and discrete (Angry, Sad, Happy, Surprise, Fear, Disgust, Contempt, Neutral) emotions.

Aff-Wild2 [68] dataset consists of 558 videos of humans in real-world conditions. 458 different individuals (60.9% males, 39.1% females) are included in the dataset and each video is labeled in dimensional (valence/arousal) emotions. Data cover a wide genre of age groups and ethnicities, as well as head poses, illumination conditions and occlusions.

The First Impressions dataset [69] contains around 10,000 audio-visual clips extracted from YouTube videos of people facing and speaking in English to a camera. Clips are labelled with apparent Big Five personality traits. In its extended version, transcriptions and job-interview annotations are also available.

MSP-Improv [70] is a dataset of audiovisual recordings of 6 dyadic interactions of English speaking actors (6 males, 6 females). The total duration of the dataset is about 9 hours.

The Crowd-sourced Emotional Multimodal Actors Dataset (CREMA-D) [71] provides a diverse collection of audio and video recordings featuring 91 individuals (48 male and 43 female actors between 20 and 74 years old) acting discrete emotions. Each speaker recorded 12 sentences in English for each emotion leading to 7442 videos ranging from 763 to 2204 utterances per emotion.

Table 5.2. Summary of affect-labeled datasets.

Name	Modality	# speakers	# samples/ duration	Annotation type	Demographics
EmoV-DB	audio	5	7590 sentences	discrete	En: 2M/2F, Fr: 1M
EMOVOME	audio	100	999 messages	cont. + discr.	50% M / 50% F
MSP-Podcast	audio	>1800	237 hours	cont. + discr.	n.a.
AffectNet	visual	1000+	450,000 images	cont. + discr.	49% M / 51% F
Aff-Wild2	av	458	558 videos	continuous	61% M / 39% F
First Impression	av	n.a.	~10,000 clips	personality traits	n.a.
MSP-Improv	av	12	~9 hours	cont. + discr.	50% M / 50% F
CREMA-D	av	91	7442 clips	discrete	n.a.

5.2 Interaction Behaviour Dataset

Due to their importance in daily communication, laughter and smiles have equally interested social scientists and computer scientists focused on human-centered applications and human-agent interactions. This is mainly due to their frequency in interactions, their diverse functionalities, and their presence in societies across the globe. Several datasets and corpora have been built, either containing laughs or smiles, or focusing solely on them [72–76].

This work is based on three different datasets of dyadic interactions. We selected them based on the similarity of the data collection setup favoring naturalistic interactions; including nonverbal expressions, diversity of cultural backgrounds, and the quality of the data. These datasets are CCDb, IFADV and NDC-ME.

The Cardiff Conversation Database (CCDb) [77] contains audiovisual recordings of dyadic interactions. These were in English, and the participants were presumably, in the majority, of British background and so, native English speakers. They were free to discuss any topic, even though some general topics were sometimes suggested to them.

The IFA Dialog Video Corpus (IFADV) [78] is a dataset of audiovisual interactions that were in Dutch with scripted and freely spoken data. The participants were presumably mostly of Dutch background and, therefore, native Dutch speakers.

The Naturalistic Dyadic Conversation on Moral Emotions (NDC-ME) dataset [79] contains audiovisual recordings as well, and the interactions were in English with participants of different backgrounds. The participants asked each other preassigned questions in turns. Most of the participants in this dataset are not native English speakers.

In the future, we intend to consider other datasets to diversify even more the content of our dataset.

5.2.1 Annotation Protocol

Categories Definition

The definition of smiles and laughs were similar to the one in [80]. We summarize them here and explain the definition of roles. The following definitions were used:

Roles A subject is considered a SPK, LSN, or none of them. Conversation segments during which the subject utters an entire utterance (short or long) meant to be communicated to their interlocutor are tagged as SPK, while segments during which the subjects perceive the messages coming from the speakers are tagged as LSN. SPK segments start and end with the utterance including surrounding nonverbal expressions if any (e.g., a spoken sentence ending with a nod). LSN segments start and end when the subjects are perceived to start listening to their interlocutor. The LSN segments

can contain backchannels (like "yes" or other expressions communicating engagement for instance): emitting a word or a short sentence meant as feedback does not make the subjects speakers since they are still in a perceiving role.

Smiles Smiles are annotated as perceived and can be expressed not only with lips but also with other areas of the face like the cheeks, eyelids, eyebrows, etc. Smiles can also be expressed using the audio cue or alongside other expressions [81, 82]. The smiles should then be segmented based on their intensity levels. The intensity of the facial and vocal expressions determine the intensity levels of the smile segments. Four different intensity levels are considered : subtle, low, medium, and high. The subtle level represents smiles that are subtle but are still perceived. They could be covered by other expressions, by speech, or are just of very low intensity but still perceivable.

Laughs A laughter segment starts when an audio, facial expression, or body movement related to laughter starts, and stops when a breath intake sound or movement are perceived (from the stomach, face, etc.). If no breath intake happens, the segment stops with the laugh's movement. Three intensity levels are considered for laughs: low, medium, and high

As mentioned before, laughter and smiling segments cannot overlap.

Annotation Process

Six annotators were asked to annotate one or several of the above-mentioned datasets. These annotators intervene at different periods and for different durations. They were all first trained on the annotation protocol and tested it on a test video. The test annotations were then rechecked by the same supervisor who gave them feedback on their work.

The annotators were instructed to first start annotating the role category, after which they were asked to annotate smiles and laughs simultaneously. The intuition behind this and what we have observed is that a first pass through the annotation of roles helps the annotators familiarize themselves with the dataset's participants' expressions, which would make it easier to annotate smiles, laughs, and their intensities. We found it more efficient to ask the annotators to annotate the smiles and laughs at the same time instead of one then the other.

Given the large amount of data and the limited availability of the annotators, some of the annotators were instructed to annotate the first two minutes of each video to obtain a more diverse dataset in terms of participants at the end, sacrificing the diversity with respect to the conversation's length. Others were instructed to annotate the full videos.

5.2.2 IB Dataset

In this section, we first give an overview of the dataset content. For reproducibility purposes and to give the reader the possibility to estimate the amount of time it would take for an annotator to annotate their datasets, we also give the time spent per annotator per minute of data for each category. We believe that this is an important metric to consider to better understand the efficiency of an annotation protocol, but also because it represents an important characteristic of an annotation protocol to take into account when comparing them with each other.

Dataset Content

Given the interventions of the annotators at different periods, as mentioned in 5.2.1, some of them completed others' work, and most of them, though not all, have annotated common data for inter-rater agreement scoring purposes. Inter-rater agreement specifies how close the annotations of two raters are. The data is separated per dataset and per annotator which makes it easy to select a specific part of the data depending on the requirements. The metrics shown hereafter allow the reader to better estimate if the data are homogeneous enough to be used together or not. They also allow to estimate the risk and limitations before their use.

Table 5.3 shows the total amount of annotated data in minutes produced per annotator (note that each of them spent a different total number of hours annotating). The annotators were asked at the end of each annotation session to manually log the category annotated, the start and end times of the data annotated, and the duration taken to annotate this portion. Most of them spent around an hour a day annotating for a period of about twelve to sixteen weeks.

Table 5.3. Amount of total data annotated in minutes in the entire dataset.

	Roles	Smiles	Laughs
Annotator 1	143	87	6
Annotator 2	105	35	8
Annotator 3	77	49	4
Annotator 4	61	28	1
Annotator 5	103	37	4
Annotator 6	104	25	6

Table 5.4. Average annotation time in minutes, taken per annotator to annotate 1 min of data.

	Roles	S&L	Gender
Annotator 1	8	23	F
Annotator 2	10	31	M
Annotator 3	5	14	F
Annotator 4	10	21	M
Annotator 5	9	12	F
Annotator 6	5	12	M

Performance

Table 5.4 shows the average time spent per minute of data by each annotator for each annotation category, i.e. how long would a minute of data take on average for a person to annotate each category. This table also shows the annotator's gender for a more complete background information. We follow the computer science literature here, and use "gender" but distinguishing male vs. female is more appropriately termed as "sex estimation". Gender is a more complex and subjective construct. They all reported being between 18 and 25 years old at the time of annotation.

Table 5.5 shows the inter-rater agreement (using the Cohen's Kappa Coefficients) of each annotator with all the others for the category of the Role. The inter-rater agreement will be calculated per intensity and for smiles and laughs separately in future work.

Table 5.5. Inter-rater agreement Cohen's Kappa Coefficients between all annotators for the category **Role**

Annot.	1	2	3	4	5	6
1	1	0.55	0.57	0.58	0.65	0.64
2	-	1	0.56	0.59	0.57	0.52
3	-	-	1	0.69	0.60	0.58
4	-	-	-	1	0.7	0.65
5	-	-	-	-	1	0.66

Data Quality and Reliability Metrics

In order to evaluate the annotations quality and their reliability, we first calculated the Cohen's Kappa Coefficient which is a common metric to evaluate inter-rater agreement. The average value was calculated between commonly annotated files among annotators. The exceeding portions annotated were discarded. The results are shown in Table 5.6 and they represent the mean value across all annotation categories.

Table 5.6. Mean Cohen's Kappa coefficient calculated on all common annotated parts for all categories (Roles, Smiles, Laughs and corresponding levels) of files between annotators

Annot.	1	2	3	4	5	6
1	-	0.55	0.81	0.68	0.5	0.48
2	-	-	0.71	0.59	0.74	0.68
3	-	-	-	0.6	0.37	0.24
4	-	-	-	-	0.68	0.56
5	-	-	-	-	-	0.41

Except from annotator 3 compared to 5, and 6 and annotator 5 compared to 6, all the results are above 0.5 which shows a moderate to substantial agreement among annotators [83]. After investigation these lower results come from a limited amount of common files (and annotation categories) between these annotators. In fact some values are above 0.75 for some annotation categories.

Cohen's Kappa is a go to metric for inter-rater agreement but, in our opinion does not give a full view of the dataset's content. So, we present in this work three other metrics that, in our opinion, give a more complete insight on the subjectivity of the task and therefore the reliability of the data. The metrics are:

- *Overlap_perc*: Given the common annotation portions of two annotators, we calculate the percentage of overlapping segments between annotators with respect to all the segments from both annotators, i.e. the ratio of the number of segments that overlap to the total number of segments from both annotators data.
- *IoU*: For each overlapping segments between two annotators, we calculate Intersection over Union (IoU) of the two segments, i.e. the ratio of the overlap duration to the union of both segments.
- *Overlap_levels*: Out of all the overlapping segments we calculate the percentage of the segments with the same label (for example, segments of smiles with the same intensity levels).

Note that we consider two segments coming from two annotators to be overlapping, if the two have more than 10% IoU. This is done to avoid counting the overlap of segments that happen out of lack of precision during the annotations (for example the start of a LSN's segment from one annotator might overlap for hundreds of milliseconds with the end of the other annotator's SPK segment).

Tables 5.7, 5.8, 5.9 show the above-described metrics for the Roles, Smiles and Laughs categories respectively. The *n.a.* indicates a non-availability of results either because no common files were annotated by the annotator pairs or no common data were found due to limited amount of data in some files (this is especially true for laughter data and even more when considering the intensity levels).

We can see that even though some difference occurs in the annotations across annotators due to the subjective nature of the task, in most cases, similar segments are selected for each category when the value of the segment is not considered, with an IoU above 50% (and goes as high as above 80% in several cases). The effect of subjectivity can be clearly seen when it comes to selecting the intensity levels of smiles and laughs though. The latter's intensities seem to be annotated more objectively than smiles. Indeed smiles can co-occur with speech and other expressions which can make them more confusing to identify. Nevertheless, these metrics show that some care has to be taken when combining the different annotations depending on the application targeted, especially if the intensities have to be considered. But the different metrics also show that a significant amount of data can still reliably be combined to obtain a relatively large and diverse dataset.

This dataset or subsets of it have already been used in diverse domains, some of which concluded in published work, from machine learning applications, such as detection systems, to behavioral studies. This further showcases the usefulness of this dataset, the reasonable quality of the annotation and therefore the efficiency of the annotation protocol. Indeed, they were successfully used to build and study smile and laugh detection systems [84, 85], to build laughter synthesis [86], and to analyses studies of smiles and laughs [87]. Some of these work are discussed in later chapters.

Table 5.7. Concerning the **Roles** for all common annotation files between annotator pairs, and from top to bottom in each cell: mean *Overlap_perc*, mean *IoU* and mean *Overlap_levels* (SPK vs LSN in this case).

Annot.	2	3	4	5	6
1	75 71 86	99 90 94	99 93 95	99 62 68	97 61 70
2	-	n.a.	100 70 87	n.a.	98 81 89
3	-	- -	100 54 67	100 57 74	100 62 69
4	-	-	-	96 65 66	99 64 70
5	-	-	-	-	99 67 75

Table 5.8. Concerning the **Smiles** for all common annotation files between annotator pairs, and from top to bottom in each cell: mean *Overlap_perc*, mean *IoU* and mean *Overlap_levels* (smile intensity levels).

Annot.	2	3	4	5	6
1	77 65 36	99 94 92	92 80 54	93 52 17	69 57 32
2	-	n.a.	91 60 34	n.a.	76 98 63
3	-	-	93 63 43	96 53 24	94 59 40
4	-	-	-	87 53 27	69 57 37
5	-	-	-	-	87 52 46

Table 5.9. Concerning the **Laughs** for all common annotation files between annotator pairs, and from top to bottom in each cell: mean *Overlap_perc*, mean *IoU* and mean *Overlap_levels* (laughs intensity levels).

Annot.	2	3	4	5	6
1	79 65 32	n.a.	100 85 100	83 89 100	87 49 51
2	-	n.a.	60 75 50	n.a.	86 68 74
3	-	-	n.a.	67 86 100	60 49 0
4	-	-	-	100 77 100	84 46 52
5	-	-	-	-	75 58 33

5.3 In Brief

Summary for Chapter 5

- We highlighted unlabeled and labeled datasets, used for pre-training and affective task evaluations respectively.
- We presented a dyadic interaction dataset annotated by several annotators with a protocol and metrics aiming at optimizing the annotation process.
- We proposed metrics measuring the annotation's reliability and efficiency in order to insure the quality of the data and the efficiency of the process.

Perspectives for Chapter 5

- As the lists are non-exhaustive, some datasets of interest can still be added. It does not include modalities other than audio and visual, such as motion capture, EEG or ECG.
- Future work will focus on improving the quality of the data provided and increase the diversity of the annotations. This will be done through continuously improving the annotation process and through deeper analyses of the results.

Chapter 6

Regions of Interest: A Focus on Lips for Smiles and Laugh Detection

Contents

This chapter is based on the following publications :

- H. Bohy, K. El Haddad, and T. Dutoit, "A new perspective on smiling and laughter detection: Intensity levels matter", *in 2022 10th International Conference on Affective Computing and Intelligent Interaction (ACII)*. IEEE, 2022.

This chapter presents the detection of non-verbal expressions in voice and in a restricted area of the face around the lips. We first introduce our goals, then present previous work in the domain (Section 6.1). Subsequently, we describe the datasets used (Section 6.2) and finally we discuss our experiment (Section 6.3). We conclude with a summary and future works (Section 6.4).

There is an enormous amount of non-verbal cues and even more combination for humans to encode and decode during interactions. To reduce the scope of this research, we decided to limit our interest on non-verbal expressions in the facial region around the lips, region that we refer to Region of Interest (ROI). Data annotated on non-verbal cues is quite limited, hence we decided to limit our analysis to two expressions: smiles and laughs. As stated in Chapter 5, these are among the most common non-verbal expressions and are less susceptible to contextual ambiguity.

The goal of this work is two-fold: first, our aim is to explore how such a model can recognize smile and laugh expressions, and to understand its decision-making process through post-training analysis; second, we examine how the intensity of the detected expressions affects classification performance. We adopt the commonly held consensus that smiling and laughing are two distinct expressions. The pre-trained model we use combines both short-term and long-term information extractors, as its original purpose was to enable speech transcription based solely on lip movements and associated audio features.

6.1 Related Work

6.1.1 Region of Interest detectors

Non-verbal expressions cause structural changes in several regions of the human body, like the pose, facial landmarks or skin reddening. Several ROIs have been identified across the face (eyebrows, eyes, nose and mouth) and resources provides models to standardise them or their dynamic. The most popular approach is FACS [45], which provides a list of face regions named AU. Authors in [88] rely on image segmentation based on anthropometric measurements to detect the eye and eyebrow regions as well as the mouth region. While the method has noticeable results, it has limitations arising for example from shadows or facial wrinkles. In [89], authors use a CNN-based model to detect iris boundaries in face images. In [90] authors work on DeepFake detection, where the identity of the person on screen is created or altered visualy. They use AUs to build a user profile and detect inconsistencies in specific ROIs.

6.1.2 Smiles and Laugh detection

A plethora of work can be found on smile detection. We estimate that the vast majority of them are based on the visual cue as we could find very few work based on other

modalities [91–93], notably the audio cue was rather absent from the state-of-the-art although smiles were proven to be recognizable audible [94–96].

Fewer work can be found on laughter detection. They focus on the audio and the visual modalities individually but also in multimodal approaches. Kantharaju et. al. in [97] present an automatic detection of different categories of laughter using audio-visual data. The authors in [98] use full-body motion capture data to detect laughter while [99] investigates the laughter detection based on audio and facial motion capture.

Surprisingly, very few work can be found where S&L are considered as two distinct expressions, and none of them attempts to classify/detect them as different entities. Indeed, even though the authors in [100] annotate them as two expressions in their work, they build classifiers considering them as the same class. The authors in [101] propose a system classifying smiles vs non-smiles based on the visual cue and laugh/non-laugh based on a single modality and on multimodal data, but no smile/laugh discrimination is presented. One reason for this might be the difficulty for the models to learn the differences between smiles and laughs, especially given the limited amount of resources available. Another reason might be the common representation for some, of smiling being a less intense expression of laughter or even both being the same expression, which is to the best of our knowledge, unproven yet.

6.2 Datasets

The data used here are subsets of the Nonverbal Dyadic Conversation on Moral Emotions (NDC-ME) [79], and of the IFA Corpus (IFADV) [78] for which the Smiles and Laughs were annotated following the protocole described in Chapter 5.

NDC-ME is an audiovisual collection of dyadic interactions focusing on the emotions expressed during speaker-listener interactions. The subset we use is distributed in 17 dyadic interactions split between 10 male and 4 female individuals, with 7 male-male, 6 male-female and 4 female-female pairs. During these interactions, each duo discusses emotional topics introduced by an open question in English. Since some of those interactions are not fully annotated, the total duration of annotated data is about 90 minutes with an unbalanced distribution between individuals.

IFADV is also a collection of audio-visual recordings of dyadic conversations. The subset we used contains 23 dyadic interactions of 15 male and 28 female Dutch individuals with 4 male-male, 8 male-female and 11 female-female pairs of interactions. The annotations cover only the first two minutes of each file, leading two around 46 minutes of annotated data.

The laughs intensities are divided in three levels (low, medium and high) and the smiles intensities in four (subtle, low, medium and high). As stated in Chapter 5, the subtle level was added to capture all the levels of smiles even the ones that are normally left out because of the difficulty to annotate them: subtle smiles co-occurring with other

expressions for instance. A third class, referred to as the None class, includes all segments of the recordings that contain neither laughter nor smiles, such as neutral expressions and speech. Therefore we ended up with three main classes Laughs, Smiles and None, which will be used for training our models without taking into account the intensity levels.

6.3 CNN-based classifier: LSN-TCN

Since the emergence of deep-learning models, architectures that rely on CNNs have improved the results on many classification tasks. In affective computing, whether we focus on speech, face or multimodality, the results have followed the same learning pace. Over the years, fine-tuning have become the generic approach for domain-specific tasks: the main principle lies in the use of models that were previously trained on huge datasets, then, a second training step is performed on a small set of custom data. This leads to improved performance with less training cost.

Based on the fine-tuning principle, we benefit from patterns already learned during pre-training tasks and makes minor changes on the model first CNN layers to fit our smile and laugh data. We explore the impact of voice-only, ROI-only or combined input data. As our focus is to study region of interest around the lips, we selected MS-TCN [102] as our backbone model. By the time of publication of our work, it was the best compromise between accuracy and cost-efficiency, characteristic non-negligeable as we were limited in computation power. MS-TCN uses CNN layers to capture both temporal and spatial information within multiple frame of the same video to perform lipreading.

The analysis is split in three parts: speech, face and multimodal analysis for expression classification (Laugh, Smile or None expression). We use the datasets presented in Section 6.2: NDC-ME and IFADV. We split NDC-ME randomly into three partitions namely train (70%), validation (15%) and evaluation (15%), while IFADV is used for evaluation only so that we can measure inter-data learning. These two datasets contains expression labels with additional information about each expression intensity. None is only neutral, smiles are either subtle, low, medium or high and laughs are either low, medium or high. The sub-label analysis allows us to look at the distribution of expression intensity by the model without any knowledge of said intensity during training. The multimodal approach compared to unimodal ones highlights the interrelations between speech and facial expressions which increases machine efficiency in laugh/smile detection. In addition to expressions analysis, we also uncover the importance of pre-training and cross-validation across different datasets by using only NDC-ME for training and both NDC-ME and IFADV for evaluation.

We named our model *Laughs, Smiles, None-Temporal Convolution Network* (LSN-TCN) [84], an adapted version of Multi Scale-Temporal Convolutional Network (MS-TCN) architecture.

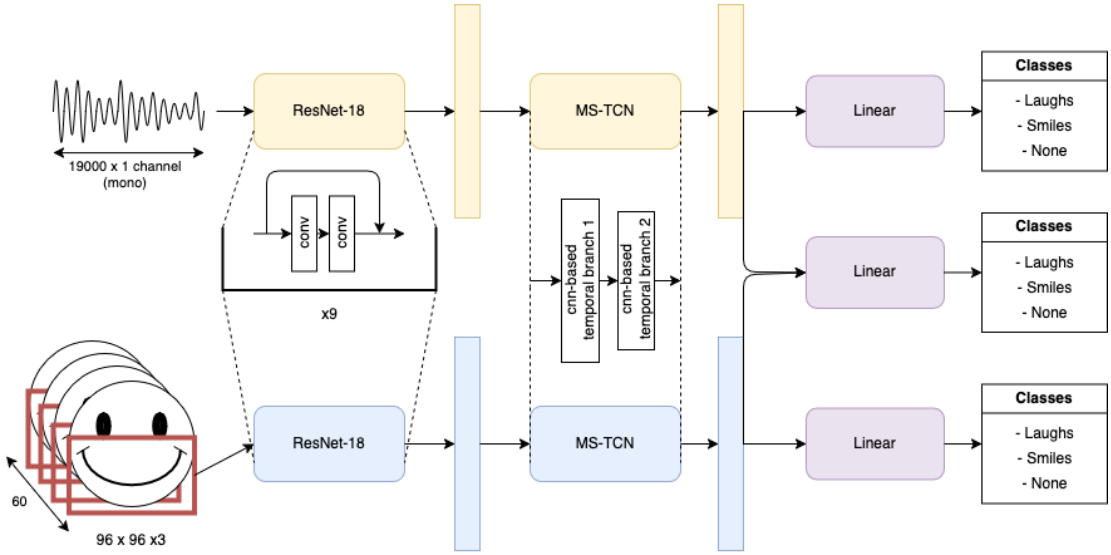


Figure 6.1. Schematic of our LSN-TCN architecture. Audio and Video branches are represented in yellow, blue respectively while purple represent classification layers. Input shapes are specified below their respective representations.

6.3.1 Speech analysis

In the audio model (upper branch in Figure 6.1) the raw audio input is projected to a higher-dimensional space using a single-layer 1D-CNN:

$$x \in \mathcal{R}^{N \times d_a}$$

x is fed to a ResNet-18 [3] backbone. This backbone consists of 18 convolutional layers and an additional residual operation $\mathcal{H}(x)$ every two layers. It attempts to learn residual functions $\mathcal{F}(x)$ and adding them to the input x to retrieve the mapping

$$\mathcal{H}(x) = \mathcal{F}(x) + x$$

learned by non-residual model. This tackles the degradation of accuracy in deep models by highlighting the features extracted. In the case of ResNet, $\mathcal{F}(x)$ is the output from two successive convolution layers.

Using a backbone such as ResNet guaranties the extraction of usable features. On top of the backbone, the MS-TCN architecture is a multi-layered combination of 1D convolutions designed to model short and long term temporal information. It is composed of multiple blocks that follow the same architecture with progressively bigger kernels and use non-linear activation functions such as ReLU between layers. MS-TCN have proven efficient on sequence-based tasks like lipreading [102] or action segmentation [103].

The model was first trained on NDC-ME either from scratch with randomly initialised weights or initialised on pre-trained ResNet and MS-TCN with lipreading data and finetuned on the MS-TCN layers only. The second step consists of evaluating the trained model on either NDC-ME (dataset used for training) or IFADV (dataset unknown to the model). Input data is preprocessed into a mono-channel audio signal of 1.2s sampled at 16kHz. Each training session lasts 80 epochs, with a batch size of 16 and a decreasing learning rate starting at 3×10^{-6} since our dataset has limited size.

Table 6.1 shows the results on Precision, Recall and F1-score metrics. Multiple observations can be made on the results. First none of the metrics reach 0.5 (with the highest score of 0.494) while being above random (0.333 with three classes). Second NDC-ME evaluation reaches better results than IFADV on all three metrics. Last finetuning slightly enhances the classification performance on both datasets, with higher improvements on IFADV. Our speech-only model is either not trained or powerful enough to capture relevant feature for the task at hand and reach impressive results. Probable causes are the training data quality and quantity that would lead to improper generalisation or the size of the model is too small. An other possible cause is the type of expressions evaluated: a smile is closer to a neutral expression than a laughter when only sound is available.

Table 6.1. Precision, Recall and F1-score for speech evaluation on NDC-ME and IFADV. The best results for each metric are in **bold** font.

	NDC-ME		IFADV	
	From scratch	Finetuned	From scratch	Finetuned
Precision (↑)	0.483	**0.494**	0.343	**0.372**
Recall (↑)	0.477	0.476	0.369	**0.438**
F1-score (↑)	0.480	**0.484**	0.355	**0.402**

Heatmaps in Figure 6.2 represent the distribution of each intensity in the model predictions. During training, only expression classes were provided as labels without any intensity information. Our goal was to determine if the expression intensity was linked to the better detection of said expression. NDC-ME heatmaps show that smiles are confused with neutral expression while laughter is better discriminated. There is no significant classification results between training methods which is coherent with Table 6.1. IFADV evaluation shows how every class is confused with the other two, especially when the model is trained from scratch. We can highlight how laughter are better detected when the intensity is high, for three out of four presented results.

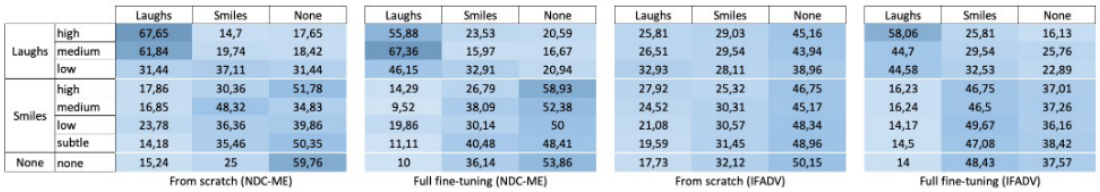


Figure 6.2. Intensity Heatmaps for audio analysis. From left to right: models trained from scratch and finetuned, evaluated on NDC-ME and on IFADV. At row i column j, the colour in shades of blue shows the percentage of expression/intensity i being predicted as expression j, with light blue being 0% and dark blue 100%. Values within a row adds up to 100%.

6.3.2 Face analysis

The video model architecture (lower branch in Figure 6.1) and the training methodology is similar to the audio model. The input data are first preprocessed in videos of 60 frames. Each frame contains the ROI around the mouth in a square of 96×96 pixels in RGB. It is composed by the ResNet-18 backbone followed by MS-TCN. The difference with the audio model appears before the ResNet, at the high-dimensional projection layer. It is made of a single-layer 3D-CNN instead of 1D to extract simultaneously temporal and spatial information. The model was first trained on NDC-ME either from scratch with randomly initialised weights or initialised on pre-trained ResNet and MS-TCN with lipreading data and finetuned on all layers. The second step consists of evaluating the trained model on either NDC-ME (dataset used for training) or IFADV (dataset unknown to the model). Each training session lasts 80 epochs, with a batch size of 16 and a decreasing learning rate starting at 3×10^{-6} since our dataset has limited size.

Table 6.2 shows the results on Precision, Recall and F1-score metrics of both datasets evaluation. On the one hand there is no significant gap in performance between from scratch and finetuned training on NDC-ME. On the other hand the zero-shot evaluation on IFADV shows better results when the model is fine-tuned on lipreading data than from scratch. As for the audio modality the model evaluation made on the same dataset as training shows higher efficiency, highlighting the better generalisation. The classification score on NDC-ME shows promising performance reaching f1-score as high as 0.693. When we compare with speech analysis we see that our visual model is able to extract better features to discriminate between smiles, laughter and neutral.

The intensity distributions across predicted expressions when the visual modality is evaluated on either NDC-ME or IFADV are shown in Figure 6.3. As for the audio analysis, only expression classes were provided as labels during training, without any intensity information. On the one hand the NDC-ME evaluation shows that confusion arises between higher intensities of smiles and lower intensities of laughter. The intuition

Table 6.2. Precision, Recall and F1-score for face expression evaluation on NDC-ME and IFADV. The best results for each metric are in **bold** font.

	NDC-ME		IFADV	
	From scratch	Finetuned	From scratch	Finetuned
Precision (↑)	**0.712**	0.707	0.399	**0.414**
Recall (↑)	0.674	**0.679**	0.432	**0.477**
F1-score (↑)	0.692	**0.693**	0.415	**0.443**

behind that observation is that the vision model can "see" smiles and laughs happening, but struggles to distinguish between them without audio. Medium and low smiles as well as neutral expression have a high recognition rate ($> 70\%$), while the subtle smiles are mostly confused with neutral. The subtle class was made due to the uncertainty of the annotators between smiles and neutral, hence the visible confusion for the model. On the other hand the IFADV evaluation shows low detection rate in most classes except for high laughter and neutral. Finetuned model has a smaller gradient in the evolution of intensity-wise detection, which leads to the better results showed in Table 6.2.

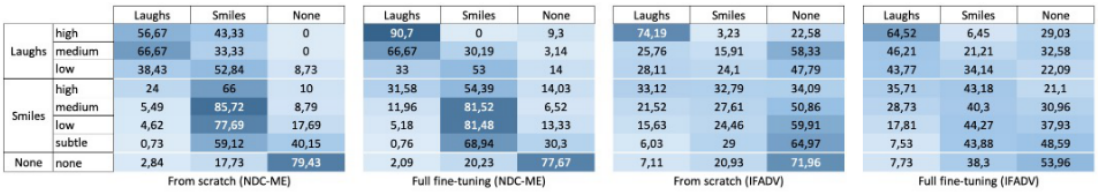


Figure 6.3. Intensity Heatmaps for face analysis. From left to right: models trained from scratch and finetuned, evaluated on NDC-ME and on IFADV. At row i column j, the colour in shades of blue shows the percentage of expression/intensity i being predicted as expression j, with light blue being 0% and dark blue 100%. Values within a row adds up to 100%.

6.3.3 Multimodal emotion analysis

Our multimodal approach fuses speech and face latent spaces together when both models are already trained. We perform late fusion (middle part of Figure 6.1), in opposition to early and mid fusion: the latent spaces fused are those of the last layer of each model, while early fusion concatenates input before the model backbone and mid-fusion concatenates features from intermediate layers. Each fusion strategy has its pros and cons as discussed in Chapter 3. We selected late fusion for the sake of simplicity, our goal was to analyse how even a simple 2-layer MLP on top of our fused latent space would impact the results. We combined the models trained from scratch in one pipeline and the

finetuned models in another one, considering the training scheme as the main criteria. Training in those scenarios lasted 30 epochs with a batch size of 16 and a decreasing learning rate starting at 3×10^{-6}. While other model configurations were experimented, they do not showcase any performance worth reporting.

Table 6.3. Precision, Recall and F1-score for fused modalities evaluated on NDC-ME and IFADV.

	NDC-ME		IFADV	
	From scratch	Finetuned	From scratch	Finetuned
Precision (↑)	**0.615**	0.602	0.384	**0.499**
Recall (↑)	**0.783**	0.714	0.385	**0.464**
F1-score (↑)	**0.690**	0.653	0.384	**0.481**

Results in Table 6.3 shows the evaluation of both model training scheme evaluated on either NDC-ME or IFADV. We observe how the combination of both modalities trained from scratch reaches a higher F1-score than that of finetuned models (more than 3.5%). This mainly comes from a higher recall which represents how the model is able to retrieve more expressions. Compared to face analysis, fusion models reach a lower F1-score but a higher one than speech analysis. Our intuition is that fusion works as a trade-off between both modalities. Zero-shot evaluation on IFADV shows that fusing finetuned models is better to generalise across set-ups than both modalities separately (improvement of 4 %) while models trained from scratch shows the same trade-off behaviour than NDC-ME evaluation.

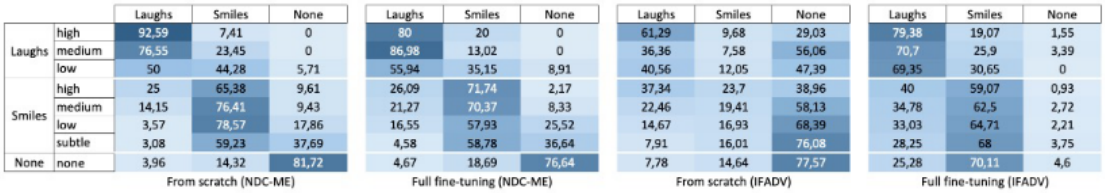


Figure 6.4. Intensity Heatmaps from the multimodal approach. From left to right: Fusion models trained from scratch and finetuned evaluated on NDC-ME and fusion models trained from scratch and finetuned evaluated on IFADV. At row i column j, the colour in shades of yellow/green shows the percentage of expression/intensity i being predicted as expression j, with yellow being 0% and dark green 100%. Values within a row adds up to 100%.

The detection rate of each class with respect to their intensity is depicted in the four heatmaps of Figure 6.4. The first two show the fusion models evaluated on NDC-ME and their ability to recognise higher intensity laughter and medium intensity smiles. Two

confusion zones are distinct: the first at the junction between smiles (high intensity) and laughs (low intensity) and the second with subtle smiles and no expression. The two other heatmaps show the evaluation on IFADV in which there is more confusion since the dataset has different recording specifications. While no significant difference is observed, a progressive evolution is still observable in the distribution of laughter and smiles when the fused models are finetuned while the neutral class is not considered.

6.3.4 Discussions

Firstly it is clear that not one model performed better than all the others in all categories. But by considering overall results, we can argue that, when training and testing on the same dataset, models fusion trained from scratch performs relatively well on all classes, even better than the finetuned visual model which, interestingly, seems to confuse low level laughs with smiles. The fusion model seems able to keep the overall good performances of the visual modality while improving the bad ones. It is worthy to note though, that in this work, a simple fusion mechanism and training were applied. Improving the fusion scheme by modifying the number of layers or exploring early- and mid-fusion might take better advantage of both modalities.

We can also note that audio laughs, when misclassified, are most often confused with smiles, especially low-level laughs which is an interesting point suggesting that a relationship might exist even in the audio modality. However, audio classification does not perform as well on smiles, for either evaluation datasets. It is true that the smiles true positives are quite high but so are the false positives represented by the None being misclassified as smiles. On an intuitive level, this makes sense. Indeed, although smiles have been shown to be audibly recognisable, smiled speech is more a change of voice than a burst of affect as is laughter, which makes it more complicated to discriminate from non-smiling speech, especially with the limited amount of data at our disposal. The audio modality seems to perform rather well on laughter, but the smile misclassification leads to poor metrics value. The visual models seem to perform overall better for the smiles than audio modality. This also intuitively makes sense since an obvious discriminating feature between smiles and laughs is the audio cue. One limitation of the visual models is the use of only lips, while other feature like head motion can be important in laugh expressions. Nevertheless the models also seem to perform rather well on laughs especially when finetuned, this is probably due to the physical movements accompanying the laughs that are less present when smiling.

Some interesting notes can also be taken concerning the fusion steps. First, fusion surprisingly seems to work better when fused models were trained from scratch than finetuned. This fusion of models trained from scratch seems to allow the model to use the best prediction of both modalities in one system by improving the recall at the cost of a decrease in precision. Another interesting observation regarding finetuning in general is that it improves laughter classification and generalisation (when applied to IFADV data) in all cases. It also seems to improve true positives in smiles detection but at the

expense of the false positives, represented by the confusion of None with Smiles. For the visual modality, finetuning seems to improve the performance of the models both for smiles and laugh detection and its generalisability most of the time which is observed on the results of the models on the IFADV data. The only slight deterioration that we can observe is that more None samples are confused for smiles than in the model trained from scratch. We can deduce that finetuning allows a model to use the knowledge gained from prior training on speech or lip-reading data to increase its robustness to other datasets.

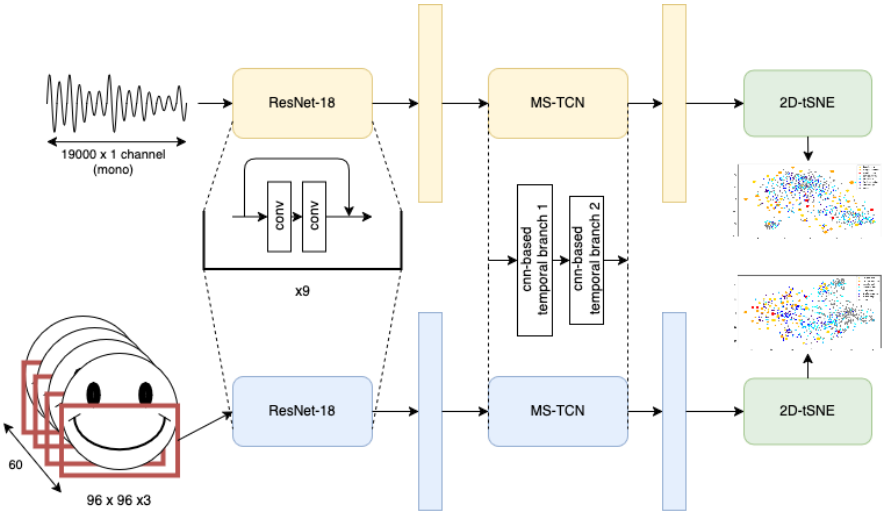
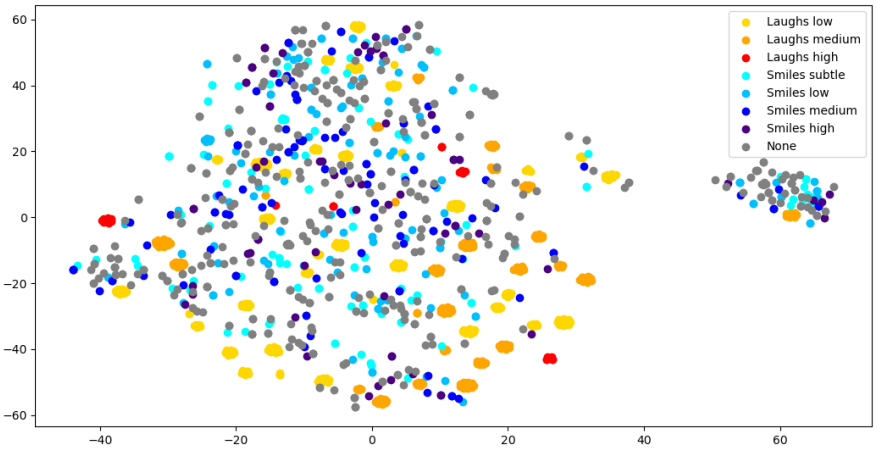
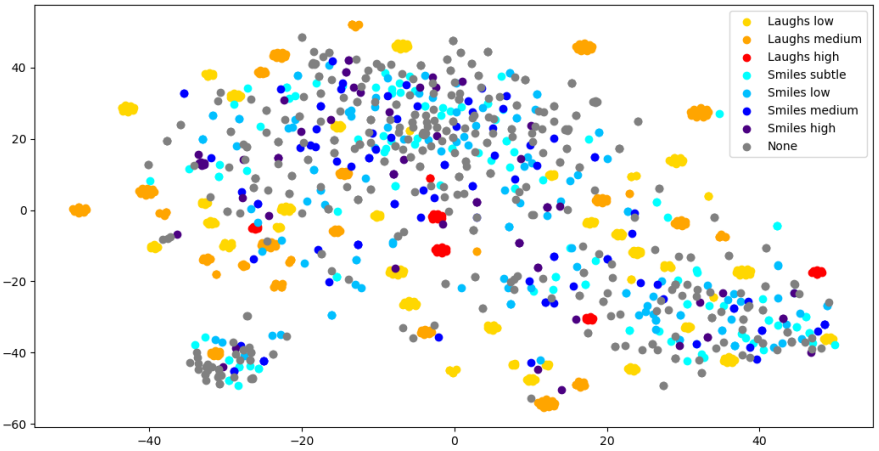


Figure 6.5. t-SNE dimensional reduction applied to each modality.

With the goal to get a better understanding of the models' data representation especially on the impact of finetuning, we present a visualisation of the ending layers of the models. For this, we extract embeddings from MS-TCN last layer. We then apply a t-SNE [104] method to reduce the embeddings dimensions to a two-dimensional space while retaining the most relevant features. The results on the audio modality are shown in Figures 6.6(a) and 6.6(b) and those on the visual modality in Figures 6.7(a) and 6.7(b). For both modalities, we can see that finetuning allows to better discriminate all three classes. Audio laughs (shades of orange on the figure) are pushed at the extremities of the pattern, while smiles are still rather mixed with None class, which is coherent with the results presented above. Another observation can be made on the visual data: we can clearly see the laughs being pushed at the left of the pattern, the low level laughs (yellow dots) tend to also be present in the centre of the pattern, the higher level smiles (darker blue) tend to be more mixed with the laughs and lower level smiles (lighter blue) with the None - all coherent with our observations made above. An analysis of the results with respect to intensity levels shows that the system tends to learn implicit knowledge of those levels from the data. Apart from the high level laughs, levels on the extremes seem to be more often confused by the models than the medium ones. Low level laughter are in general mostly confused as smiles and the high levels of smiles (medium and high)

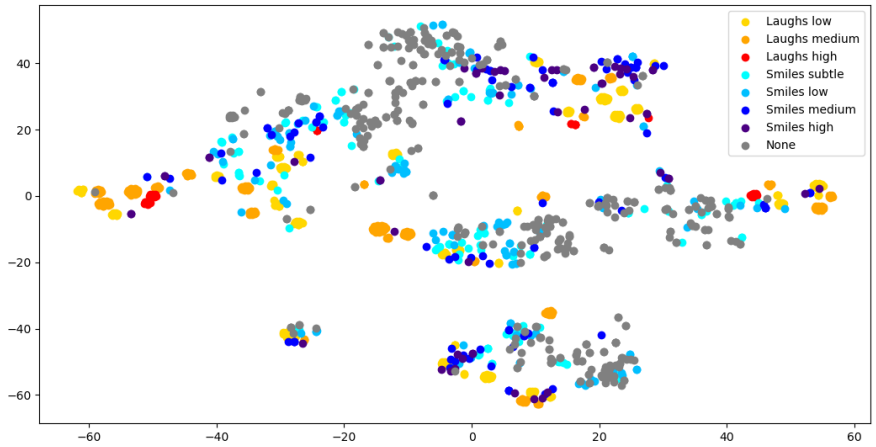


(a) Model trained from scratch. We can hardly distinguish between Smiles and No expression. Laughs, while mixed with others, are gathered at the low-right edge of the cluster.

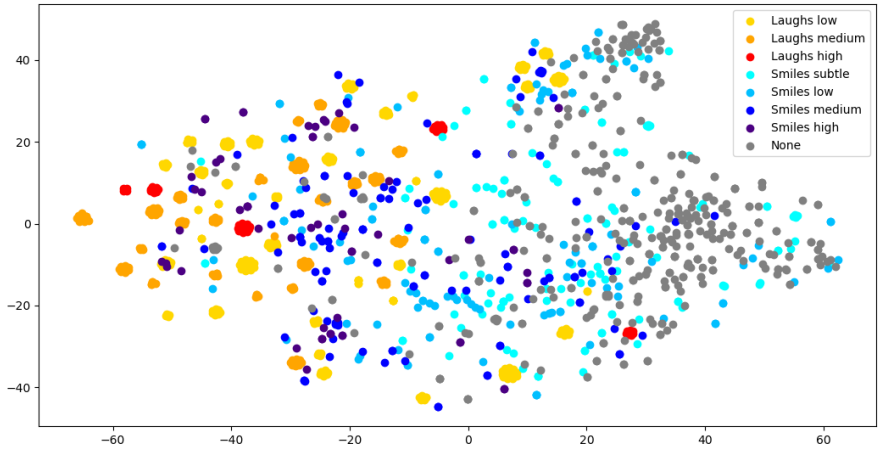


(b) Finetuned model. While the Smiles and No expression are grouped in the middle of the shape, Laughs are distributed across the outer perimeter.

Figure 6.6. 2D t-SNE representations of audio models. Axis dimensions have no physical significance.We observe the distribution of expressions with respect to their intensities: yellow/red shades stand for Laughs, blue shades for Smiles and Grey for None, with darker shades for higher intensities.



(a) Model trained from scratch. While no particular shape is visible, we see that grey dots stay at the center of the plot, and ohter expressions are grouped around the edge.



(b) Finetuned model. We see a linear evolution following the x-axis. From left to right: High laughs to No expression with Smiles in-between (higher than lower intensities).

Figure 6.7. 2D t-SNE representations of visual models. Axis dimensions have no physical significance. We observe the distribution of expressions with respect to their intensities: yellow/red shades stand for Laughs, blue shades for Smiles and Grey for None, with darker shades for higher intensities.

are mostly confused as laughs while the low levels (subtle and low) are mostly confused as being None (which, as we could observe in our dataset, contains a majority of neutral expressions or speech).

These observations can be seen in almost all the presented results from the visual modality. This confusion by models are intuitive to us: although they were not given any information about the intensity levels of the expressions during training, the models seem to have more difficulty with some intensities on the extreme levels than with others. In the visual modality, if we revised the current results by considering the samples classified as higher levels of smiles (medium and high) as laughs and lower level smiles (subtle and low) as None (thus having only 2 classes at the end instead of 3), the data would be correctly classified as laughs at an average 69.15% with an standard deviation of 5.58% compared to the current average rate 66.46% with an standard deviation of 10.06%. We assume that this is due to the nature of the expressions themselves, since the features representative of some intensities in one expression can be shared with features in another expression (high level smiles and laughs can both show pulled lip corners and raised cheeks for instance). We can therefore mainly conclude that:

1. Fine-tuning is beneficial for performance and generalization in most cases and should be considered instead of training a model from scratch.
2. Given all the observations and analyses made regarding the intensity levels, the relationship between smiles and laughs is not as simple as a binary or single class relationship. A more complex relationship should therefore be considered when dealing with these expressions

Finally, we suspect that some aspects of the dataset to have probably influenced negatively some of the results. First, some aspect is that some files contain speech coming from the interlocutor of the concerned subject, overlapping the subjects laughter. More accurate detection could be achieved by removing those artefacts from the dataset. A second drawback is that some of the annotations contain subjectivity due to the limited number of annotators and this can make the annotations more sensitive to human error.

6.4 In Brief

Summary for Chapter 6

- We studied deep learning-based classifiers that distinguish smiles and laughs as two distinct facial expressions.
- Our analysis covered models originally designed for word recognition and lipreading, repurposed for smiles and laugh detection using audio, visual and audio-visual fusion modalities.
- We demonstrated that fusion outperforms single-modality systems and that fine-tuning improves generalisation especially when testing on different datasets.
- Our post-hoc analysis showed that even without explicit training on intensity labels, model behavior varies significantly across intensity ranges.

Perspectives for Chapter 6

- We intend to investigate on fusion approaches and their effect on classification, as well as other deep learning methods to better highlight the complex relationship between smiles and laughs.
- Other types of DL approaches can be considered to improve result or allow for larger regions of interest. In the next chapter we present a Transformer-based model evaluated on Smiling and Laughter detection.

Chapter 7

Focusing on Global Area: Face and Voice

Contents

This chapter is based on the following publication:

- Bohy, Hugo, Minh Tran, Kevin El Haddad, Thierry Dutoit, and Mohammad Soleymani. "Social-MAE: A Transformer-Based Multimodal Autoencoder for Face and Voice." In *2024 IEEE 18th International Conference on Automatic Face and Gesture Recognition (FG)*. IEEE, 2024.

7.1 Introduction

Human emotions and social behaviours are expressed and perceived through multiple modalities. While verbal communication can provide information on a person's communicative intent and emotions, non-verbal communication has shown to be equally or even more important [42]. Socially intelligent systems require multimodal methods allowing them to perceive human social and expressive behaviours. Understanding expressions and social behaviours can be achieved by analysing audiovisual modalities, i.e., face, body and voice. Although unimodal approaches, e.g., vision from facial expression or audio for tracking arousal, can reach a high performance [105, 106], fusing two modalities increases the efficiency and robustness of multimodal systems [27, 107] as shown in Chapter 6.

Despite the popularity of emotion and social behavior perception, datasets for such tasks are often limited in size due to the high cost of labelling. Most existing audiovisual methods are based either on transfer learning with models trained on out-of-domain data, e.g., AudioSet [108], or trained from scratch. However, the desired input data should contain human faces and voices. Existing audiovisual encoders, e.g., [109], also lack the temporal fidelity in the visual domain. In contrast, expressive behaviours in the human face are rather dynamic and fast-moving. There is limited work, e.g., [110], on audiovisual encoders suitable for the automatic perception of human emotional and communicative behaviours.

In this chapter, we present Social-MAE, a pre-trained audiovisual model based on Masked Autoencoder. We aim to adapt a self-supervised method with superior results on audio event recognition for the audiovisual understanding of human social behaviours. We evaluate our model against several baselines on three different social and affective tasks: emotion recognition, laughter detection and apparent personality estimation. The main contributions of this work are as follows.

- We present Social-MAE, a model based on CAV-MAE architecture adapted to affective context by pre-training on a large-scale social dataset;
- To develop Social-MAE, we modify CAV-MAE to accept multiple frames providing higher temporal fidelity at visual input;
- Our experiments demonstrate the importance of in-domain pre-training for affective and social tasks. Our model reaches or outperforms SOTA models on relevant tasks.

7.2 Related Work

Past work extensively explored the natural interactions between audio and visual signals for representation learning [111–117] through self-supervision with a variety of pretext tasks. Synthesis-based strategies [111, 112, 117] have been proposed, where audio and visual signals are artificially combined to facilitate learning cross-modal associations.

Alignment-based methods [113, 118–120], on the other hand, focus on aligning signals from both modalities in time or space, aiming to extract meaningful correlations between them. Another line of research involves the application of masked autoencoding (MAE) [121], where the model learns to reconstruct the missing portions of either the audio or visual input, fostering representation learning through learning the structure of the data.

Recently, two models, namely MAViL [122] and CAV-MAE [123], have explored the combination of MAE with contrastive learning and demonstrated state-of-the-art (SOTA) performance on audio-visual classification. Adding contrastive learning allows the models to learn inter-modality representations.

7.3 Datasets

We present the datasets that were used for either pretraining or inference on affective tasks. While they were described in Chapter 5, we provide additional information about the dimensions and the preprocessing applied.

VoxCeleb2 VoxCeleb2 dataset is a large-scale audio-visual dataset primarily focused on speaker recognition. We resize every frame to 224 pixels by 224 pixels. Default mean value for each channel is (0.4850, 0.4560, 0.4060) and standard deviation (0.2290, 0.2240, 0.2250) for Red, Green and Blue respectively. These values were calculated beforehand and are the average values on the whole dataset. Based on that, we set the mean to 0 and standard deviation to 1 for both audio waveform and video frames. The size of the available data for training and validation together reaches up to 70 hours and 8 hours for the test partition, reduced by the unavailability of some videos on Youtube.

CREMA-D CREMA-D provides a diverse collection of audio and video recordings featuring actors conveying a wide range of emotions. Default mean and standard deviation for each channel have been found to be similar to these of VoxCeleb2. Based on that, we set the mean to 0 and standard deviation to 1 for both audio waveform and video frames. The total duration of the dataset is 2.5 hours.

First Impressions The First Impressions dataset [69] comprises audio-visual clips extracted from YouTube videos of people facing a camera. Default mean and standard deviation for each channel have been found to be similar to these of VoxCeleb2. Also based on that, we set the mean to 0 and standard deviation to 1 for both audio waveform and video frames.

NDC-ME NDC-ME, described in Chapters 5 and 6, is also processed to have a zero mean and standard deviation of 1. The average default mean and standard deviation are

(0.5452, 0.3590, 0.2160) and (0.2193, 0.1876, 0.1488) for Red, Green and Blue channels respectively. Based on that, we set the mean to 0 and standard deviation to 1 for both audio waveform and video frames.

7.4 Method

We describe Social-MAE (Fig. 7.1), an adapted version of CAV-MAE that focuses on voice and face. The model is composed of two modality-specific encoders followed by a joint encoder module and a joint decoder module. Each module relies on a set of Transformer layers [16] made of an attention block, a feed-forward network, residual connections and layer normalization [124]. We describe the pre-processing pipeline in Section 7.4.1, the model overview in Section 7.4.2 and the self-supervised training in Section 7.4.3.

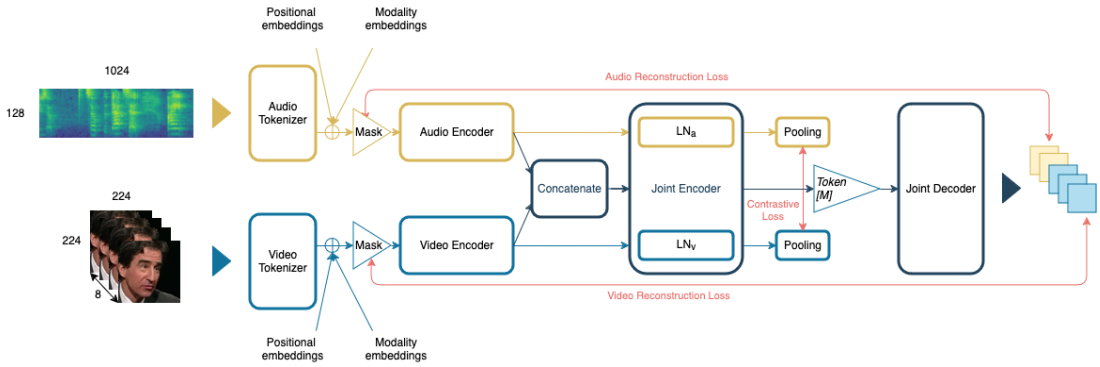


Figure 7.1. Social-MAE model for voice and face analysis in videos. The model is pre-trained to reconstruct audio and visual modalities from masked portions of their corresponding input, narrowing the difference between each modality representation.

7.4.1 Audiovisual Tokenization

The architecture follows a mid-fusion scheme: both audio and video are first encoded in two separate branches for several encoder layers before merging into a joint encoder. Audio data are pre-processed as in CAV-MAE: we convert the input audio waveform into a sequence of 128-dimensional log Mel filterbank features computed with a 25 ms Hamming window and an overlap of 10 ms. We pad or crop the length of the input to keep 1024 audio frames, resulting in a 128×1024 spectrogram. The spectrogram is processed as an image that we split in N 16×16 non-overlapping patches. Each patch is projected with a linear layer to a 1-dimensional embedding of size 768, referred to as a *token*. We add a trainable positional embedding to each token to encode information about the token order.

Visual inputs differ from CAV-MAE as they consist of eight randomly selected frames as proposed in [122] rather than single frame. Each frame is an RGB image of the face bounding box scaled to 224×224 pixels, resulting in a $8 \times 224 \times 224 \times 3$ video input. We split the video into N $2 \times 16 \times 16$ patches with no overlap, flatten and project with a linear layer into tokens of size 768. A trainable positional embedding is added to each token as well. Another trainable parameter provides information about each token's modality and weights the modality's importance. After adding positional and modality embeddings, a random mask with a rate of p% is applied to the input tokens, providing the model only with (1-p)% of the original audio and/or video sequence.

7.4.2 Model Description

We present an overview of the autoencoder architecture as described in [123]. The model first processes an input sequence in separate encoders, each leveraging unimodal information. The modality encoders are stacks of 11 Transformer layers that aim to encode internal patterns in the input sequence. The joint encoder comprises a single Transformer layer on top of the modality encoders. Each modality is processed by the respective encoder followed by the joint encoder either individually or concatenated with the second modality depending on the targeted loss. The layer normalization on top of the joint encoder differs for audio, video and multimodal processing. It is trained in a three-pass scheme: the first pass with only the audio tokens and audio-specific layer normalization, the second with only the video tokens and video-specific layer normalization, and the third is the concatenation of audio and video tokens in a single sequence. The weights of the joint encoder are shared regardless of its input modality, as it was shown that weight sharing lightens the model without degrading performance [125]. The unimodal tokens are averaged following the average pooling method, while the multimodal tokens are fed to the joint decoder, which is a stack of 8 Transformer layers. It aims to retrieve the original video and audio from an input sequence made of the encoded tokens and a learnable token M repeated at masked positions. The reconstruction loss is described later in the chapter.

7.4.3 Self-Supervised Pre-Training

We adapted the pre-trained CAV-MAE model by training with self-supervision on the VoxCeleb2 dataset [63]. As self-supervised pre-training often requires vast amounts of data, we chose VoxCeleb2 (Chapter 5), as a suitable large and diverse audiovisual dataset with social content.

The learning phase relies on the weighted combination of contrastive and reconstruction loss, each providing complementary information. For an input sequence of N pairs of audio and video tokens a_i, v_i, the contrastive loss $\mathcal{L}_c$ is computed on modality averaged

(a) Reconstruction on CREMA-D.

(b) Reconstruction on First Impressions.

Figure 7.2. Randomly selected Social-MAE visual zero-shot reconstruction on (a) CREMA-D and (b) First Impressions datasets. The first row shows the original input, the second row the visual equivalent to masked tokens, and the last row the reconstructed frames.

tokens c_i^a, c_i^v defined as:

$$c_i^a = MeanPool(E_j(E_a(a_i))) \tag{7.1}$$
$$c_i^v = MeanPool(E_j(E_v(v_i))) \tag{7.2}$$

where $E_m(.)$ is the encoder of modality m and $E_j(.)$ the joint encoder. $\mathcal{L}_c$ aims to leverage relevant inter-modal information by following a LogSoftmax loss:

$$\mathcal{L}_c = -\frac{1}{N}\sum_{i=1}^{N}\log\left(\frac{exp(f(ca_i, cv_i))}{\sum_{j=1}^{N} exp(f(ca_i, cv_j)}\right) \tag{7.3}$$

with $f(ca_i, cv_j) = \frac{\|c_v^i\|^T \|c_a^j\|}{\tau}$ and τ the temperature. This aims to narrow the difference between audio and video tokens from the same file and increase that of different files. This efficiency of the loss depends on the mini-batch size N.

We pad the decoder input sequence x_i with a learnable token $\mathbf{M}$ at the masked positions.

$$x_i = E_j([E_a(a_i), E_v(v_i)]) \tag{7.4}$$
$$x_i' = concat(x_i, M) \tag{7.5}$$
$$y_i' = D_j(x_i') = concat(\widehat{y}_i, \widehat{M}_i) \tag{7.6}$$

where $D_j(.)$ is the decoder, $\widehat{y}_i$ and $\widehat{M}_i$ the reconstructed tokens. The reconstruction loss $\mathcal{L}_r$ evaluates the model ability to reconstruct the masked tokens x_i^{mask} from the tokens at the output of the decoder $\widehat{M}_i$ with an MSE loss defined as:

$$\mathcal{L}_r = \frac{1}{N}\sum_{i=1}^{N}[\frac{\sum(\hat{a}_i^{\text{mask}} - a_i^{\text{mask}})^2}{|a_i^{\text{mask}}|} + \frac{\sum(\hat{v}_i^{\text{mask}} - v_i^{\text{mask}})^2}{|v_i^{\text{mask}}|}] \tag{7.7}$$

This loss aims to force the model into learning internal patterns by trying to reconstruct only masked tokens with only $(1-p)\%$ of the input tokens available.

The final loss is the weighted sum of the contrastive and the reconstruction losses: $\mathcal{L} = \mathcal{L}_c \cdot \lambda_c + \mathcal{L}_r$.

7.5 Experiments and Results

We pre-trained our Social-MAE during 25 epochs with a learning rate starting at 10^{-4} and decreasing at a decay rate of 0.5 every 5 epochs with a masking ratio p=75%. For comparison, we also pre-trained CAV-MAE (as it uses 1 frame instead of 8) following the same settings. Both models were initialized on CAV-MAE^{scale+} weights pre-trained on AudioSet-2M with self-supervision. We report visual zero-shot reconstruction in Fig. 7.2 using pre-trained Social-MAE on two downstream task datasets: CREMA-

D [71] and First Impressions (FI) [69]. The model is able to provide a convincing output on previously unseen data. Most reconstruction errors, although not obvious at first sight, come from the most dynamic areas of the face, such as the eyes or lips. Table 7.1 reports the reconstruction losses on the three downstream tasks evaluated later in this work. CAV-MAE has similar results for audio reconstruction, but our multi-frame method is almost four times better at visual reconstruction.

Table 7.1. Zero-shot Audiovisual reconstruction losses on CREMA-D, First Impressions and NDC-ME. Best results are in **bold**.

	CREMA-D		FI		NDC-ME	
	A.	V.	A.	V.	A.	V.
CAV-MAE	**0.014**	0.097	0.016	0.124	0.104	0.122
Social-MAE	0.016	**0.028**	**0.01**	**0.033**	**0.1125**	**0.0169**

For downstream tasks, we remove the decoder from the architecture and replace it with a randomly initialized linear layer. We evaluate our pre-trained model by fine-tuning it on three different social and affective tasks: emotions recognition on CREMA-D, personality traits regression on First Impressions and smiles and laughter detection on NDC-ME. For each task, we describe the dataset, the fine-tuning pipeline and the evaluation metrics to compare CAV-MAE and Social-MAE models against published baselines, following their experimental settings for consistency.

7.5.1 Emotion Recognition

Experimental setup This task is evaluated on CREMA-D, desribed in Chapter 5. Fine-tuning requires no masking on audio and visual tokens. We fine-tune pre-trained Social-MAE as well as our pre-trained version of CAV-MAE for 20 epochs using a mini-batch size of 8, learning rates at 10^{-4} and 10^{-5} for the encoders and the head respectively and we use the Cross-Entropy Loss.

Baselines

UAVM [109] presented UAVM, a unified audiovisual framework for classification. The model uses pre-trained CNN-based feature extractors on log Mel filterbanks and multi-frame visual inputs that are fed to Transformer layers.

AuxFormer [126] proposed AuxFormer, a multimodal model that fuses audio and visual tokens through Transformer inputs. The model also processes separate modalities

through auxiliary networks. The model loss is a weighted combination of the network losses. Audio inputs are low-level descriptors from OpenSmile [127] toolkit, and visual inputs are face clips processed by pre-trained VGG-face architecture [128].

VAVL [129] proposed an audiovisual model, named Versatile AudioVisual Learning (VAVL), which relies on the Conformer architecture [130]. Each modality input flows through a separate encoder followed by a shared-weight conformer. Audio inputs are high-dimensional features from Wav2vec2.0 [131] and visual inputs are face clips processed into emotional feature representations.

Table 7.2. F1-score performance Comparison on CREMA-D. Mi and Ma refer to F1-score Micro and Macro. The best results are in **bold** face font. * p-value < 1E-5

	Audio		**Visual**		**AV**	
	Mi	**Ma**	**Mi**	**Ma**	**Mi**	**Ma**
AuxFormer [126]	0.648	0.593	0.626	0.560	0.763	0.698
UAVM [109]	0.554	0.614	0.672	0.617	0.769	0.749
VAVL [129]	**0.701**	0.628	**0.787**	0.738	0.826	0.779
CAV-MAE	0.694	**0.694**	0.630	0.635	0.766	0.759
Social-MAE	0.601*	0.607*	0.749*	**0.755***	**0.837***	**0.842***

Results and discussion Table 7.2 reports the F1-score with micro and macro averaging techniques. Social-MAE outperforms previously published methods for audiovisual classification. The micro F1 score shows the global accuracy, and the macro F1 score shows the unweighted average accuracy across each class, so the macro F1 score can be influenced by class imbalance. Since classes in CREMA-D range from 763 utterances (Sadness) to 2204 utterances (Neutral), we interpret the similarities between the macro and micro F1-scores reached by our pre-trained models as their ability to recognize emotions regardless of their prevalence.

Adapted CAV-MAE competes for best audio-only classification against VAVL model. Social-MAE rivals the best baseline for visual classification. The performance is impressive when you consider that the former processes 8 frames and the latter processes high-level features from all input frames. We also find it interesting that adapted CAV-MAE is able to outperform multi-frame baselines AuxFormer and UAVM on both unimodal and multimodal classification tasks, highlighting the efficiency of in-domain self-supervised pre-training.

7.5.2 Personality Trait Prediction

Experimental setup We evaluate personliaty prediction with the First Impressions (FI) dataset, a collection of 10,000 *in-the-wild* videos, in average 15s long. Videos are annotated with apparent personality traits known as *big-5* [56]: Openness, Conscientiousness, Extraversion, Agreeableness and Neuroticism. Fine-tuning requires no masking on audio and visual tokens. We fine-tuned both models presented in Sec. 7.4.3 for 10 epochs using a mini-batch size of 8, an encoder learning rate of 1e-4 and the classification head learning rate of 1e-5. We use a Mean Absolute Error loss and our accuracy metric is 1 − Mean Absolute Error.

Baselines We compare our fine-tuned CAV-MAE and Social-MAE to the best team of the challenge associated to the dataset:

DCC DCC [132] reaches the third place with randomly initialised ResNet backbones for each modality (single frame or audio) and a fully-connected layer on top as prediction head. The training session lasted for 900 epochs with a mini-batch size of 32.

evolgen evolgen [69] reaches the second place using MFCC features and CNN-based deep representations of one frame as audio inputs and visual inputs respectively. The model is composed of LSTM [133] layers followed by a fully-connected layer for trait prediction. The training lasted 1200 epochs with a mini-batch size of 128.

NJU-LAMDA NJU-LAMDA [134] is a model pre-trained on VGG-face. The audio input is log Mel filterbank and the visual input is the deep features from 100 frames. Authors train their model in 100 epochs for the audio stream and 3 epochs for the pre-trained visual stream, with a mini-batch of 128.

Table 7.3. Model Accuracy on First Impressions Dataset. Best results are in **bold** face font. * p-value < 1E-5.

	Ope.	Con.	Ext.	Agr.	Neu.	Avg.
DCC [132]	0.911	0.914	0.911	0.910	0.909	0.911
evolgen [69]	0.912	0.912	0.915	0.912	0.910	0.912
NJU-LAMDA [134]	**0.912**	**0.916**	**0.913**	**0.913**	**0.910**	**0.913**
CAV-MAE	0.899	0.899	0.899	0.902	0.896	0.899
Social-MAE	0.908*	0.902*	0.895*	0.907*	0.905*	0.903*

Results and discussion Table 7.3 shows the accuracy of each personality trait on ChaLearn First Impressions dataset as well as the mean accuracy. Social-MAE shows a performance of 90.32% on average. While the accuracy is lower than the baseline, it remains impressive considering it was trained for only 10 epochs and with a smaller mini-batch size. We can also observe that processing multiple frames simultaneously (Social-MAE) demonstrates better regressions on four out of five traits compared to the single frame method (CAV-MAE).

7.5.3 Smiles and Laughter Detection

Experimental setup The Naturalistic Dyadic Conversation on Moral Emotions (NDC-ME) dataset contains 8,352 clips of interactions in English of participants from different backgrounds. Each clip lasts 1.22 seconds, is cropped around the face, and is annotated with non-verbal expressions of smile, laughter, and neutral. We fine-tuned for 10 epochs, with no masking strategy, a mini-batch of 8, and learning rates of 1e-5 and 1e-4 for the backbone and classification head, respectively. Our training objective is the Cross-Entropy Loss. The baseline for smile and laughter detection is LSN-TCN [84], our CNN-based architecture that processes embedded representations of audio and video input separately and feeds them to two fully-connected joint layers (Chapter 6).

Table 7.4. F1-score on NDC-ME. Best results are in **bold** face font. * p-value < 1E-5.

	Pre-training	Audio	Visual	Audiovisual
LSN-TCN [84]	Supervised	0.438	0.608	0.590
CAV-MAE	Self-Supervised	0.471	0.629	0.766
Social-MAE	Self-Supervised	**0.546***	**0.728***	**0.776***

Results and discussion Table 7.4 shows that both self-supervised methods reach higher F1-scores than the supervised baseline. Using multiple frames instead of one significantly improves the performance of the visual modality while slightly improving that of the multimodal classification. The poor results in audio-based classification can be explain by the modalities of each expression. While laughs are both audible and visually noticeable, smiles are mostly visual and can be confused with no expressions when audio classification is performed.

7.6 In Brief

Summary for Chapter 7

- In this chapter, we presented Social-MAE, our pre-trained audiovisual Masked AutoEncoder on audiovisual social data. We modified existing CAV-MAE to accept multiple frames on a large human social behavior dataset.
- We evaluated our model on three relevant downstream tasks, demonstrating its effectiveness in achieving state-of-the-art results in audiovisual emotion recognition with a 0.837 F1 score and laughter detection with a 0.776 F1 score.
- With this work, we demonstrated the significance of in-domain adaptation of a large multimodal model trained through self-supervised pre-training.

Perspectives for Chapter 7

- The proposed pre-trained encoder can be easily fine-tuned for other audio-visual social behaviour understanding tasks, enabling more robust and performant models for perceiving human behaviour.
- Building on contrastive learning, future research could focus on enhancing the independence of each modality while preserving their complementarity.

Chapter 8

Affect Disentanglement

Contents

Recent developments in large-scale models have significantly influenced the way most DL tasks are approached. Models offer strong performance and versatility, but their size and complexity present challenges, particularly for smaller research groups with limited resources. Training, fine-tuning, or even deploying such systems often demands infrastructure that is not readily available to all.

Some of these models are built with general-purpose objectives in mind, designed to manage a wide range of tasks within a single framework [5,18,135]. Multimodal systems, in particular, can process diverse input types such as audio, video, and text, enabling them to carry out Automatic Speech Recognition (ASR), Text-to-Speech (TTS), and affect analysis in parallel.

In this chapter, we investigate how large models encode affective information. Our assumption is that pre-trained models still contain latent social information such as speaker identity or emotional content. Our aim is to study whether such content is disentangled and discarded during pre-training and if it can be saved.

We suggest that many models tend to mix different types of information within their latent spaces. This entanglement may make generalisation across speakers, languages, or contexts more complex. Our approach explores ways to promote more independent internal representations, so that affect-related signals can be better isolated and used when appropriate.

8.1 Related Work

Disentangling affective information from other latent factors like speaker identity or lexical content has been a long-standing challenge in affective computing. The process of disentangling has been shown to be impossible without additional assumptions on the architecture and the data [136, 137]. In most systems, emotional features tend to be blended with other cues in the embedding space, which limits interpretability and generalisation. Our aim is to save some of the disentangled information that is lost in pretrained models.

Several approaches have tried to achieve this, either through architectural design or by applying specific training constraints. Multimodal approaches are rather complex to set up due to the variety of non-verbal cues to tackle. Peri et al. [138] proposed an audio-visual supervised method to disentangle emotion from speaker content based on multitask learning. Their approach involves an auxiliary branch to force disentanglement at the output of each encoder. Text and speech fusion have also been studied for supervised emotion disentanglement. Authors in [139] uses HuBERT [22] for speech content, MPNet features for text understanding and a joint decoder prior to the emotion classification. Ispas et al. [140] extends this work by replacing the text encoder with DeBERTav3 features and cross-modal attention decoders prior to emotion classification. In the context of speech-only, much of the recent progress comes from models trained for

ASR tasks. These models, including wav2vec2 [21], HuBERT, and Whisper, focus on lexical content, but their latent spaces also seem to capture speaker traits and emotional signals. To counter that, Qu et al. [141] suggested the use of two pretrained models to disentangle prosodic content: HuBERT and ECAPA-TDNN [142] for the semantic and speaker content respectively. While these methods have shown impressive results in emotion recognition tasks, they still requires supervised speaker knowledge to keep affective content only.

In contrast to prior work, we put more emphasis on how the information is represented internally. The objective is not so to improve performance, but rather to understand what social information is disentangled and lost within a set of input embedding. We aim to focus on how cleanly it can be isolated from other information without the need of simultaneous use of multiple pre-trained models for each content.

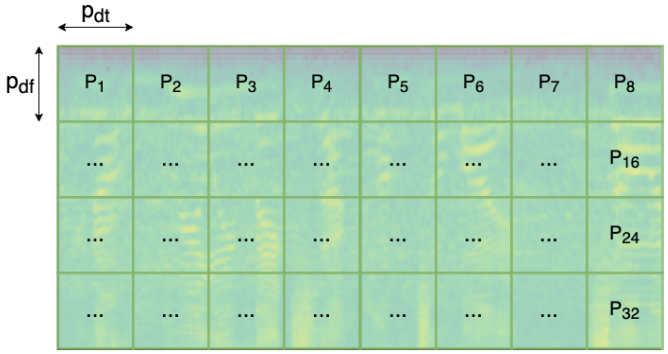
8.2 Methodology

We chose two pretrained models based their initial task: Audio Spectrogram Transformer (AST) [106] which was trained to recognise various sounds, and Whisper [4], a weakly supervised speech recognition system.

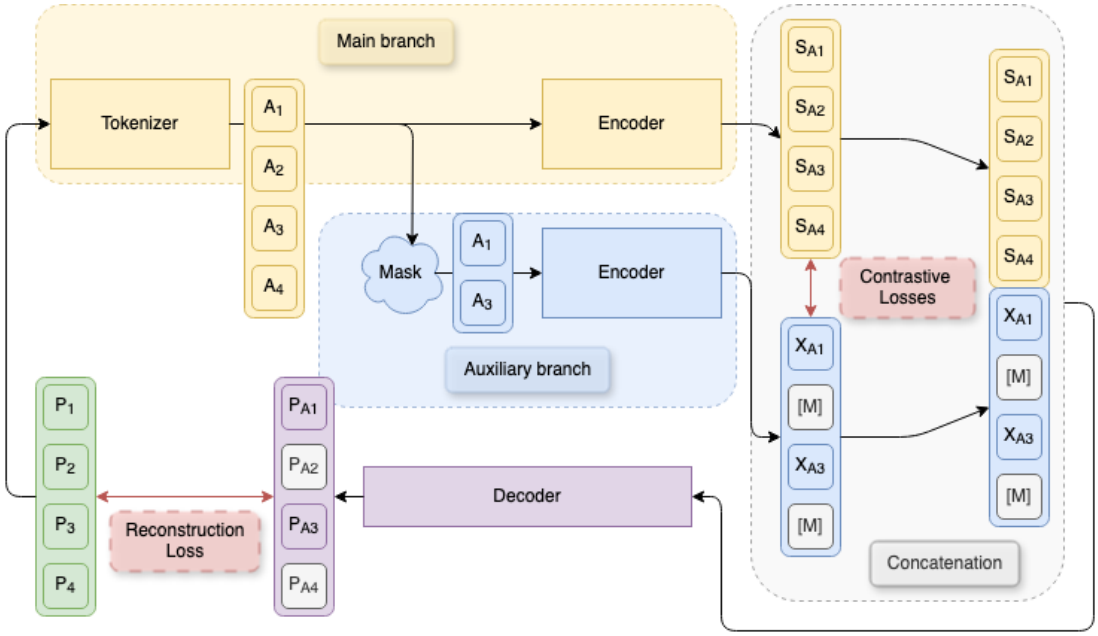
Our framework is as depicted in Figure 8.1: we extended the model into a dual-branch architecture to study disentanglement. The model processes audio input through two parallel encoders: one frozen (main branch) and one trainable (auxiliary branch). The goal is to extract complementary information in the auxiliary branch to reconstruct the original input.

The main encoder consists of the original tokenizer and transformer-based encoder layers. All components in this branch are frozen throughout training, including parameters such as positional encodings. This branch is expected to preserve the model original performance for its initial task. In parallel, the second encoder with the same architecture is initialised from the main model weights and trained to extract complementary features. This auxiliary encoder shares the same input embeddings and runs in parallel, without weight sharing. The outputs from both encoders are concatenated (late fusion) and passed through a joint decoder, made up of several transformer layers.

The training objective combines three loss terms. The first is a reconstruction loss, applied to the output of the decoder, which encourages the auxiliary encoder to preserve lost but useful information. The second is a mutual information penalty between the outputs of both encoders, designed to reduce redundancy and promote complementary representations. Finally, a contrastive loss is applied to align representations of the same sample across branches, while pushing apart representations of different inputs. This combination encourages a latent space that is shared when necessary, but structured enough to allow partial disentanglement.



(a) Patching approach. One patch covers p_{df} pixels on the frequency axis and p_{dt} on the time axis, with no overlap between patches.



(b) The main branch is depicted in yellow, the auxiliary branch in blue and the joint decoder in purple while losses are indicated in red.

Figure 8.1. Our proposed pipeline during pretraining. (a) Patching strategy of a spectrogram. Each patch is then flatten to $\mathcal{R}^{p_{df} \times p_{dt}}$ and converted in a token P_i of higher dimension $\mathcal{R}^{d_m}$ in a tokenizer. (b) Generic pipeline of our work: the main branch correspond to a pretrained model which weights are frozen, it includes the tokenizer and the encoder. The auxiliary branch keeps the same architecture as the main encoder but is trainable and use masking on its input. The decoder focuses on information provided by both encoders to reconstruct the masked positions of the auxiliary branch.

Importantly, no labels are used during our training sessions. Emotion recognition and expression detection are only evaluated afterwards, as a way to leverage each encoder output and analyse whether affective information were retained and how they are distributed. Both branches are tested with identical classification heads to compare their performance. This way we highlight whether disentangled information is preserved as a result of the constraints and structure imposed during training.

8.3 Pretraining

8.3.1 Datasets

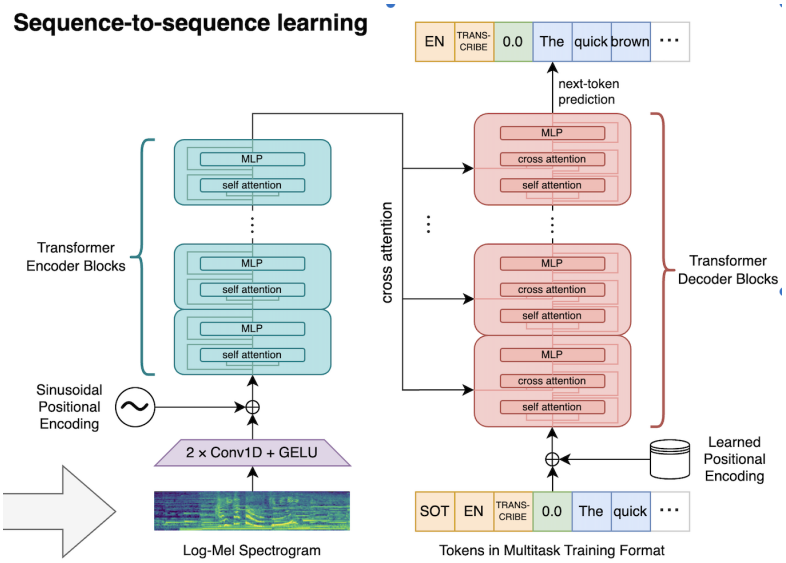
The model was first pre-trained in a self-supervised setting using two separate datasets: Librispeech-960 and VoxCeleb2. This dual-dataset setup acts as an indirect ablation to test how different types of input data affect what the model learns in the absence of emotional supervision. Librispeech offers clean, read speech with minimal expressive variation, while VoxCeleb2 includes in-the-wild recordings where emotional tone may be present but is not annotated or controlled. A complete description of each dataset is available in Chapter 5.

8.3.2 Main branches

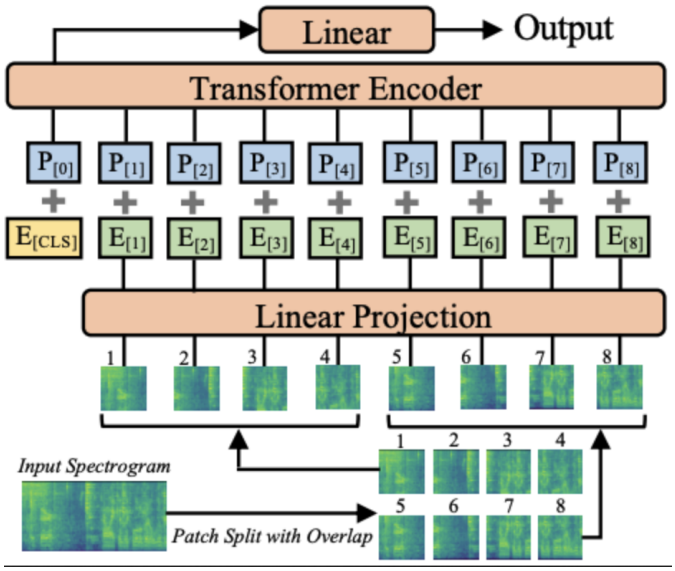
Whisper During our experiments we first attempted to disentangle speech and non-speech features based on Whisper [4] (*base* model, English only). Whisper is an ASR system that has proven efficient for transcription. It follows an encoder-decoder scheme where it first encode information from 3000×80 spectrograms and decode that information into text using token generation (Figure 8.2(a)). While mutliple configurations exist for Whisper, we decided to use OpenAI's *Whisper-base.en* as it was the most downloaded model on HuggingFace[1]. It is composed of 6 encoder layers of dimension 512 and a tokenizer made up of two convolution layers of the converts 3000×80 spectrograms into 1500 patches of dimension 512 with overlapping between patches. As we apply a masking strategy based on patches, we considered patches of 10×16 pixels.

AST Our second experiment was to analyse the information lost from AST [106], an attention-based sound recognition system (Figure 8.2(b)). It converts audio input into 128-bin log-mel spectrograms using a 1024-point Fast Fourier Transform (FFT). The spectrogram is split in 16 by 16 pixels patches and fed to a CNN-based tokenizer which extend the dimension from 256 to $\mathcal{R}^{d_m}$.

[1] https://huggingface.co/



(a) Whisper architecture. It transcribes audio from spectrogram into text based on token generation. Our pipeline uses the encoder blocks only (blue). From [4]



(b) AST architecture. It split spectrogram into patches and fed them to an encoder for classification tasks. Note that the CLS token E_{CLS} is removed from our pipeline. From [106]

Figure 8.2. The original architecture of the main branches considered in this work: (a) Whisper and (b) AST. While the first has a encoder-decoder framework, the second uses only an encoder followed by a classification layer.

8.3.3 Training specifications

Masking Strategy We randomly mask 75% of the input patches and feed it to the auxiliary branch while the main branch receives the original input. After the auxiliary encoder pass, the masked positions are filled with a special token M that serves for the reconstruction task. This is then concatenated with the output of the main branch and fed to the joint decoder to reconstruct the original set of patches.

Hyperparameters The models were trained for 10 epochs using the Adam optimiser with a learning rate of 1×10^{-3} and a batch size of 4. During training, the main encoder remained frozen, and only the auxiliary encoder, decoder, and classification head were updated. Positional encodings were also left unchanged. Model checkpoints were saved based on the best validation loss, and the best weights were loaded for the evaluation on the test subset.

Losses The pretraining of the auxiliary branch relies on three different losses: a reconstruct loss, a contrastive loss and a mutual information loss. They were combined as a weighted sum:

$$\mathcal{L} = \mathcal{L}_r + \lambda \times (\mathcal{L}_c + \mathcal{L}_{mi}) \tag{8.1}$$

with $\lambda = 0.01$. The reconstruction loss $\mathcal{L}_r$ computes the distance between pixels from masked patches e^{mask} and decoded patches $\hat{e}^{\text{mask}}$ at the same position i:

$$\mathcal{L}_r = \frac{1}{N}\sum_{i=1}^{N}[\frac{\sum(\hat{e}_i^{\text{mask}} - e_i^{\text{mask}})^2}{|e_i^{\text{mask}}|}] \tag{8.2}$$

It forces the system to extract relevant features to capture the structure of the input data, based solely on the provided unmask tokens. The contrastive loss $\mathcal{L}_c$ is inspired from Equation 7.3, replacing each modality by one of our branches:

$$\mathcal{L}_c = -\frac{1}{N}\sum_{i=1}^{N}\log\left(\frac{exp(f(cs_i, cx_i))}{\sum_{j=1}^{N} exp(f(cs_i, cx_j)}\right) \tag{8.3}$$

Its sole purpose is to bring the mean output of each encoder closer in value, so the auxiliary weights cannot be zeroed during training. Finally, we use the mutual information as a third loss $\mathcal{L}_{mi}$ and it is defined as:

$$\mathcal{L}_{mi} = \sum_{s\in \mathrm{S}}\sum_{x\in \mathrm{X}} \mathrm{P}_{(X,S)}(x,s)\log\left(\frac{\mathrm{P}_{(X,S)}(x,s)}{\mathrm{P}_X(x)\mathrm{P}_S(s)}\right) \tag{8.4}$$

where $\mathrm{P}_{(X,S)}$ is the joint probability function of X and S, and P_X and P_S are the marginal probability functions of X and S respectively. It can also be expressed as the sum of

marginal entropies $\mathrm{H}(X)$, $\mathrm{H}(S)$ and the joint entropy $\mathrm{H}(X;S)$:

$$\mathcal{L}_{mi} = \mathrm{I}(X;S) = \mathrm{H}(X) + \mathrm{H}(S) - \mathrm{H}(X;S) \tag{8.5}$$

The mutual information is a statistical mesure of the similarities between two sets of random variables. The higher the value, the more information is shared between the sets, and the minimal value of 0 indicates no shared information (Figure 8.3).

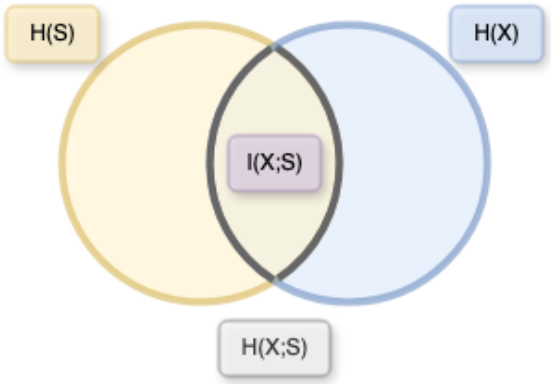


Figure 8.3. Mutual Information representation. The outer circles represent the entropy of each random variable S and X (yellow and blue respectively), while the overlapping area represents the shared information.

8.3.4 Results

In this section we discuss the results of the six training configurations that we were able to train. Table 8.1 shows the total loss for Whisper trained on VoxCeleb2, and for AST trained on either VoxCeleb2 on Librispeech (960h). The results indicate that the models

Table 8.1. Evaluation Loss values for Whisper and AST main branches (lower is better). We conducted two experiments for each configuration: with reconstruction loss only and the weighted sum defined in Equation 8.1.

Whisper	VoxCeleb2	**AST**	VoxCeleb2	Librispeech
reconstruction only	0.0026		0.0038	0.0055
recon-cont-mi	0.0032		0.0087	0.0051

were able to efficiently reconstruct the input spectrogram in all configuration, with a slightly higher loss on AST trained with VoxCeleb2. The performance in pretraining is important to understand whether enough information is provided by the encoder to help reconstruct the masked part of the input. But it does not guarantee that the

information embedded in the auxiliary encoder contains affective features that can be used to perform downstream tasks.

In addition, we provide some zero-shot reconstructions on CREMA-D and NDC-ME spectrograms (Figure 8.4). The original spectrogram is masked then reconstructed by the model while the audio comes from an unknown dataset.

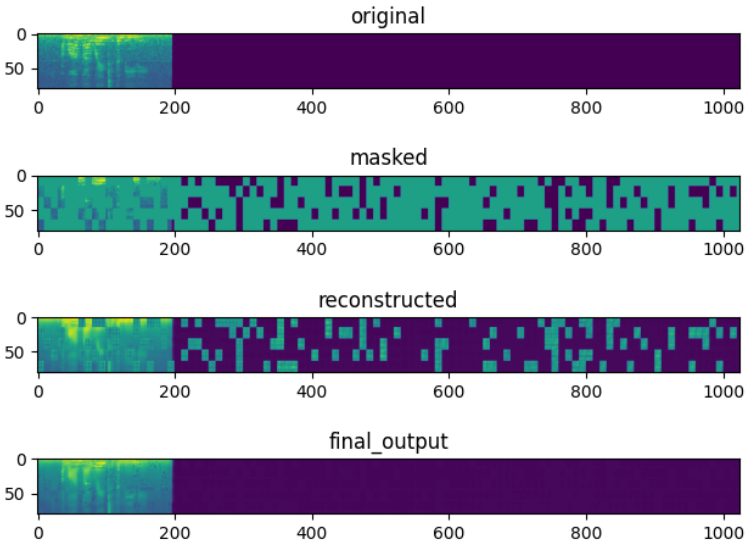
8.4 Downstream tasks

To evaluate social aspect of the content from the auxiliary branch, we discard the masking process and the decoder layers. The process pipeline is depicted in Figure 8.5. We experiment on two affective tasks previously described in this work: emotion recognition on CREMA-D [71] and laughs and smiles detection on NDC-ME [79, 143]. We train the different configurations during 10 epochs using an Adam optimizer with a learning rate starting at 1e-3 and a batch size of 4. The loss function is the Cross Entropy Loss, and the test metrics are the micro and macro f1-score. To assess robustness, the results are averaged across multiple runs.

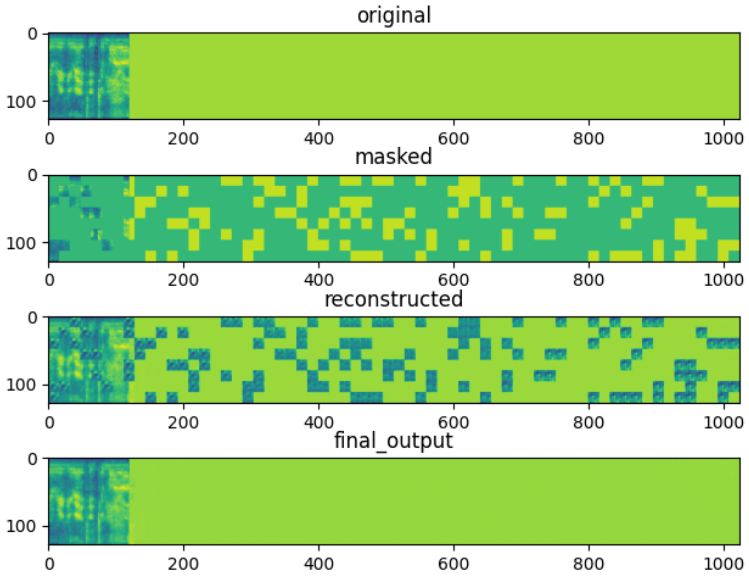
8.4.1 Emotion Recognition

Experimental Setup We evaluated the representations learned during pretraining using a categorical emotion recognition task on CREMA-D. The classification was done across six emotion classes as presented in Chapter 4. To analyse the latent representation learned during pre-training, we froze both the main and auxiliary encoders to prevent fine-tuning and trained only a fully connected layer. We also observe the results from the concatenation of both encoder output to see if their shared information provide better results than a single encoder.

Results and Discussion Table 8.2 contains the micro and macro f1-score for several model configurations pretrained on VoxCeleb2. We observe that the main branch perform better than the auxiliary branch on all configurations. While the late fusion of both encoders show an increase in f1-score, it is still worse than the original method. While these results are not satisfactory regarding our original assumption, we believe that this might be due to a misdesign in the transfert of information between branches. Indeed previous work [27, 144] have shown that improper bottleneck in cross-attention methods lead to decreased performance. Regarding the difference in performance between the best configuration of each architecture, we observe that Whisper reaches better f1-scores than AST. We assume that the original task (Figure 8.2) enables Whisper to leverage more affective information than AST.



(a) Whisper-based reconstruction of CREMA-D (weighted sum loss).



(b) AST-based reconstruction of CREMA-D (weighted sum loss).

Figure 8.4. Zero-shot spectrogram reconstructions after pretraining using (a) Whisper or (b) AST as main branch. We observe a efficient reconstruction from both, with some smoothing in the most intense values (represented by the yellow shades).

In Table 8.3 we observe that the main branch-only still perfom better than both auxiliary configurations by a large margin. But it also informs us on the importance of

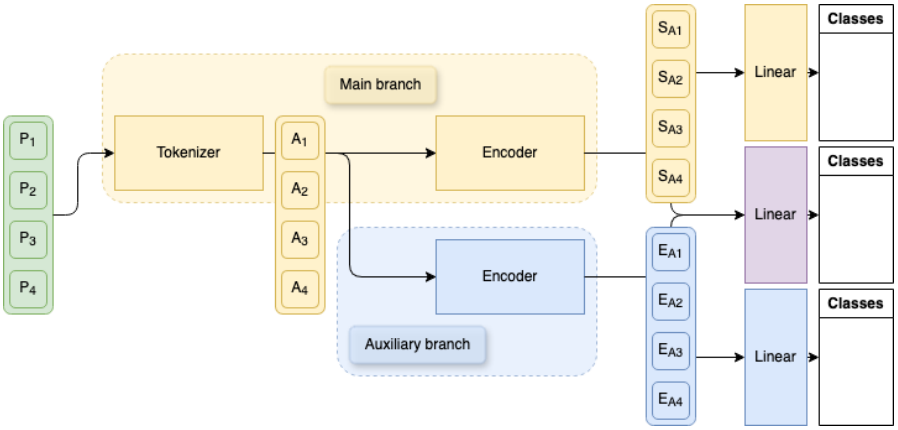


Figure 8.5. The evaluation pipeline for downstream tasks. Compared to the pretrained model, we keep only the encoders and apply no masking strategy to the input tokens. The classification layer is put on top of either one of the encoder branch (main in yellow, auxiliary in blue) or the concatenation of both output (purple).

Table 8.2. Emotion Recognition f1-score (micro and macro). Several configurations, all pretrained on VoxCeleb2, are reported for each main branch architecture: the frozen main branch only, branches from models based reconstruction-only pretraining and branches with all losses applied during pretraining (either auxiliary only or concatenation of both). The two best results for each main architecture are in **bold**.

Configuration		micro f1-score	macro f1-score
whisper-main-only		**0.665**	**0.665**
whisper-reconstruction-only	auxiliary-only	0.420	0.372
	concatenated	**0.657**	**0.657**
whisper-all-losses	auxiliary-only	0.460	0.447
	concatenated	0.466	0.452
ast-main-only		**0.592**	**0.590**
ast-reconstruction-only	auxiliary-only	0.355	0.319
	concatenated	**0.564**	**0.558**
ast-all-losses	auxiliary-only	0.460	0.441
	concatenated	0.522	0.516

using all three losses during pretraining with non-affective data rather than only the reconstruction loss. As mentioned previously, VoxCeleb2 has an in-the-wild recording conditions while Librispeech is considered as "clean". This could explain the difference between *ast-reconstruction-only* and *ast-all-losses* configuration.

Table 8.3. Emotion Recognition f1-score (micro and macro). Three configurations, all AST-based and pretrained on Librispeech, are reported: the frozen main branch only and the auxiliary branches either trained with reconstruction loss only or with all losses. The best results are in **bold**.

Configuration	micro f1-score	macro f1-score
ast-main-only	**0.592**	**0.590**
ast-reconstruction-only	0.248	0.130
ast-all-losses	0.450	0.437

Also unlike previously mentionned approaches, our pretraining method relies only on self-supervision: there is no possibility to efficiently force the transfert of particular information into the auxiliary branch. Among other causes for weak performance is the fact that emotions are already extracted by the main branches we experimented on.

8.4.2 Laughs and Smiles Detection

Experimental Setup To understand how low-level descriptors are encoded, we evaluated the model on NDC-ME, which offers smile and laugh annotations in dyadic interactions. As with the previous task, we extracted embeddings from one encoder branch without finetuning it, and trained a single fully connected layer to predict the expressions included in the input.

Results and Discussion We report the Laughs and Smiles detection performance on NDC-ME in Table 8.4. We used the same evaluation procedure as before: micro and macro F1-score across all classes, averaged over three runs.

As we can see, the main branch perform better on both Whisper and AST. It shows that enough affective information is stored in the original model to reach better results than our auxiliary branches. AST-based auxiliary branches highlight two main behaviours. The first observation is that the contrastive losses increase the performance, especially when it was pretrained on Librispeech. The second observation is that in-the-wild data from VoxCeleb2 provide more affective features in the auxiliary branch than "clean" data from Librispeech. We also observe that Whisper has less features relevant for paralinguistics such as laughs and smiles than AST, resulting in poorer performance in detection.

We extend our analysis of *whisper-main-only* and *ast-main-only* performance with the confusion matrices in Table 8.5. As defined in Chapter 2, it provides the prediction distribution compared to the actual classes. As expected, Laughs are more easily de-

Table 8.4. Laughs and Smiles f1-score (micro and macro). Five AST-based (pretrained either on VoxCeleb2 or Librispeech) and three Whisper-based (pretrained on VoxCeleb2) configurations, are reported: the frozen main branch only and the branches either trained with reconstruction loss only or with all losses. The best results are in **bold**.

Configuration		micro f1-score	macro f1-score
whisper-main-only		**0.530**	**0.532**
whisper-reconstruction-only	VoxCeleb2	0.471	0.426
whisper-all-losses	VoxCeleb2	0.459	0.433
ast-main-only		**0.592**	**0.590**
ast-reconstruction-only	VoxCeleb2	0.460	0.379
	Librispeech	0.337	0.168
ast-all-losses	VoxCeleb2	0.503	0.496
	Librispeech	0.469	0.454

Table 8.5. Confusion Matrix of the best performing Whisper-based and AST-based configurations. Values are expressed in percentage (%). Each row corresponds to the expected label and each column is the predicted label. Row values from each main configuration add up to 100%. **Bold font** highlights the TP.

	whisper-main-only			ast-main-only		
	Laughs	Smiles	None	Laughs	Smiles	None
Laughs	**64.0**	17.5	18.5	**82.9**	11.9	5.2
Smiles	15.9	**37.4**	46.7	17.8	**61.0**	21.2
None	11.4	33.5	**55.1**	13.8	48.0	**38.2**

tected than Smiles and None classes. AST outperforms Whisper mainly due to its higher Precision, as the other classes shared similar distribution. The result reported here is consistent with that observed in Chapter 6 for audio-only modality: Smiles and None are less audible than Laughs.

8.5 In Brief

Summary for Chapter 8

- This chapter investigates how large pre-trained models (such as Whisper and AST) encode and retain affective information, especially when such information is not explicitly supervised during training.
- A dual-branch architecture is introduced to study latent disentanglement: a frozen main branch keeps original capabilities, while a trainable auxiliary branch attempts to extract complementary information for input reconstruction.
- Experiments evaluate whether affective features (e.g., emotions, laughs, and smiles) can be captured in the auxiliary branch using self-supervised learning and loss constraints (reconstruction, contrastive, mutual information).
- Results on emotion recognition and laugh/smile detection indicate that while auxiliary branches may retain some affective cues, frozen main branches typically outperform them, especially when trained on emotionally rich data like VoxCeleb2.

Perspectives for Chapter 8

- We believe more research could be done about the sharing of information between main and auxiliary branches, as our pipeline only masks the auxiliary branch.
- Outside of affective computing, other aspect of audio could be embedded in the latent from the auxiliary branch, such as accent.
- Future work could explore asymmetric encoders, with different architectures for each branch.

Chapter 9

Conclusion and Contributions

9.1 Conclusion

This work investigated the behaviour of large deep learning models in the field of affective computing, focusing on multimodal analysis of non-verbal communication through audio and visual modalities. The main objective was to explore how these systems perceive and process affective signals, with the broader aim of improving transparency, interpretability and reliability of these models in real-world human-agent interactions.

Starting with fundamental concepts, the work examined the theoretical underpinnings of deep learning architectures, including convolutional models and transformer-based models (Chapter 2). Their role in feature extraction and representation learning was discussed in detail, particularly in relation to structured inputs such as images, video and speech. Building on this foundation, we addressed the specific challenges and mechanisms of audiovisual processing and fusion, highlighting the importance of modality alignment, data representation and adaptive integration strategies (Chapter 3).

The study then examined affective computing in terms of low- and high-level descriptors, covering facial expressions, tone of voice, gestures and personality traits. These descriptors were systematically explored using curated and extended datasets, resulting in the design and annotation of the IB dataset (Chapter 5). By enhancing the existing datasets with temporal roles (speaker/listener) and graded intensity labels for smiles and laughter, this work addressed the sparsity of refined affective annotations and enabled non-verbal affective cues to be modelled more precisely.

Empirically, this thesis presented a progression from specific targeted analysis (e.g. lips) to global multimodal encoding using attention and masking strategies. The LSN-TCN architecture (Chapter 6) demonstrated the benefits of localised region analysis, while Social-MAE, (Chapter 7) extended this understanding to full face and voice input, allowing exploration of affect dissociation. This work has resulted in a new architecture that augments pre-trained models with an auxiliary branch (Chapter 8), offering a proof of concept for the conservation of affective information without degrading the performance of the primary model.

Several key results were obtained:

- Multimodal fusion consistently outperformed unimodal baselines, particularly when temporal and intensity cues were available.
- The intensity distribution of smiles and laughter revealed interpretable patterns in social interactions, suggesting potential applications in assistive technologies for populations with ASD.
- Pre-training and transfer learning strategies enabled robust generalisation across datasets, even in low-data regimes.
- The disentanglement approach opened promising directions for post-hoc interpretability in complex networks.

Ultimately, this thesis provides technical innovations and conceptual insights into how deep models can perceive and learn affect. Future research could build on these findings by exploring cross-cultural models of affect, adaptive systems that evolve according to the user's specific affective traits, or closer integration between verbal and non-verbal channels. As affective computing develops, it will be essential to ensure that intelligent systems remain sensitive, interpretable and inclusive if they are to have an impact on society.

9.2 Publications

- **H. Bohy**, K. El Haddad, and T. Dutoit, "A new perspective on smiling and laughter detection: Intensity levels matter", *in 2022 10th International Conference on Affective Computing and Intelligent Interaction (ACII)*. IEEE, 2022.
- **H. Bohy**, "Sensing the mood of a conversation using non-verbal cues with Deep Learning," In *2022 10th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, Nara, Japan, 2022.
- **H. Bohy**, A. Hammoudeh, A. Maiorca, S. Dupont, and T. Dutoit. "Analysis of Co-Laughter Gesture Relationship on RGB Videos in Dyadic Conversation Context". In *Proceedings of the Workshop on Smiling and Laughter across Contexts and the Life-span within the 13th Language Resources and Evaluation Conference*, 2022.
- A. Maiorca, **H. Bohy**, Y. Yoon, and T. Dutoit. "Objective Evaluation Metric for Motion Generative Models: Validating Fréchet Motion Distance on Foot Skating and Over-smoothing Artifacts". In *Proceedings of the 16th ACM SIGGRAPH Conference on Motion, Interaction and Games*, 2023.
- **H. Bohy**, M. Tran, K. El Haddad, T. Dutoit, and M. Soleymani. "Social-MAE: A Transformer-Based Multimodal Autoencoder for Face and Voice." In *2024 IEEE 18th International Conference on Automatic Face and Gesture Recognition (FG)*. IEEE, 2024.

- K. El Haddad, **H. Bohy**, and T. Dutoit. "The Interaction Behavior Dataset: A Dataset of Smiles and Laughs in Dyadic Interaction". In *2025 IEEE International Conference on Artificial Intelligence and eXtended and Virtual Reality (AIxVR)*, 2025.

Bibliography

[1] R. e. Kaliouby, R. Picard, and S. Baron-Cohen, "Affective computing and autism", *Annals of the New York Academy of Sciences*, vol. 1093, no. 1, pp. 228–248, 2006.

[2] T. Madiega, "Artificial intelligence act", 2021.

[3] K. He, X. Zhang, S. Ren, and J. Sun. (2015-12-10) Deep Residual Learning for Image Recognition.

[4] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision", in *International Conference on Machine Learning*. PMLR, 2023, pp. 28 492–28 518.

[5] D.-A. et al., "Deepseek-v3 technical report", 2024. [Online]. Available: https://arxiv.org/abs/2412.19437

[6] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization", *arXiv preprint arXiv:1412.6980*, 2014.

[7] S. Ruder, "An overview of gradient descent optimization algorithms", *arXiv preprint arXiv:1609.04747*, 2016.

[8] K. O'shea and R. Nash, "An introduction to convolutional neural networks", *arXiv preprint arXiv:1511.08458*, 2015.

[9] L. Lu, L. Tavabi, and M. Soleymani, "Self-supervised learning for facial action unit recognition through temporal consistency." in *BMVC*, 2020.

[10] A. Asokan, J. Anitha, M. Ciobanu, A. Gabor, A. Naaji, and D. J. Hemanth, "Image processing techniques for analysis of satellite images for historical maps classification—an overview", *Applied Sciences*, vol. 10, no. 12, p. 4207, 2020.

[11] R. L. Kumar, J. Kakarla, B. V. Isunuri, and M. Singh, "Multi-class brain tumor classification using residual network and global average pooling", *Multimedia Tools and Applications*, vol. 80, no. 9, pp. 13 429–13 438, 2021.

[12] J. Li, Y. Han, M. Zhang, G. Li, and B. Zhang, "Multi-scale residual network

model combined with global average pooling for action recognition", *Multimedia Tools and Applications*, pp. 1–19, 2022.

[13] S. A. Mahmoudi, O. Amel, S. Stassin, M. Liagre, M. Benkedadra, and M. Mancas, "A review and comparative study of explainable deep learning models applied on action recognition in real time", *Electronics*, vol. 12, no. 9, p. 2027, 2023.

[14] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao, "Review of image classification algorithms based on convolutional neural networks", *Remote Sensing*, vol. 13, no. 22, p. 4712, 2021.

[15] C. Luo, X. Li, L. Wang, J. He, D. Li, and J. Zhou, "How does the data set affect cnn-based image classification performance?" in *2018 5th international conference on systems and informatics (ICSAI)*. IEEE, 2018, pp. 361–366.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need", *Advances in neural information processing systems*, vol. 30, 2017.

[17] M. V. Koroteev, "Bert: a review of applications in natural language processing and understanding", *arXiv preprint arXiv:2103.11943*, 2021.

[18] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report", *arXiv preprint arXiv:2303.08774*, 2023.

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale", *arXiv preprint arXiv:2010.11929*, 2020.

[20] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin, "Convolutional sequence to sequence learning", in *International conference on machine learning*. PMLR, 2017, pp. 1243–1252.

[21] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations", *Advances in Neural Information Processing Systems*, vol. 33, pp. 12 449–12 460, 2020.

[22] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units", *IEEE/ACM transactions on audio, speech, and language*

processing, vol. 29, pp. 3451–3460, 2021.

[23] M. Tagliasacchi, B. Gfeller, F. de Chaumont Quitry, and D. Roblek, "Pre-training audio representations with self-supervision", *IEEE Signal Processing Letters*, vol. 27, pp. 600–604, 2020.

[24] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks", in *International conference on machine learning*. PMLR, 2019, pp. 6105–6114.

[25] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers", in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 9650–9660.

[26] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision", in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.

[27] A. Nagrani, S. Yang, A. Arnab, A. Jansen, C. Schmid, and C. Sun, "Attention bottlenecks for multimodal fusion", *Advances in neural information processing systems*, vol. 34, pp. 14 200–14 213, 2021.

[28] R. W. Picard, *Affective computing*. MIT press, 2000.

[29] C. E. Izard, *The psychology of emotions*. Springer Science & Business Media, 1991.

[30] K. Leidelmeijer, *Emotions: An experimental approach*. Tilburg University Press Tilburg, 1991.

[31] S. Gedam and S. Paul, "A review on mental stress detection using wearable sensors and machine learning techniques", *IEEE Access*, vol. 9, pp. 84 045–84 066, 2021.

[32] T. Nijhawan, G. Attigeri, and T. Ananthakrishna, "Stress detection using natural language processing and machine learning over social interactions", *Journal of Big Data*, vol. 9, no. 1, p. 33, 2022.

[33] L. Mou, C. Zhou, P. Zhao, B. Nakisa, M. N. Rastgoo, R. Jain, and W. Gao, "Driver stress detection via multimodal fusion using attention-based cnn-lstm", *Expert Systems with Applications*, vol. 173, p. 114693, 2021.

[34] N. V. Babu and E. G. M. Kanaga, "Sentiment analysis in social media data for depression detection using artificial intelligence: a review", *SN computer science*,

vol. 3, no. 1, p. 74, 2022.

[35] M. Rodríguez-Ibánez, A. Casánez-Ventura, F. Castejón-Mateos, and P.-M. Cuenca-Jiménez, "A review on sentiment analysis from social media platforms", *Expert Systems with Applications*, vol. 223, p. 119862, 2023.

[36] L. F. Barrett, R. Adolphs, S. Marsella, A. M. Martinez, and S. D. Pollak, "Emotional expressions reconsidered: Challenges to inferring emotion from human facial movements", *Psychological science in the public interest*, vol. 20, no. 1, pp. 1–68, 2019.

[37] K. Crawford and T. Paglen, "Excavating ai: The politics of images in machine learning training sets", *Ai & Society*, vol. 36, no. 4, pp. 1105–1116, 2021.

[38] S. Cano, C. S. González, R. M. Gil-Iranzo, and S. Albiol-Pérez, "Affective communication for socially assistive robots (sars) for children with autism spectrum disorder: A systematic review", *Sensors*, vol. 21, no. 15, p. 5166, 2021.

[39] L. Balcombe and D. De Leo, "Human-computer interaction in digital mental health", in *Informatics*, vol. 9, no. 1. MDPI, 2022, p. 14.

[40] N. Mejbri, F. Essalmi, M. Jemni, and B. A. Alyoubi, "Trends in the use of affective computing in e-learning environments", *Education and Information Technologies*, pp. 1–23, 2022.

[41] G. N. Yannakakis and D. Melhart, "Affective game computing: A survey", *Proceedings of the IEEE*, vol. 111, no. 10, pp. 1423–1444, 2023.

[42] M. L. Knapp, J. A. Hall, and T. G. Horgan, *Nonverbal communication in human interaction*. Holt, Rinehart and Winston New York, 1978, vol. 1.

[43] A. Mehrabian, *Nonverbal communication*. Routledge, 2017.

[44] P. Ekman, "Emotional and conversational nonverbal signals", in *Language, knowledge, and representation: Proceedings of the sixth international colloquium on cognitive science (ICCS-99)*. Springer, 2004, pp. 39–50.

[45] P. Ekman and W. V. Friesen, "Facial action coding system", *Environmental Psychology & Nonverbal Behavior*, 1978.

[46] T. Kanade, J. F. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis", in *Proceedings fourth IEEE international conference on automatic face and gesture recognition (cat. No. PR00580)*. IEEE, 2000, pp. 46–53.

[47] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields", *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 1, pp. 172–186, 2019.

[48] D. S. Johnson, O. Hakobyan, J. Paletschek, and H. Drimalla, "Explainable ai for audio and visual affective computing: A scoping review", *IEEE Transactions on Affective Computing*, 2024.

[49] J. A. Bargh and T. L. Chartrand, "The unbearable automaticity of being." *American psychologist*, vol. 54, no. 7, p. 462, 1999.

[50] W. James, "The emotions." 1922.

[51] R. S. Lazarus, *Emotion and adaptation*. Oxford University Press, 1991.

[52] P. Ekman, "Are there basic emotions?" *Current Psychology*, 1992.

[53] R. Plutchik, "A general psychoevolutionary theory of emotion", in *Theories of emotion*. Elsevier, 1980, pp. 3–33.

[54] J. A. Russell, "A circumplex model of affect." *Journal of personality and social psychology*, vol. 39, no. 6, p. 1161, 1980.

[55] I. Bakker, T. Van Der Voordt, P. Vink, and J. De Boon, "Pleasure, arousal, dominance: Mehrabian and russell revisited", *Current Psychology*, vol. 33, pp. 405–421, 2014.

[56] L. R. Goldberg, "An alternative "description of personality": The big-five factor structure", in *Personality and personality disorders*. Routledge, 2013, pp. 34–47.

[57] Y. Kopparapu, P. K. Rai, V. Kambalyal, S. Siddiqui, and P. Patil, "Audio denoising using machine learning", in *International Conference on Computing and Machine Learning*. Springer, 2024, pp. 489–500.

[58] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books", in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[59] G. Zhu, J.-P. Caceres, and J. Salamon, "Filler word detection and classification: A dataset and benchmark", *arXiv preprint arXiv:2203.15135*, 2022.

[60] C. Wang, M. Riviere, A. Lee, A. Wu, C. Talnikar, D. Haziza, M. Williamson, J. Pino, and E. Dupoux, "Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation", *arXiv*

preprint arXiv:2101.00390, 2021.

[61] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "Vggface2: A dataset for recognising faces across pose and age", in *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*. IEEE, 2018, pp. 67–74.

[62] R. Sanabria, N. Bogoychev, N. Markl, A. Carmantini, O. Klejch, and P. Bell, "The Edinburgh International Accents of English Corpus: Towards the Democratization of English ASR", in *ICASSP 2023*, 2023.

[63] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition", *arXiv preprint arXiv:1806.05622*, 2018.

[64] A. Adigwe, N. Tits, K. E. Haddad, S. Ostadabbas, and T. Dutoit, "The emotional voices database: Towards controlling the emotion dimension in voice generation systems", *arXiv preprint arXiv:1806.09514*, 2018.

[65] L. G. Zaragozá, R. del Amor, E. P. Vargas, V. Naranjo, M. A. Raya, and J. Marín-Morales, "Emotional voice messages (emovome) database: emotion recognition in spontaneous voice messages", *arXiv preprint arXiv:2402.17496*, 2024.

[66] R. Lotfian and C. Busso, "Building naturalistic emotionally balanced speech corpus by retrieving emotional speech from existing podcast recordings", *IEEE Transactions on Affective Computing*, vol. 10, no. 4, pp. 471–483, October-December 2019.

[67] A. Mollahosseini, B. Hasani, and M. H. Mahoor, "Affectnet: A database for facial expression, valence, and arousal computing in the wild", *IEEE Transactions on Affective Computing*, vol. 10, no. 1, pp. 18–31, 2017.

[68] D. Kollias and S. Zafeiriou, "Aff-wild2: Extending the aff-wild database for affect recognition", *arXiv preprint arXiv:1811.07770*, 2018.

[69] A. Subramaniam, V. Patel, A. Mishra, P. Balasubramanian, and A. Mittal, "Bimodal First Impressions Recognition Using Temporally Ordered Deep Audio and Stochastic Visual Features", in *Computer Vision – ECCV 2016 Workshops*, G. Hua and H. Jégou, Eds. Springer International Publishing, 2016, vol. 9915, pp. 337–348.

[70] C. Busso, S. Parthasarathy, A. Burmania, M. AbdelWahab, N. Sadoughi, and E. Mower Provost, "MSP-IMPROV: An acted corpus of dyadic interactions to study emotion perception", *IEEE Transactions on Affective Computing*, vol. 8, no. 1, pp. 67–80, January-March 2017.

[71] H. Cao, D. G. Cooper, M. K. Keutmann, R. C. Gur, A. Nenkova, and R. Verma, "Crema-d: Crowd-sourced emotional multimodal actors dataset", *IEEE transactions on affective computing*, vol. 5, no. 4, pp. 377–390, 2014.

[72] C. Mazzocconi and J. Ginzburg, "A longitudinal characterization of typical laughter development in mother–child interaction from 12 to 36 months: Formal features and reciprocal responsiveness", *Journal of Nonverbal Behavior*, vol. 46, no. 4, pp. 327–362, 2022.

[73] J. Hough, Y. Tian, L. de Ruiter, S. Betz, S. Kousidis, D. Schlangen, and J. Ginzburg, "DUEL: A multi-lingual multimodal dialogue corpus for disfluency, exclamations and laughter", in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*. Portorož, Slovenia: European Language Resources Association (ELRA), May 2016, pp. 1784–1788. [Online]. Available: https://aclanthology.org/L16-1281

[74] K. Bodur, M. Nikolaus, F. Kassim, L. Prévot, and A. Fourtassi, "Chico: A multimodal corpus for the study of child conversation", in *Companion Publication of the 2021 International Conference on Multimodal Interaction*, ser. ICMI '21 Companion. New York, NY, USA: Association for Computing Machinery, 2021, p. 158–163. [Online]. Available: https://doi.org/10.1145/3461615.3485399

[75] B. Priego-Valverde, B. Bigi, and M. Amoyal, ""cheese!": a corpus of face-to-face French interactions. a case study for analyzing smiling and conversational humor", in *Proceedings of the Twelfth Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, May 2020, pp. 467–475. [Online]. Available: https://aclanthology.org/2020.lrec-1.59

[76] G. McKeown, R. Cowie, W. Curran, W. Ruch, and E. Douglas-Cowie, "Ilhaire laughter database", in *Proceedings of 4th International Workshop on Corpora for Research on Emotion, Sentiment & Social Signals, LREC*, 2012, pp. 32–35.

[77] A. J. Aubrey, D. Marshall, P. L. Rosin, J. Vendeventer, D. W. Cunningham, and C. Wallraven, "Cardiff conversation database (ccdb): A database of natural dyadic conversations", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2013, pp. 277–282.

[78] R. J. Van Son, D. Binnenpoorte, H. v. d. Heuvel, and L. Pols, "The ifa corpus: a phonemically segmented dutch" open source" speech database", 2001.

[79] L. Heron, J. Kim, M. Lee, K. El Haddad, S. Dupont, T. Dutoit, and K. Truong, "A

dyadic conversation dataset on moral emotions", in *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*. IEEE, 2018, pp. 687–691.

[80] K. El Haddad, N. Tits, and T. Dutoit, "Annotating nonverbal conversation expressions in interaction datasets", in *Proc. LW 2018*, 2018, pp. 54–57.

[81] M. Schröder, V. Aubergé, and M.-A. Cathiard, "Can we hear smile?" in *Fifth International Conference on Spoken Language Processing*, 1998.

[82] C. Harris and N. Alvarado, "Facial expressions, smile types, and self-report during humour, tickle, and pain", *Cognition & Emotion*, vol. 19, no. 5, pp. 655–669, 2005.

[83] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data", *biometrics*, pp. 159–174, 1977.

[84] H. Bohy, K. El Haddad, and T. Dutoit, "A new perspective on smiling and laughter detection: Intensity levels matter", in *2022 10th International Conference on Affective Computing and Intelligent Interaction (ACII)*. IEEE, 2022, pp. 1–8.

[85] H. Bohy, M. Tran, K. El Haddad, T. Dutoit, and M. Soleymani, "Social-mae: A transformer-based multimodal autoencoder for face and voice", in *2024 IEEE 18th International Conference on Automatic Face and Gesture Recognition (FG)*. IEEE, 2024, pp. 1–5.

[86] N. Tits, K. E. Haddad, and T. Dutoit, "Laughter synthesis: Combining seq2seq modeling with transfer learning", *arXiv preprint arXiv:2008.09483*, 2020.

[87] H. Bohy, A. Hammoudeh, A. Maiorca, S. Dupont, and T. Dutoit, "Analysis of co-laughter gesture relationship on rgb videos in dyadic conversation contex", *arXiv preprint arXiv:2205.10266*, 2022.

[88] P. Saha, D. Bhattacharjee, B. K. De, and M. Nasipuri, "An approach to detect the region of interest of expressive face images", *Procedia Computer Science*, vol. 46, pp. 1739–1746, 2015.

[89] M. Arsalan, H. G. Hong, R. A. Naqvi, M. B. Lee, M. C. Kim, D. S. Kim, C. S. Kim, and K. R. Park, "Deep learning-based iris segmentation for iris recognition in visible light environment", *Symmetry*, vol. 9, no. 11, p. 263, 2017.

[90] Q. Jaleel and I. Hadi, "Facial action unit-based deepfake video detection using deep learning", in *2022 4th International Conference on Current Research in Engineering and Science Applications (ICCRESA)*. IEEE, 2022, pp. 228–233.

[91] X. Guo, L. Polania, and K. Barner, "Smile detection in the wild based on transfer learning", in *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*. IEEE, 2018, pp. 679–686.

[92] J. Whitehill, G. Littlewort, I. Fasel, M. Bartlett, and J. Movellan, "Toward practical smile detection", *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 11, pp. 2106–2111, 2009.

[93] D. Cui, G.-B. Huang, and T. Liu, "Elm based smile detection using distance vector", *Pattern Recognition*, vol. 79, pp. 356–369, 2018.

[94] V. C. Tartter and D. Braun, "Hearing smiles and frowns in normal and whisper registers", *The Journal of the Acoustical Society of America*, vol. 96, no. 4, pp. 2101–2107, 1994.

[95] S. UMR9912 and I. UMR7289, "Hearing smiles and smiling back", *Jonathan Ginzburg, Catherine Pelachaud (eds.)*, p. 12, 2018.

[96] P. Arias, P. Belin, and J.-J. Aucouturier, "Auditory smiles trigger unconscious facial imitation", *Current Biology*, vol. 28, no. 14, pp. R782–R783, 2018.

[97] R. B. Kantharaju, F. Ringeval, and L. Besacier, "Automatic recognition of affective laughter in spontaneous dyadic interactions from audiovisual signals", in *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, 2018, pp. 220–228.

[98] R. Niewiadomski, M. Mancini, G. Varni, G. Volpe, and A. Camurri, "Automated laughter detection from full-body movements", *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 1, pp. 113–123, 2015.

[99] B. B. Turker, Y. Yemez, T. M. Sezgin, and E. Erzin, "Audio-facial laughter detection in naturalistic dyadic conversations", *IEEE Transactions on Affective Computing*, vol. 8, no. 4, pp. 534–545, 2017.

[100] F. Yang, M. A. Sehili, C. Barras, and L. Devillers, "Smile and laughter detection for elderly people-robot interaction", in *Social Robotics: 7th International Conference, ICSR 2015, Paris, France, October 26-30, 2015, Proceedings 7*. Springer, 2015, pp. 694–703.

[101] A. Ito, X. Wang, M. Suzuki, and S. Makino, "Smile and laughter recognition using speech processing and face recognition from conversation video", in *2005 International Conference on Cyberworlds (CW'05)*, 2005, pp. 8 pp.–444.

[102] B. Martinez, P. Ma, S. Petridis, and M. Pantic, "Lipreading using temporal convolutional networks", in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6319–6323.

[103] Y. A. Farha and J. Gall, "Ms-tcn: Multi-stage temporal convolutional network for action segmentation", in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3575–3584.

[104] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

[105] Z. Cai, S. Ghosh, K. Stefanov, A. Dhall, J. Cai, H. Rezatofighi, R. Haffari, and M. Hayat, "Marlin: Masked autoencoder for facial video representation learning", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1493–1504.

[106] Y. Gong, Y.-A. Chung, and J. Glass, "Ast: Audio spectrogram transformer", *arXiv preprint arXiv:2104.01778*, 2021.

[107] H. M. Fayek and A. Kumar, "Large scale audiovisual learning of sounds with weakly labeled data", *arXiv preprint arXiv:2006.01595*, 2020.

[108] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events", in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[109] Y. Gong, A. H. Liu, A. Rouditchenko, and J. Glass, "Uavm: Towards unifying audio and visual models", *IEEE Signal Processing Letters*, vol. 29, pp. 2437–2441, 2022.

[110] M. Tran, Y. Kim, C.-C. Su, C.-H. Kuo, and M. Soleymani, "Saaml: A framework for semi-supervised affective adaptation via metric learning", in *Proceedings of the 31st ACM International Conference on Multimedia*, ser. MM '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 6004–6015.

[111] A. Owens, P. Isola, J. McDermott, A. Torralba, E. H. Adelson, and W. T. Freeman, "Visually indicated sounds", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2405–2413.

[112] A. Owens, J. Wu, J. H. McDermott, W. T. Freeman, and A. Torralba, "Ambient sound provides supervision for visual learning", in *Computer Vision–ECCV 2016:*

14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer, 2016, pp. 801–816.

[113] A. Owens and A. A. Efros, "Audio-visual scene analysis with self-supervised multisensory features", in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 631–648.

[114] E. Ng, D. Xiang, H. Joo, and K. Grauman, "You2me: Inferring body pose in egocentric video via first and second person interactions", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9890–9900.

[115] T. Afouras, A. Owens, J. S. Chung, and A. Zisserman, "Self-supervised learning of audio-visual objects from video", in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*. Springer, 2020, pp. 208–224.

[116] A. Furnari and G. M. Farinella, "Rolling-unrolling lstms for action anticipation from first-person video", *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 11, pp. 4021–4036, 2020.

[117] B. Shi, A. Mohamed, and W.-N. Hsu, "Learning lip-based audio-visual speaker embeddings with av-hubert", *arXiv preprint arXiv:2205.07180*, 2022.

[118] B. Korbar, D. Tran, and L. Torresani, "Cooperative learning of audio and video models from self-supervised synchronization", *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[119] R. Arandjelovic and A. Zisserman, "Look, listen and learn", in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 609–617.

[120] R. Gao, C. Chen, Z. Al-Halah, C. Schissler, and K. Grauman, "Visualechoes: Spatial image representation learning through echolocation", in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*. Springer, 2020, pp. 658–676.

[121] M.-I. Georgescu, E. Fonseca, R. T. Ionescu, M. Lucic, C. Schmid, and A. Arnab, "Audiovisual masked autoencoders", in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 16 144–16 154.

[122] P.-Y. Huang, V. Sharma, H. Xu, C. Ryali, Y. Li, S.-W. Li, G. Ghosh, J. Malik, C. Feichtenhofer *et al.*, "Mavil: Masked audio-video learners", *Advances in Neural*

Information Processing Systems, vol. 36, 2024.

[123] Y. Gong, A. Rouditchenko, A. H. Liu, D. Harwath, L. Karlinsky, H. Kuehne, and J. Glass, "Contrastive audio-visual masked autoencoder", *arXiv preprint arXiv:2210.07839*, 2022.

[124] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization", *arXiv preprint arXiv:1607.06450*, 2016.

[125] S. Lee, Y. Yu, G. Kim, T. Breuel, J. Kautz, and Y. Song, "Parameter efficient multimodal transformers for video representation learning", *arXiv preprint arXiv:2012.04124*, 2020.

[126] L. Goncalves and C. Busso, "AuxFormer: Robust Approach to Audiovisual Emotion Recognition", in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022-05-23, pp. 7357–7361.

[127] F. Eyben, M. Wöllmer, and B. Schuller, "Opensmile: The munich versatile and fast open-source audio feature extractor", in *Proceedings of the 18th ACM International Conference on Multimedia*, ser. MM '10. Association for Computing Machinery, 2010-10-25, pp. 1459–1462.

[128] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition", in *Procedings of the British Machine Vision Conference 2015*. British Machine Vision Association, 2015, pp. 41.1–41.12.

[129] L. Goncalves, S.-G. Leem, W.-C. Lin, B. Sisman, and C. Busso, "Versatile audio-visual learning for handling single and multi modalities in emotion regression and classification tasks", *arXiv preprint arXiv:2305.07216*, 2023.

[130] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition", *arXiv preprint arXiv:2005.08100*, 2020.

[131] W.-N. Hsu, A. Sriram, A. Baevski, T. Likhomanenko, Q. Xu, V. Pratap, J. Kahn, A. Lee, R. Collobert, G. Synnaeve *et al.*, "Robust wav2vec 2.0: Analyzing domain shift in self-supervised pre-training", *arXiv preprint arXiv:2104.01027*, 2021.

[132] Y. Güçlütürk, U. Güçlü, M. A. J. Van Gerven, and R. Van Lier, "Deep Impression: Audiovisual Deep Residual Networks for Multimodal Apparent Personality Trait Recognition", in *Computer Vision – ECCV 2016 Workshops*, G. Hua and H. Jégou,

Eds. Springer International Publishing, 2016, vol. 9915, pp. 349–358.

[133] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey", *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2016.

[134] C.-L. Zhang, H. Zhang, X.-S. Wei, and J. Wu, "Deep Bimodal Regression for Apparent Personality Analysis", in *Computer Vision – ECCV 2016 Workshops*, ser. Lecture Notes in Computer Science, G. Hua and H. Jégou, Eds. Springer International Publishing, 2016, pp. 311–324.

[135] C. I. Gutierrez, A. Aguirre, R. Uuk, C. C. Boine, and M. Franklin, "A proposal for a definition of general purpose artificial intelligence systems", *Digital Society*, vol. 2, no. 3, p. 36, 2023.

[136] I. Khemakhem, D. Kingma, R. Monti, and A. Hyvarinen, "Variational autoencoders and nonlinear ica: A unifying framework", in *International conference on artificial intelligence and statistics*, 2020, pp. 2207–2217.

[137] F. Locatello, S. Bauer, M. Lucic, G. Raetsch, S. Gelly, B. Schölkopf, and O. Bachem, "Challenging common assumptions in the unsupervised learning of disentangled representations", in *international conference on machine learning*. PMLR, 2019, pp. 4114–4124.

[138] R. Peri, S. Parthasarathy, C. Bradshaw, and S. Sundaram, "Disentanglement for audio-visual emotion recognition using multitask setup", in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6344–6348.

[139] S. Wang, Y. Ma, and Y. Ding, "Exploring Complementary Features in Multi-Modal Speech Emotion Recognition", in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Rhodes Island, Greece: IEEE, Jun. 2023, pp. 1–5.

[140] A.-R. Ispas, T. Deschamps-Berger, and L. Devillers, "A Multi-Task, Multi-Modal Approach for Predicting Categorical and Dimensional Emotions", in *International Cconference on Multimodal Interaction*, Oct. 2023, pp. 311–317.

[141] L. Qu, T. Li, C. Weber, T. Pekarek-Rosin, F. Ren, and S. Wermter, "Disentangling prosody representations with unsupervised speech reconstruction", *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 39–54, 2023.

[142] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification", *arXiv preprint arXiv:2005.07143*, 2020.

[143] K. El Haddad, H. Bohy, and T. Dutoit, "The interaction behavior dataset: A dataset of smiles and laughs in dyadic interaction", in *2025 IEEE International Conference on Artificial Intelligence and eXtended and Virtual Reality (AIxVR)*. IEEE, 2025, pp. 399–403.

[144] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson, "Autovc: Zero-shot voice style transfer with only autoencoder loss", in *International Conference on Machine Learning*. PMLR, 2019, pp. 5210–5219.

List of Figures

List of Tables