The Pennsylvania State University
The Graduate School

IMPROVED FACIAL EXPRESSION RECOGNITION USING NOISY STUDENT TRAINING AND SYNTHETIC DATA GENERATION

A Thesis in
Computer Science and Engineering
by
Vikas Kumar

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

August 2021

The thesis of Vikas Kumar was reviewed and approved by the following:

Daniel Kifer
Professor of Computer Science and Engineering
Thesis Advisor

C. Lee Giles
David Reese Professor at the College of Information Sciences and Technology
Committee Member

Chitaranjan Das
Distinguished Professor of Computer Science and Engineering
Department Head and Distinguished Professor

Abstract

Facial expression recognition from videos in the wild is a challenging task due to the lack of abundant labelled training data. Large DNN (deep neural network) architectures and ensemble methods have resulted in better performance, but soon reach saturation at some point due to data inadequacy. This thesis presents a video-based cost-effective facial expression recognition system that is capable of recognizing basic facial expressions. Our method addresses three fundamental issues: (i) it isolates different regions of the face and processes them independently using a multi-level attention mechanism achieving good performance in a cost-effective manner, (ii) it uses a self-training method that utilizes a combination of a labelled dataset and an unlabelled dataset (Body Language Dataset - BoLD) iteratively helping in addressing the lack of labelled datasets, and (iii) it generates a large-scale facial expression dataset Affect-Net-Vid using a proposed generative network StarGAN-EgVA. Our results show that the proposed method achieves state-of-the-art performance on benchmark datasets Affect-Net-Vid and AFEW 8.0 when compared to other single models.

Table of Contents

List of Figures

List of Tables

Acknowledgments

We acknowledge the contribution of Dr. James Wang for providing the opportunity to work on this project during his course on Artificial Emotion Intelligence taught at the Pennsylvania State University.

Chapter 1
Introduction

1.1 Motivation of the problem

Facial changes of facial expressions are responses to a person's internal emotional states, intentions, or social communications [4]. Darwin et al. [5] established facial expression analysis as a research field in 1872. Since then, facial expression recognition (FER) has received a great deal of attention and been an active research topic across a variety of disciplines, such as biology [5], neuroscience [6], psychology [7], and computer vision. Especially in computer vision, for its impact and prominent potentiality, automatic FER has been growing in an extensive range of applications, e.g., HCI, biometric identification, surveillance and security [8], intensive care monitoring [9], aerial image analysis [10], driver state surveillance [11], and human entertainment industry and virtual reality.

The origins of the growing interest in FER in the past decade can be summarized in two main points. The first gives credit to the developing progress accomplished in related research fields such as machine learning, image processing, and human cognition [11] and tasks such as face detection, face tracking and face recognition. The second is due to the recent availability of relatively cheap computational power [12]. Although there has been extensive research on this subject, facial expression recognition in-the-wild remains a challenging problem because of several factors such as occlusion, illumination, motion blur, subject-specific facial variations, along with the lack of extensive labelled training datasets [13]. Following a similar line of research, our task aims to classify a given video in the wild to one of the seven broad categorical emotions. We propose an efficient model that addresses the challenges posed by videos in the wild while tackling the issue of labelled data inadequacy using noisy student training [1] and synthetic data generation. The input data used for facial expression recognition can be multi-modal, i.e. it may have visual information as well as audio information. However, the scope of this work is

limited to emotion classification using only visual information.

1.2 Challenges

Although FER under controlled conditions is already mature and no longer a substantial problem, it is still a challenge for computers to make accurate inferences in real-life scenarios. The challenges of achieving computational facial expression analysis can be classified into four main aspects. Firstly, the subjects differ in terms of head shapes and ethnic groups across the dataset. Different subjects in the same dataset express the same emotion to various extent. Secondly, for the same dataset, images can vary substantially in illumination, head-pose, or background. The third problem is that most existing FER datasets (e.g., CK+ [14], JAFFE [15], MMI [16], RaFD [17], etc.) contain posed facial expressions that are presented by professional actors/actresses or stylized characters, which is different from real-life scenarios, in which people generally do not express their emotions with exaggerated facial expressions. High similarity between two specific classes of facial expressions, e.g., disgust with angry and sadness with fear, which sometimes leads to misclassification, represents the fourth challenge. Most datsets collected from mainstream cinema are imbalanced and biased against some categories such as fear, surprise and disgust.

Most of the recent research on the publicly-available in-the-wild datasets such as AFEW 8.0 (Acted Facial Expressions in the Wild) [18] dataset has focused on improving accuracy without regard to computational complexity, architectural complexity, energy and policy considerations, generality, and training efficiency. Several state-of-the-art methods [19–21] on this dataset have originated from the EmotiW [22] challenge with no clear computational-cost analysis. Prior research [23–26] uses simple aggregation or averaging operation on features from multiple frames to form a fixed-dimensional feature vector. However, such methods do not account for the fact that a few principal frames in a video can be used to identify the target emotion, while the rest of the frames have a negligible contribution. Fan et al. [19] achieved the highest validation accuracy on AFEW 8.0 [18] based on visual cues, but they used a fusion of five different architectures with more than 300 million parameters. Though these models achieve good performance, they are impractical for real-time uses and cannot be used in fields such as biometric identification, survelliance and security, HCI, etc. In contrast, our proposed method uses a single model with approximately 25 million parameters and comparable performance.

Despite being a long-established dataset, AFEW 8.0 [18] has several shortcomings.

Firstly, the dataset contains significantly fewer training examples for fear, surprise and disgust categories which makes the dataset imbalanced. Secondly, the videos are extracted from mainstream cinema, and scenes depicting fear are often shot in the dark, which again makes the model biased towards other categories [20, 27]. Such limitations warrant the use of additional datasets for better generalization. However, not many in-the-wild labelled video datasets are publicly available for facial expression recognition. Several related datasets are captured in posed conditions, while other in-the-wild datasets are only labelled for image-based expression analysis instead of video based analysis. Aff-Wild2 [29] and Affect-Net [28] are other popular datasets, but they contain per-frame annotations, and thus cannot be used in our work which performs video-level classification based on facial expressions.

1.3 Objectives

The objectives of our thesis are as follows:

1. Build a low-cost new training method that is trained using semi-supervised learning to achieve maximum efficiency. The objective of our trained model is to achieve accuracy superior to the current state-of-the-art methods on two publicly available datasets, while being feasible for real-time applications.

2. Use semi-supervised learning to train the model on supervised as well as unsupervised data. Semi-supervised learning helps to overcome the lack of in-the-wild labelled video datasets for facial expression recognition.

3. Generate a new video-level annotated in-the-wild video dataset using GAN (Generative Adversarial Networks), that is trained on a per-frame annotated in-the-wild dataset.

1.4 Thesis Statement

The deep learning methods have already shown its prominence in feature learning and classification power relative to conventional methods. In our work, a new convolutional neural network model is applied to accomplish this FER task, which has a much smaller size and lower computation complexity than other state-of-the-art methods. To address the problem of the insufficient size of those small facial expression datasets, we use

semi-supervised learning to take advantage of both labelled and unlabelled datsets. Additionally, we generate a synthetic facial expression dataset using generative adversarial network to address the problems posed by labelled data inadequacy. While previous work focused on improving performance by increasing model capacity, our method focuses on better pre-processing, feature selection, and semi-supervised training.

The main contributions of this work are as follows:

1. We use a three-level attention mechanism in our model: a) spatial-attention block that helps to selectively process feature maps of a frame, b) channel-attention block that focuses on the face regions at a local and a global level, i.e. eyes region (upper face), mouth region (lower face) and whole face, and c) frame-attention block that helps to identify the most important frames in a video.

2. We use noisy student training [1] for semi-supervised learning, in which the trick involves the student to be deliberately noised when it trains on the combined labelled and unlabelled dataset.

3. We adapt one of the most successful networks, StarGAN [29], and propose StarGAN-EgVA to generate continuous facial emotions based on 2D emotional representations, i.e., valence and arousal (VA).

4. Comprehensive evaluation configurations for two benchmark datasets (AFEW 8.0 and Affect-Net-Vid) are carried out to provide a baseline for valid comparisons with other related work.

1.5 Thesis Outline

The remainder of the thesis is organized as follows:

1. Chapter 2 elaborates the background and related work of facial expression analysis from conventional methods to novel state-of-the-art deep learning related methods.

2. Chapter 3 presents the methodology used in the proposed system for facial expression recognition and the detailed process of synthetic data generation.

3. Chapter 4 provides different experiments and evaluations on two datasets, showing the effectiveness of the proposed methodology and results.

4. Chapter 5 provides the ablation studies, summarizes the merits of this work, notes the limitations, and discusses planned future work.

Chapter 2
Background

The strategies for facial expression analysis in the literature are elaborated in Section 2.1.The strategies for facial expression generation in the literature are elaborated in Section 2.2. We also briefly discuss semi-supervised learning and noisy student training in Section 2.3.

2.1 Facial Expression Recognition Task

According to Lopes et al. [30], the two main branches of facial expression recognition systems are as follows: those addressing static images [31, 32] and those addressing dynamic image sequences [33, 34]. Systems that work with static images consider only one still image at a time (frame-by-frame) and do not use temporal information. In contrast, those involving dynamic image sequences encoding a range of frames within a temporal window as an individual concentrate more on analyzing temporal variation [35]. This work adopts the dynamic-images based scheme.

Automatic FER systems generally receive the two kinds of expected input (still images or a sequence of frames) and output one of the seven basic universal emotions (i.e., angry, disgust, fear, happiness, sadness, surprise and neutral) that were classified by Ekman [36] in 1975 in a cross-cultural study on the existence of "universal categories of emotional expressions". In 1978, Ekman and Friesen developed the Facial Action Coding System (FACS) to describe observable facial muscle movements as action units (AU) [37]. This system taxonomizes these basic emotions by decomposing each facial expression into core AU and has been widely applied to FER tasks, providing a powerful tool for feature extraction.

2.1.1 Face Detection and Alignment

Face acquisition refers to detecting the face region in a frame (face detection). One of the most widely used face detectors was proposed by Viola and Jones [38] and is termed the VJ detector. The VJ detector can perform robust and efficient face detection in real time. However, it concentrates only on frontal faces. Quite a few works [39–41] indicate that this kind of detector may degrade significantly in real-world applications with larger visual variations of human faces even with more advanced features and classifiers. Besides the cascade structure, [39,42] introduce deformable part models (DPM) for face detection and achieve remarkable performance. However, they are computationally expensive and may usually require expensive annotation in the training stage. Recently, convolutional neural networks (CNNs) achieve remarkable progresses in a variety of computer vision tasks, such as image classification [43] and face recognition [44]. Inspired by the significant successes of deep learning methods in computer vision tasks, several studies utilize deep CNNs for face detection. Yang et al. [45] train deep convolution neural networks for facial attribute recognition to obtain high response in face regions which further yield candidate windows of faces. However, due to its complex CNN structure, this approach is time costly in practice. Li et al. [46] use cascaded CNNs for face detection, but it requires bounding box calibration from face detection with extra computational expense and ignores the inherent correlation between facial landmarks localization and bounding box regression.

Face alignment also attracts extensive research interests. Researches in this area can be roughly divided into two categories, regression-based methods [47,48] and template fitting approaches [49]. Recently, Zhang et al. [50] proposed to use facial attribute recognition as an auxiliary task to enhance face alignment performance using deep convolutional neural network. However, most of previous face detection and face alignment methods ignore the inherent correlation between these two tasks. Though several existing works attempt to jointly solve them, there are still limitations in these works. For example, Chen et al. [51] jointly conduct alignment and detection with random forest using features of pixel value difference. But, these handcraft features limit its performance a lot. Zhang et al. [20] use multi-task CNN to improve the accuracy of multi-view face detection, but the detection recall is limited by the initial detection window produced by a weak face detector.

In [2], the authors propose a new framework to integrate these two tasks using unified cascaded CNNs by multi-task learning (MTCNN). The proposed CNNs consist of three stages. In the first stage, it produces candidate windows quickly through a shallow CNN.

Then, it refines the windows by rejecting a large number of non-faces windows through a more complex CNN. Finally, it uses a more powerful CNN to refine the result again and output five facial landmarks positions. Thanks to this multi-task learning framework, the performance of the algorithm can be improved. In our work, the MTCNN is used for facial detection and alignment.

2.1.2 Classifiers for FER

A number of methods have been proposed on the AFEW 8.0 dataset [18] since the first EmotiW [52] challenge in 2013. Earlier approaches include non-deep learning methods such as multiple kernel learning [53], least-square regression on grassmanian manifold [54], and feature fusion with kernel learning [55], whereas recent approaches include deep-learning methods such as frame-attention networks [13], multiple spatial-temporal learning [20], and deeply supervised emotion recognition [19]. Although several methods [19–21, 56] have achieved impressive results on the AFEW 8.0 dataset, many have used ensemble (fusion) based methods and considered multiple modalities without commenting on the resources and time required to train such models. Spatial-temporal methods [21, 57] aim to model motion information or temporal coherency in the videos using 3D Convolution [58] or LSTM (Long short-term memory) [59]. However, owing to computational efficiency and the ability to treat sequential information with a global context, several studies [13, 60] related to facial expression recognition have successfully implemented attention-based methods by assigning a weight to each timestep in the video. Similarly, spatial self-attention has been used [60–62] as a means to guide the process of feature extraction and find the importance of each local image feature. Our model builds upon the spatial self-attention mechanism and additionally uses a channel-attention mechanism to exploit the differential effects of facial feedback signals from the upper-face and lower-face regions [63, 64].

2.2 Facial Expression Generation

Recent advancement of Generative Adversarial Network (GAN) based architectures has achieved impressive performance on static facial expression synthesis. Continuous affect synthesis, which has applications in generating videos and movies, is under-explored. We adapt one of the most successful networks, StarGAN, and propose StarGAN-EgVA to generate continuous facial emotions based on 2D emotional representations, *i.e.*, valence

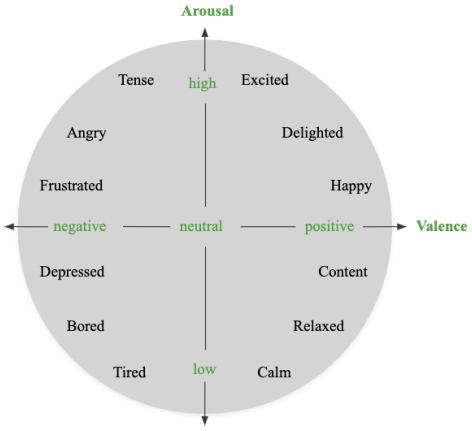

Figure 2.1. The 2D dimensional model for emotions, with Valence and Arousal axes.

and arousal (VA). We propose to utilize categorical emotions (*e.g.,* happy, sad) to guide the regression training on VA intensities so that the model learns both the domain-specific features and subtle changes introduced by different VA intensities. Our motivation is to train the GAN model on a static dataset, and generate a video-based dataset for supervised training.

Recent advancement of General Adversarial Network (GAN) [65] incorporating the functionalities of these sub-tasks has greatly improved performance on face emotion generation. StarGAN [29] introduces a novel adversarial architecture that enables mapping across different domains for facial expression synthesis. Although StarGAN can generate high-resolution images, it can only change a particular aspect of a face given a discrete domain type. It can thus only synthesize a discrete number of emotions such as anger, disgust, or happiness. To address this limitation, GANimation [66] introduced a novel conditioning GAN model based on Face Action Units (AUs) [17] annotations, which describe facial expressions in a continuous manifold. By combining different values of AUs, they claim that facial expression can transmit from one to another (*e.g.*, from fear to happy). Although their method shows some high-resolution images demonstrating continuous transition between different facial expressions, generating a smooth transition between any two facial expressions with the combination of a limited 30 attributes still suffers from artifacts. Particularly in unconstrained conditions, the accuracy of detecting certain AUs is not high enough, partially due to the lack of high-quality annotated data as such types of labels require annotation by human experts which incur a prohibitive cost. Moreover, AUs are not complete in the sense that it falls short in expressing all possible lip motion patterns present in the speech [67].

Another approach for generating continuous face emotions is to utilize the well-known

circumplex model of affect by Russell [68]. All emotion states can be specified in the two-dimensional Valence-Arousal (VA) plane, as shown in Figure 2.1. Different from the basic emotion theories that have discrete emotions and indirect AU combinations, any absolute values in the VA space can represent a specific affective state and, relative VA values can be used as a comparison between different states. This information can be exploited with the release of several datasets with VA intensities for affective computing, such as AFEW-VA [69, 70] and AffectNet [28].

VA-StarGAN [67] builds upon StarGAN [29] and adapts it to allow for continuous VA inputs rather than the original discrete inputs, and generates photo-realistic emotions. Though Kollias's method [67] is effective, there are a few limitations on the use of 2D valence and arousal model in continuous affect synthesis. Firstly, psychological experiments suggest that the two dimensions are inter-correlated and interdependent [71, 72], which indicates the distance between two points on the VA plane is not a true measure of the difference between the two emotion states. As a result, a path on the VA plane may not depict a smooth transition from one emotional state to the other. Secondly, certain basic emotions such as anger and disgust tend to overlap in the 2D VA space, making it difficult for learning models to correctly correlate facial expressions with emotional states. The learning model could get confused by input expressions with close VA intensities whereas with very different emotions. Thirdly, the annotation and prediction of VA intensities on face images are more challenging and error-prone than single emotion labels [28,73]. The valence and arousal model is more subtle and subjective as compared to the basic emotional model. Dimensional annotation of emotions not only requires consistent labeling over a series of images but also relies on a gold standard for different raters. There is evidence that predictive models performing well on one set of data do not generalize to another set, indicating the difficulty of setting a reliable and consistent standard [69]. Inconsistent VA annotation will make generated facial expression sequence unreliable and unstable.

In light of the aforementioned problems, we propose to incorporate emotion labels during the training of generative models on VA intensities, to serve as additional supervision to the less stable or reliable learning based solely on VA intensities. That is, in addition to the regression branch on VA, which is what has been done in [67], we also add a classification branch to StarGAN [29] to keep the model learning more reliable emotion-related features along with subtle changes from VA intensities. We also propose a trick for generating target-domain emotions during inference time to make the output face sequence more stable and smooth. Figure 2.2 shows some example sequence

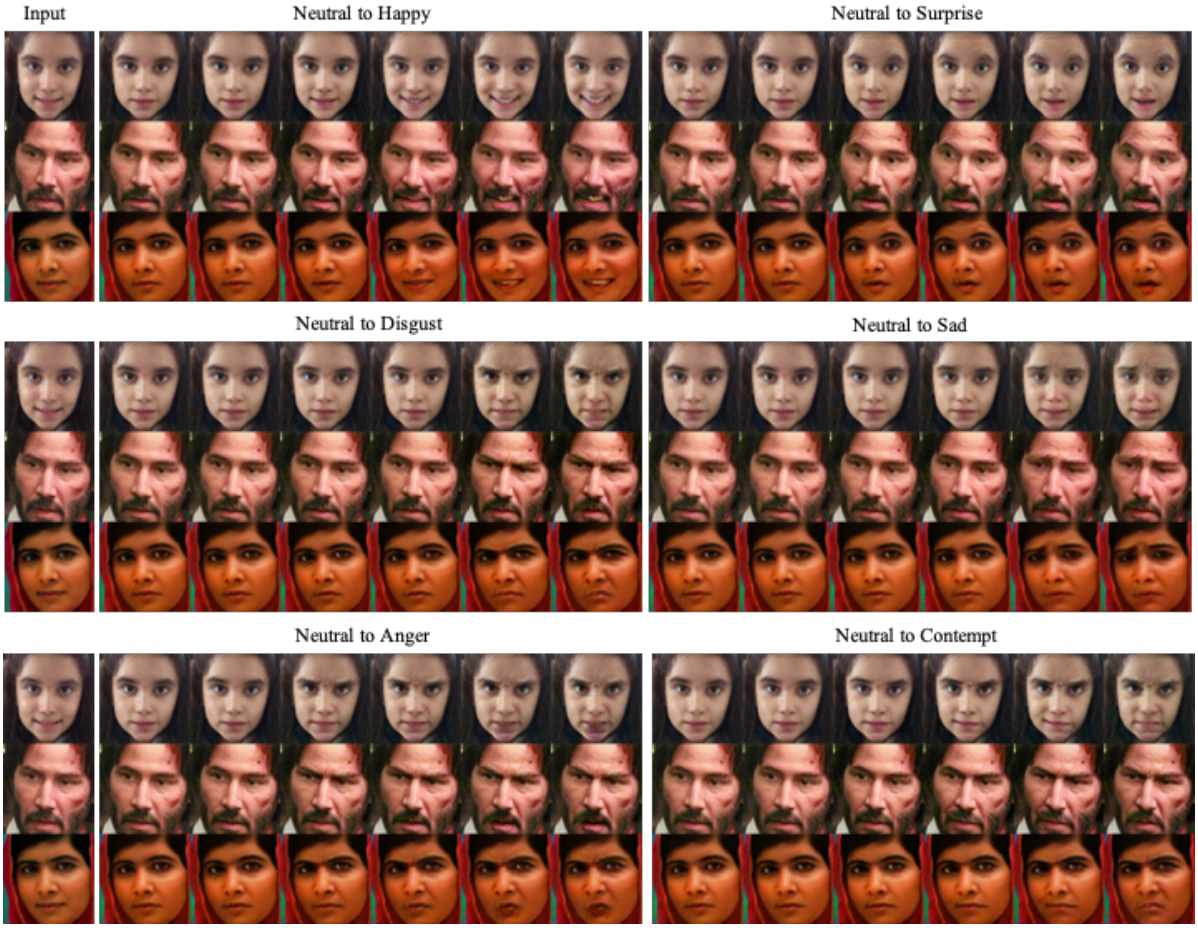

Figure 2.2. Continuous affect synthesis results. We propose an emotion-guided affect synthesis approach that generates a smooth sequence from one emotion to another. In these examples, the inputs are a single face image and a path on the 2D Valence-Arousal plane.

results generated by our proposed StarGAN-EgVA. The sequence of facial expressions is controlled by a path on the 2D VA-plane as inputs.

2.2.1 Generative Adversarial Network

GAN has achieved impressive results in image generation [74, 75], image-editing at the semantic level, representation learning [75, 76], and face image synthesis [75, 77]. GAN's architecture consists of two main components, namely the generator and the discriminator. Generator learns to generate fake images that are indistinguishable from authentic images and discriminator learns to distinguish between authentic and fake images. By training to reduce the *adversarial loss*, generator and discriminator competes with each other, eventually forcing the network to generate fake images that are indistinguishable from

authentic ones. We adopt *adversarial loss* to learn the mapping. Recent research [78, 79] has shown improved stability relying on the continuous Wasserstein Distance metric, which we also use to train our model.

Conditional GAN (cGAN) [80] allows us to generate new images conditioned on some auxiliary attributes such as class labels. There have been interesting methods exploring image-based conditioning as in image super-resolution [81], image-to-image translation [82], and realistic facial expression synthesis [83]. In our work, by providing conditional domain information (*e.g.*, VA values), the model flexibly translates input image to output image with desired facial expressions.

2.2.2 Image-to-Image Translation

Pix2pix [82] is among one of the first attempts to translate image to image with cGANs [80], but it requires paired data samples as inputs. To overcome this issue, unpaired image-to-image synthesis models were proposed [84–86]. CoGAN [87] and cross-modal scene networks use a weight-sharing strategy to learn a common representation across domains. Liu et al. [84] extend the above framework with a combination of variational autoencoders [88] and generative adversarial networks [65]. CycleGAN [85] and DiscoGAN [86] preserve key attributes between the input and the translated image by utilizing a cycle consistency loss. To alleviate the need of training different models for transferring between two domains, StarGAN [29] proposes a single model that learns the relations among several domains.

2.2.3 Affect Synthesis

Based on the achievement of the aforementioned frameworks in domain transfer, image translation, and image generation, face emotion synthesis has achieved high-resolution photo-realistic images in either discrete or continuous space [29, 66, 67, 89, 90]. StarGAN [29] generates natural-looking expressions in any of the eight common emotion categories while preserving personal identity and facial features. Kollias [67] builds on top of StarGAN and modifies it to allow for face generation based on valence and arousal intensities. Their results are impressive, but they only consider the use of VA annotations without considering the guidance of categorical emotions, which is insufficient, as we will demonstrate later. Almost at the same time, Kollias et al. [90] propose a quite different approach that leverages a 3D morphable model and 3D face deformation to generate image sequences given VA inputs. They performed dimensional annotation of

the large 4DFAB database [91] and generated mean faces on a discretized 2D affect space via clustering. While effective, the entire procedure is very complex and, the fact that they work on discretized clusters of VA points makes their method not truly continuous in terms of 2D emotion plane. They only generate discrete, albeit fine-grained, facial expressions.

Another approach of continuous affect synthesis is based on facial AUs. CDAAE [89] uses conditional difference adversarial auto-encoder for face emotion generation. It takes a static face image and generates an image with a target emotion or facial AUs label [17]. Similarly, GANimation [66] introduces a GAN conditioning scheme based on AU annotations, which describes the anatomical facial movements defining a human expression in a continuous manifold. AUs have limitations in a way that accuracy for detecting certain AUs is not high enough yet and, AUs cannot describe a continuous set of expressions as it lacks indication of certain muscle movements of the face.

2.3 Semi-Supervised Learning

The semi-supervised approach is effective in classification problems when the labelled training data is not sufficient. We use noisy student training [1] for semi-supervised learning, in which the trick involves the student to be deliberately noised when it trains on the combined labelled and unlabelled dataset. Input noise is added to the student model in the form of data augmentations, which ensures that different alterations of the same video should have the same emotion, hence making the student model more robust. Additionally, model noise is added in the form of dropout, which forces the student (single model) to match the performance of an ensemble model. Other techniques for semi-supervised learning include self-training [92,93], data-distillation [94] and consistency training [95, 96]. Self-training is similar to noisy student training, but it does not use or justify the role of noise in training a powerful student. Data-distillation uses the approach of strengthening the teacher using ensembles instead of weakening the student; however, a smaller student makes it difficult to mimic the teacher. Consistency training adds regularization parameters to the teacher model during training to induce invariance to input and model noise, resulting in confident pseudo-labels. However, such constraints lead to lower accuracy and a less powerful teacher [1].

Chapter 3
Methodology

3.1 Facial Expression Recognition

Our proposed methodology is divided into two phases, i.e. a) architecture implementation that defines the backbone network with the three-level attention mechanism, and b) semi-supervised learning.

3.1.1 Architecture

3.1.1.1 Backbone Network:

We use ResNet-18 [97] architecture as our backbone network, with minor modifications to increase its computational efficiency. Features from each residual block are combined to form the final feature vector (see Fig. 3.1). Hence, the final vector has a "multi-level knowledge" from all the residual blocks, ensuring more diverse and robust features. The model is first pre-trained on the FERPlus dataset [98]. Our input at frame-level is an image with nine channels (RGB channels from the face, eyes, and mouth region). To process them independently, the model uses group convolution [43] (groups = 3), i.e. it uses a different set of filters for each of the three regions to get the final output feature maps. Group convolution results in a lower computational cost since each kernel filter does not have to convolve on all the feature maps of the previous layer. Simultaneously, it allows data parallelism where each filter group is learning a unique representation and forms a global (face region) or local (eyes and mouth region) context vector from each frame of a video. To allow more filters per group, we increase the number of filters in each residual block, as shown in Fig. 3.1.

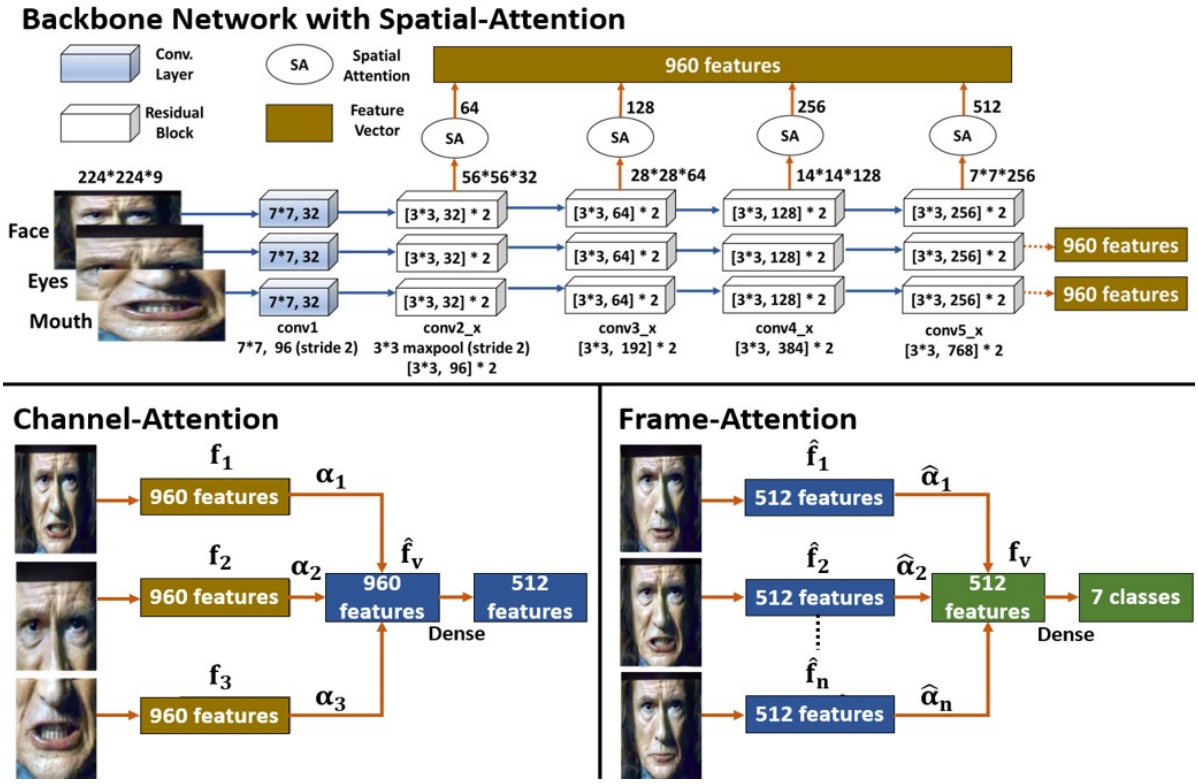

Figure 3.1. Figure shows the backbone network (ResNet-18) and the three-level attention mechanism. Inputs are first processed via Spatial-Attention, followed by Channel-Attention and finally by Frame-Attention.

3.1.1.2 Spatial-Attention:

A common approach in previous methods is a simple aggregation or average pooling of feature maps to form a fixed dimensional feature vector. However, we use spatial-attention [62] that concatenates the feature maps based on the attention weight it has been assigned. Let us assume the output from a residual block is of shape $C = H \times W \times D$ where H and W are the output height and width, and D is the number of output filters. This 3D tensor C is reshaped to a 2D matrix L of shape $R \times D$ where $R = H * W$. The spatial-attention mechanism takes the input matrix L and outputs a weight matrix M of shape $h \times R$ ($h = 2$, h is for multiple hops of attention). Each row of the output matrix represents a different hop of attention, and each column has normalized weights due to softmax (see Equation 3.1). The objective is to find the weighted average of R frame descriptors to obtain a vector v of length D (or $h * D$ with multiple hops).

$$M = softmax(W_{s2} tanh(W_{s1} L^T)) \tag{3.1}$$

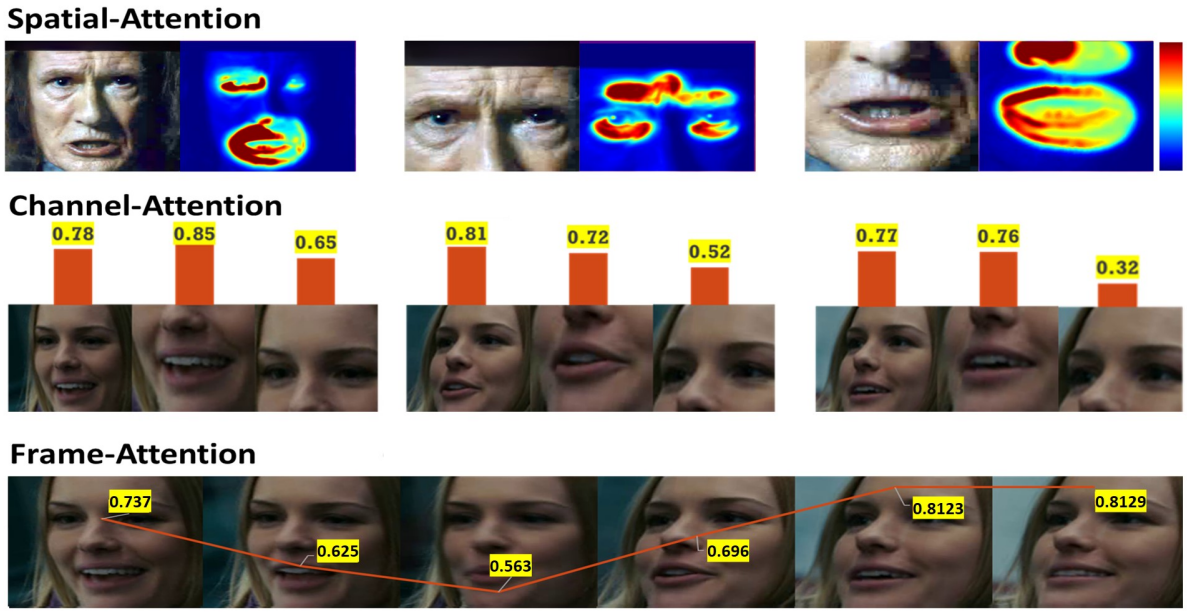

Figure 3.2. This figure shows how multi-level attention works in the proposed method. Spatial-attention (from last residual block) chooses the dominant feature maps from each region. Channel-attention picks the most important region that most clearly shows the target emotion. Frame-attention assigns the salient frames a higher weight.

$$v = flatten(M * L) \tag{3.2}$$

Equation 3.1 represents multi-head spatial-attention where W_{s1} is of shape $U \times D$ and W_{s2} is of shape $h \times U$ (U can be set arbitrarily). From this, we obtain flattened vector v using Equation 3.2. The spatial-attention module is applied on each residual block (see Fig. 3.1) and the output vectors are aggregated to obtain a final vector of length $l = 960$ each for face (f_1), eyes (f_2) and mouth (f_3) regions. The advantages of spatial attention can be seen in Fig. 3.2. While the feature vector from the face is encoded with a global context, the feature maps from the eyes and mouth region have additional information regarding the minute expressions such as furrowed brow or flared nostrils.

3.1.1.3 Channel-Attention:

Let f_1, f_2, and f_3 be the feature vectors obtained from the face, the eyes, and the mouth region respectively. We model the cross-channel interactions using a lightweight attention module. We use two fully-connected layers to obtain a weight α (Equation 3.3) for each channel group using which we obtain a weighted average $\hat{f}_v$ (Equation 3.4) of the three feature vectors. ReLU (Rectified Linear Unit) activation is used after the first layer to

capture non-linear interactions among the channels.

$$\alpha_i = \sigma(w^T(ReLU(W^T f_i))) \tag{3.3}$$

$$\hat{f}_v = \frac{\sum_{i=1}^{3} \alpha_i * f_i}{\sum_{i=1}^{3} \alpha_i} \tag{3.4}$$

where σ is the sigmoid activation function, w is a vector of length r (set arbitrarily), and W is a matrix of shape $l \times r$. In Fig. 3.2, we see that the model assigns more weight to the mouth region instead of the eyes region for an expression depicting happiness which is consistent with our findings that mouth region is more prominent for the happy category (Fig. 4.2).

3.1.1.4 Frame-Attention:

For a video having n frames, we obtain vector $\hat{f}_i$ of length $\hat{l}$ from each frame after the channel-attention module. Finally, we use frame-attention to assign the most discriminative frames a higher weight. Following a similar intuition as in channel-attention, we use two fully-connected layers to obtain a weight $\hat{\alpha}$ (Equation 3.5) for each frame using which we find a weighted average f_v (Equation 3.6) of the frame features.

$$\hat{\alpha_i} = \sigma(\hat{w}^T(ReLU(\hat{W}^T \hat{f}_i))) \tag{3.5}$$

$$f_v = \frac{\sum_{i=1}^{n} \hat{\alpha_i} * \hat{f}_i}{\sum_{i=1}^{n} \hat{\alpha_i}} \tag{3.6}$$

where $\hat{w}$ is a vector of length $\hat{r}$ (set arbitrarily), and $\hat{W}$ is a matrix of shape $\hat{l} \times \hat{r}$. Fig. 3.2 shows how the model assigns a higher weight to the frames which distinctively contains expression depicting happiness. The feature vector f_v is passed through a fully-connected layer to obtain the final 7-dimensional output.

3.1.1.5 Implementation Details:

We use weighted cross-entropy as our loss function where class weights are assigned based on number of training samples to alleviate the problem of unbalanced data. Additionally, M (Equation 3.1) is regularized by adding the frobenius norm of matrix $MM^T - I$ to the loss function which enforces multi spatial-attention to focus on different regions [62]. We use Adam optimizer with an initial learning rate of 1e-5 (reduced by 40% after every

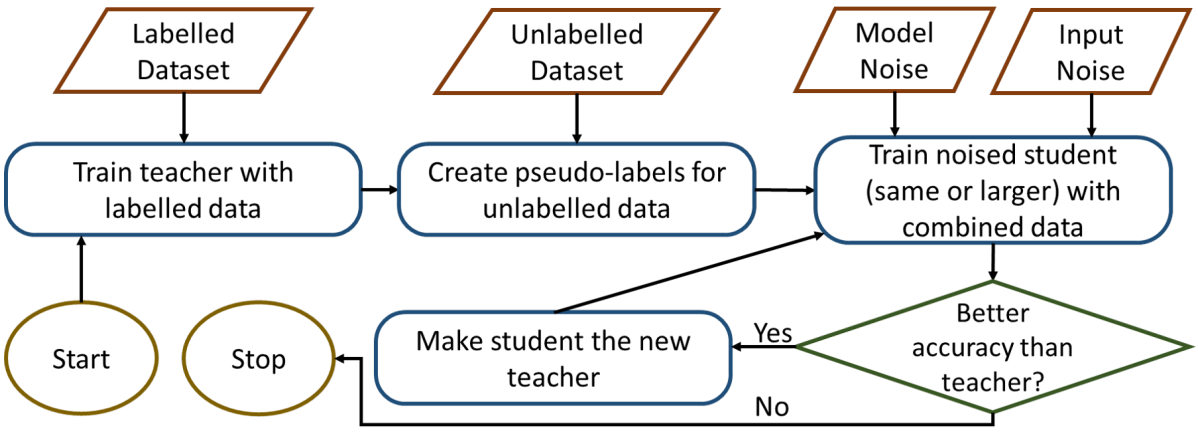

Figure 3.3. Semi-supervised algorithm is presented in the flow-chart. We also show an example video from AFEW 8.0 dataset where the frames underwent different augmentations. Predictions without iterative training are shown in red and predictions after iterative training are shown in black.

30 epochs) and the model is trained for 100 epochs. The training takes around 8 minutes for 1 epoch for AFEW 8.0 training dataset with two NVIDIA Tesla K80 cards.

3.1.2 Noisy Student Training [1]

Once the model is trained on the labelled set and the best possible model is obtained, we use it as a teacher model to create pseudo-labels on the subset of BoLD dataset that we collected. After generating the pseudo-labels, a student model (same size or larger than teacher) is trained on the combination of labelled and unlabelled dataset. While training the student model, we deliberately add noise in the form of random data augmentations and dropout (with 0.5 probability at the final hidden layer). Random data augmentations (using RandAugment [99]) include transformations such as brightness change, contrast change, translation, sharpness change and flips. RandAugment automatically applies $n \epsilon [2, 4]$ random operations with a random magnitude $m \epsilon [0, 9]$. After the noisy student is trained on the combined data, the trained student becomes the new teacher that generates new pseudo-labels for the unlabelled dataset. The iterative training continues until we observe a saturation in performance. From Fig. 3.3, we see how noisy training helps the student become more robust with the addition of noise. While the teacher may give different predictions for different alterations of the same video, the student is more accurate and stable with its predictions.

3.2 Facial Expression Generation

3.2.1 Problem Statement and Notations

Our goal is to train a generator G that can generate continuous facial expressions given valence and arousal values. The strategy of training is emotion-guided VA regression. The training images are in the RGB space, denoted as $x \in R^{H \times W \times 3}$. Each image has an emotion category label $c \in \{0, 1, 2, ..., 7\}$ which corresponds to seven emotions plus neutral, and valence v & arousal a values ranging in $[-1, 1]$. The seven emotions labeled from 1 to 7 are as follows: happy, sad, surprise, fear, disgust, anger, and contempt. A neutral expression is labeled as 0. The generator G of StarGAN-EgVA will translate an input image x to an output image y, whereas the discriminator D will differentiate between authentic and fake images, and also output the domain emotion category and VA estimates.

3.2.2 Network Architecture

We propose to modify StarGAN [29] by adding additional input VA channel to G, and also output VA channel to D. The network architecture is shown in Figure 3.4. As in a usual GAN framework, it has two subnetworks: generator G and discriminator D.

The generator G starts with three 2D convolution layers, each followed by instance normalization [100] and ReLU. It then incorporates six consecutive Residual blocks [97] for feature extraction and learning. Right after Residual blocks, two 2D transpose convolution layers are applied to upsample feature maps back to the input shape. Again each of these two upsampling layers is followed with instance normalization and ReLU. A final convolution layer followed by tanh activation is used to generate the final image output. Inputs of G consist of three parts: input image x, emotion label c, and VA values v, a. Note that both emotion label and VA values are replicated spatially and concatenated with the input image, so that it ends with an input shape $(x, c, va) \in R^{H \times W \times (3+8+2)} = R^{H \times W \times 13}$. As illustrated in 3.4, during training, G is used twice: a) first, authentic image x and target emotion c , target VA values $v\ a$ are fed into G to generate an output image $G(x, c\ , v\ a\ ) \rightarrow y$, b) next the generated output image y is rendered back to a reconstructed image x , with original emotion label c and original VA values va: $G(y, c, va) \rightarrow x$. A cycle consistency loss [86] is applied to x and x , to ensure that the generator is only translating domain-related features of the inputs.

The discriminator D references the idea of PatchGANs [82] that consists of six

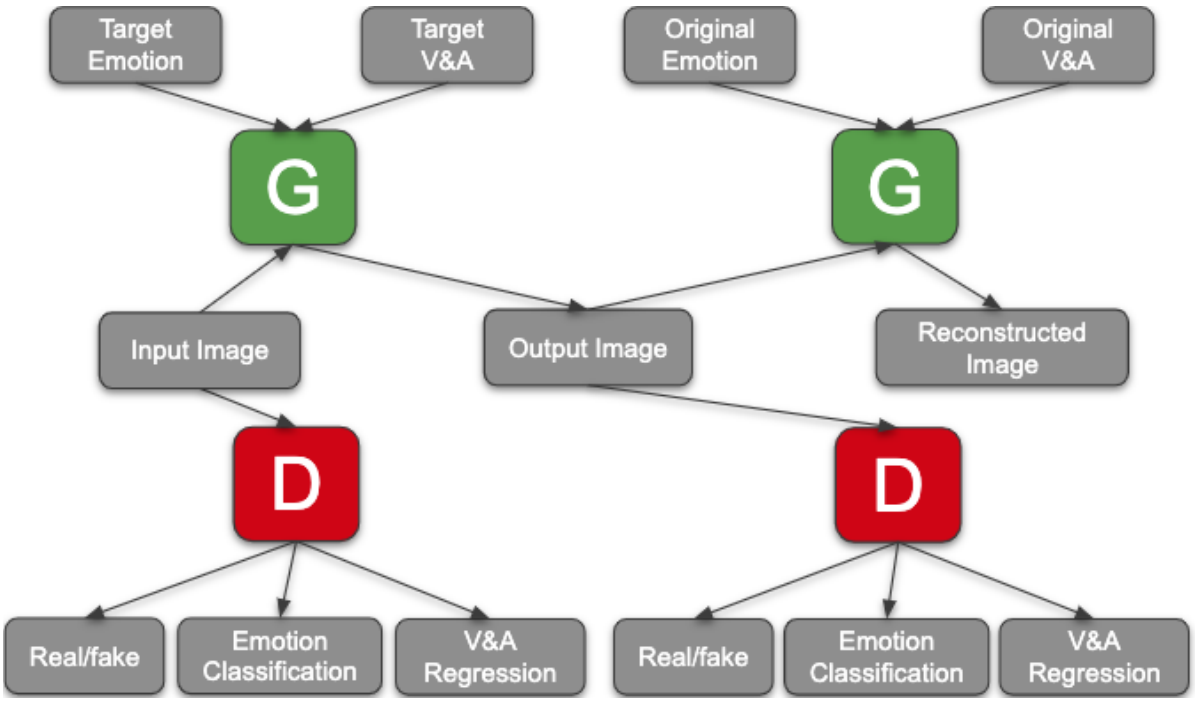

Figure 3.4. The network architecture of StarGAN-EgVA.

convolutional downsampling layers (kernel size 4, stride 2, and pad 1) with each followed by LeakyReLU. On top of them, there are three output channels, which is different from the original two channels as in StarGAN. The outputs are generated from convolution layers without activations. They are the output label for either authentic or fake, predicted emotion label, and estimated VA intensities. During training, discriminator D is applied to both the input image x and generated image y to enforce photo-realistic and domain fulfillment between them.

3.2.3 Loss Functions

We introduce regression loss in addition to the loss functions of StarGAN and, change the original domain classification loss to emotion classification loss, so that the model learns from both discrete emotions and continuous VA intensities.

Adversarial Loss. The generator G takes an input image x, target emotion label c , and target VA values $v\ a$, to generate an output image y. The generated image y is then passed to the discriminator D that determines the realistic probability of the image. We define the adversarial loss for the generator to be:

$$L_{\text{Gadv}} = -E_{x,c',v'a'}[D_{r,f}(G(x, c\ , v\ a\ )]\ , \tag{3.7}$$

where $D_{r,f}$ is the branch of D that outputs the probability of image to be real or fake.

We then pass x and y to discriminator D, to strengthen its capability to tell whether an image is real or fake. The adversarial loss of D is thus as follows:

$$L_{\text{Dadv}} = -E_{x,c,va}[D_{r,f}(x)] + E_{x,c',v'a'}[D_{r,f}(G(x, c\,, v\ a\ ))]\ , \tag{3.8}$$

where the first part is the negative classification loss on real image x and the second part is the classification loss on generated image y.

Regression Loss. The quality of affect synthesis is regulated using G by a regression loss between input target VAs $v\ a$ and its regression estimates on generated image y. Similarly, we apply regression loss on the VA output channel of D. It evaluates on the original image x and its corresponding VA values va. The loss functions for G and D are formulated as:

$$L_{\text{Greg}} = E_{x,c',v'a'}[-D_{reg}(v\ a\ |G(x, c\,, v\ a\ ))]\ , \tag{3.9}$$

$$L_{\text{Dreg}} = E_{x,c,va}[-D_{reg}(va|x)]\ , \tag{3.10}$$

where D_{reg} is the regression output branch of D that estimates VA values.

Classification Loss. An additional emotion category input is added to G and classification output branch is added to D to help regulate the training on regression. As seen in Section 3.2.4.1, valence, and arousal intensities are not reliable, when compared to emotion categories, as they are more subtle and subjective. To help the generator learn domain-specific features and enforce reliable training of the discriminator, we propose to apply a classification loss to both of them:

$$L_{\text{Gcls}} = E_{x,c',v'a'}[-D_{\text{cls}}(c\ |G(x, c\,, v\ a\ ))]\ , \tag{3.11}$$

$$L_{\text{Dcls}} = E_{x,c,va}[-D_{\text{cls}}(c|x)]\ , \tag{3.12}$$

where D_{cls} is the classification output branch of D that predicts emotion category labels.

Overall Objectives. In addition to the aforementioned loss functions, two other loss terms are inherited from StarGAN: reconstruction loss L_{rec} on G in the form of cycle consistency loss [86], and Wasserstein GAN Loss L_{Dgp} on D with gradient penalty [101]. The overall objective functions for G and D can thus be summarized as follows:

$$L_G = L_{\text{Gadv}} + \lambda_{\text{rec}} L_{\text{Grec}} + \lambda_{\text{reg}} L_{\text{Greg}} + \lambda_{\text{cls}} L_{\text{Gcls}}\ , \tag{3.13}$$

$$L_D = L_{\text{Dadv}} + \lambda_{\text{gp}} L_{\text{Dgp}} + \lambda_{\text{reg}} L_{\text{Dreg}} + \lambda_{\text{cls}} L_{\text{Dcls}}\ , \tag{3.14}$$

where λ_{reg} and λ_{cls} are hyper-parameters that control the relative weights of regression versus classification, which are shared by both losses on G and D. If the weight on classification is set to zero, the model will train completely on regression, which is the case as in [67]. The weights are carefully chosen so that classification serves as supervision on regression but not overwhelmingly control the training. The other two hyper-parameters λ_{rec} and λ_{gp} are fixed throughout our experiments, with λ_{rec} controlling the weight on image reconstruction by G and λ_{gp} controlling the update speed of D in the form of gradient penalty.

3.2.4 Experiments

3.2.4.1 Dataset

We evaluate the results of our proposed model on the benchmark dataset: AffectNet [28], which contains around 1M facial images. It is one of the largest datasets that contain both categorical and dimensional emotion labels. Images were downloaded from the Internet by querying certain emotion-related keywords. Around half $(420,000)$ of the images are manually annotated by twelve human experts, into seven basic emotions as well as dimensional (valence and arousal) states within the range $[-1, 1]$. Another half of the dataset is automatically labeled by a predictive neural network trained on manually annotated images. To measure the agreement among the human annotators, 36,000 images were annotated by two annotators. There was only a 60.7% agreement in terms of categories, and the RMS errors on valence and arousal were 0.340 and 0.362 respectively, indicating that affective state labeling is a subjective and challenging task. We only use manually labeled images because they have lower RMS errors compared to automatically labeled images (0.394 on valence and 0.402 on arousal). Also, we have removed images that don't have emotion category labels in the eight categories. This reduces the dataset size to around $200K$. Figure 3.5 displays the VA values of the samples from our training set, together with their basic emotion labels. It is easy to observe that a) certain emotion categories, especially those in the second quadrant, heavily overlap in the 2D space; b) the data is unbalanced in terms of high arousal vs. low arousal.

3.2.4.2 Baseline Models

We adopt VA-StarGAN [67], which trains StarGAN on VA inputs, and GANimation [66] as the baseline models, to compare with our proposed emotion-guided VA regression strategy. VA-StarGAN uses both concordance correlation coefficient (CCC) and mean

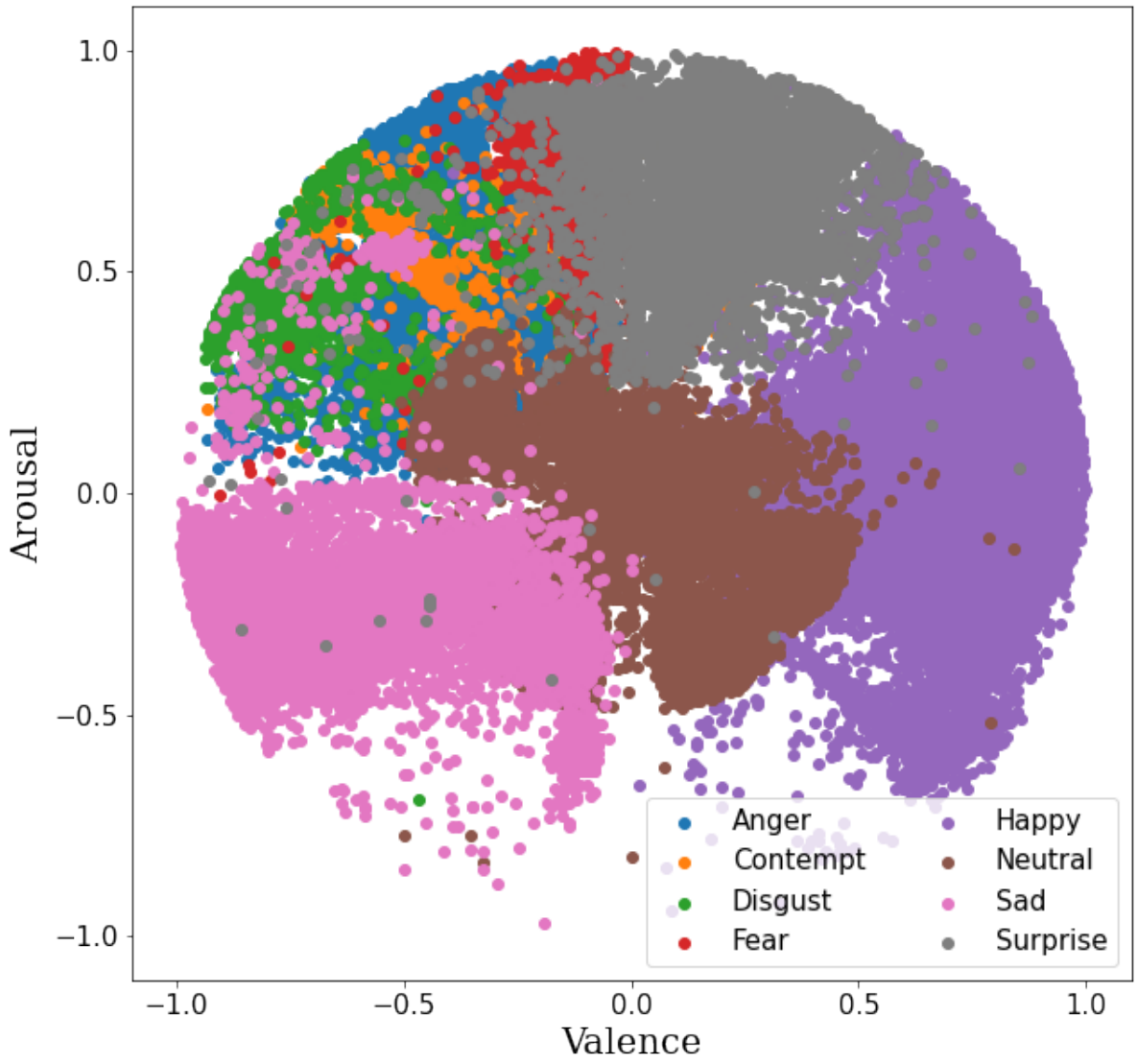

Figure 3.5. The 2D Valence-Arousal space of AffectNet dataset.

squared error (MSE) as the regression loss functions, from which we only replicate MSE because both of them have achieved comparable results. As mentioned earlier, StarGAN-EgVA can be easily converted to VA-StarGAN by setting the weight on classification λ_{cls} to be zero. For GANimation, we have to change the inputs from AUs to VAs in order to compare.

3.2.4.3 Training

We train StarGAN-EgVA in two parallel settings: a) train the model in emotion-guided manner by setting $\lambda_{\text{cls}} = 0.5$ and $\lambda_{\text{reg}} = 20$; b) train the model solely on VA by setting

$\lambda_{\text{cls}} = 0$ and $\lambda_{\text{reg}} = 40$, as the baseline VA-StarGAN. The weights on λ_{reg} are chosen to make sure that the summed loss from regression and classification stays roughly the same for both settings, and balances among other terms in the overall loss functions including reconstruction loss ($\lambda_{\text{rec}} = 10$) of G and gradient penalty term ($\lambda_{\text{gp}} = 10$) on D.

During training, Adam [102] optimizer with $\beta_1 = 0.5$ and $\beta_2 = 0.999$ is applied. We trained both StarGAN-EgVA models for $200,000$ iterations with a batch size of 16. An initial learning rate of 0.0001 is set for both G and D, and decays after every 50,000 iterations. The weights of D is updated five times before every update of G, following the strategy of [79]. GANimation was trained using the default parameters, for 200,000 iterations with batch size 16.

3.2.4.4 Testing

We propose a strategy during testing to infer the emotion label from VA positions on the 2D plane. Because our goal is to generate continuous emotional expressions based on a sequence of VA values, emotion labels of each VA point are not assumed to be known beforehand. The complete set of inputs for generator G, however, includes emotion labels. To alleviate this problem, one direct solution is to keep all emotion labels at a fixed value like 0 or 1. This method is simple but it conflicts with the idea of emotion supervision that we expect the model to have learned correct emotion-specific features from categorical emotions as well as corresponding continuous VA intensities. Here, we present a more reasonable solution: inferring the emotion label from VA positions on the 2D emotion plane. During training, each emotion label is converted to a one-hot vector of length eight. To get a similar vector for a VA point at testing, we first count the number of sample points from the training dataset corresponding to all eight emotion categories within the vicinity of the point (radius of 0.05). We then normalize the counts into a unit vector and use it as the emotion label. In this way, we can get a rough guess on the right emotions for any VA points. More importantly, surrounding-based inference guarantees a smooth transition from one point to another, if we take a dense sampling strategy along the path.

3.2.5 Facial Expression Generation Results

In this section we evaluate our proposed model on the benchmark dataset - AffectNet and show the results in terms of both quantitative as well as qualitative aspects. Both these perspectives shows the superiority of our model compared to the two baseline methods.

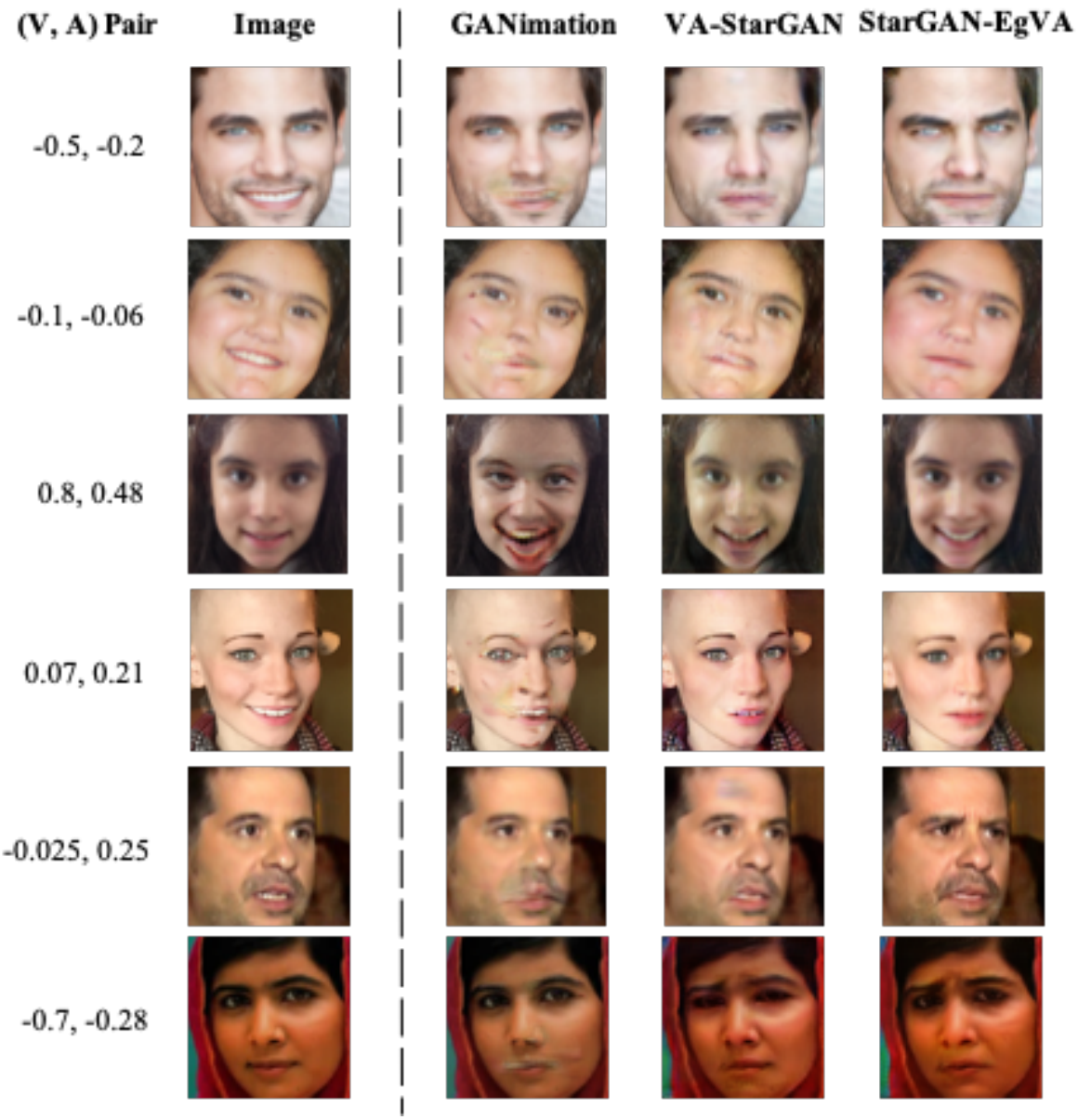

Figure 3.6. Visual comparison of the generated sequences in facial expression synthesis from GANimation, VA-StarGAN, and StarGAN-EgVA.

3.2.6 Qualitative Results

In this sub-section we demonstrate the continuous facial emotion sequence generated by our proposed network - StarGAN-EgVA to be more photo-realistic and consistent when compared to the generated images by baseline models, with examples in Figures 3.6 and 3.7, respectively. Figure 3.6 shows some examples of the two baseline models and our StarGAN-EgVA. It is quite evident that the results from GANimation [66] are the

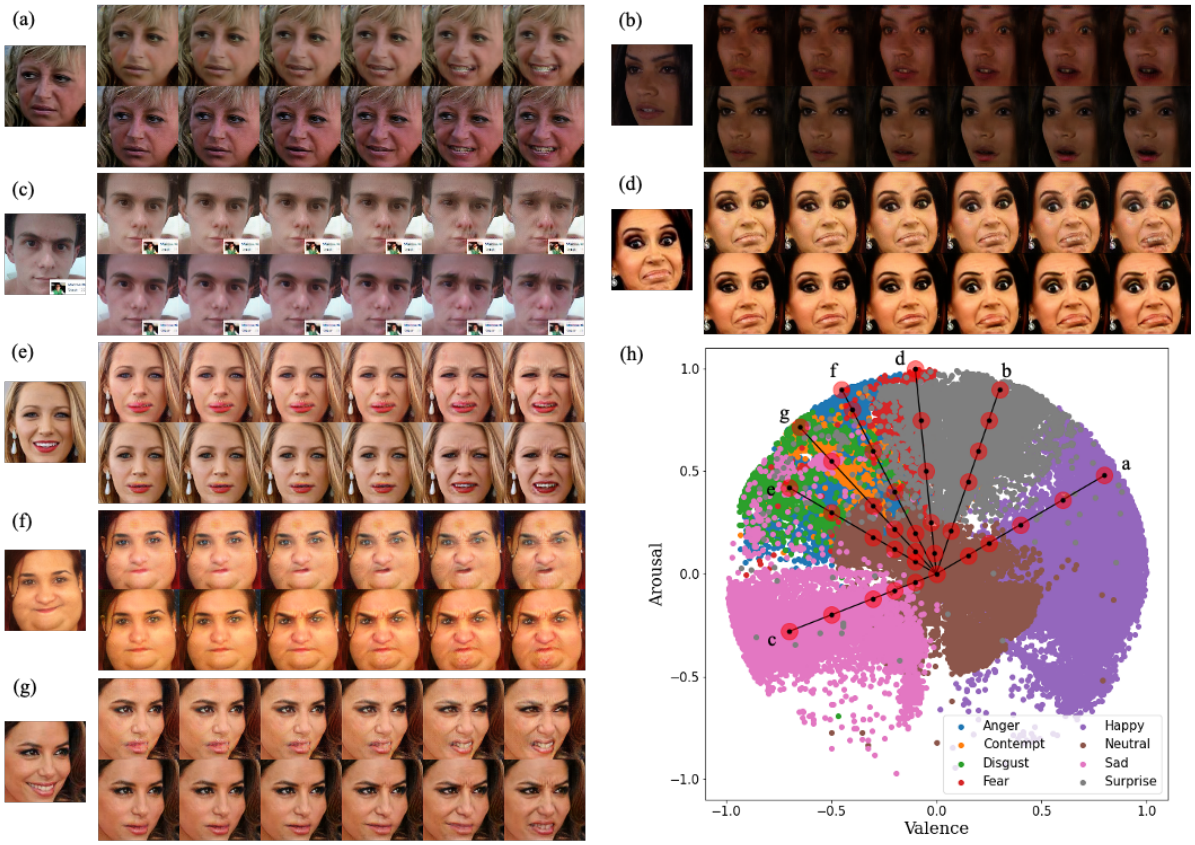

Figure 3.7. Visual comparison of consistency over facial expression sequence results of VA-StarGAN and StarGAN-EgVA. From a to g, seven paths originating from the center (neutral) are taken to generate seven basic emotions in the order of happy, surprise, sad, fear, disgust, anger, and contempt. The single images on the left are the input images. First rows are the results of VA-StarGAN and second rows are from StarGAN-EgVA. All six sample points on a specific path is shown in h, with the red circle denoting the vicinity of the point to infer its emotion category.

least visually appealing, whereas the results from VA-StarGAN [67] are quite close to our StarGAN-EgVA, but still suffer from subtle artifacts. Evident differences between our model and VA-StarGAN can be seen in the samples shown in second, third, fourth and the last rows of Figure 3.6. Generally, the facial expressions from our model are more vivid and coherent to the input images, when compared to those from VA-StarGAN. We believe this is because by incorporating domain emotion guidance, our model can learn domain-specific features and avoid extreme expressions which was missing from VA-StarGAN. Hence, our model not only learns the subtle changes in expressions of different valence and arousal intensities, but also actively adjusts itself with domain emotion inputs during training.

We further compare the quality of the generated affect sequence in terms of consistency

Table 3.1. MSE evaluation of VGGFace trained on AffectNet dataset and augmented by GANimation, VA-StarGAN, and our StarGAN-EgVA.

Augmented by	Valence	Arousal
None	0.367	0.188
GANimation	0.361	0.187
VA-StarGAN	0.354	0.185
StarGAN-EgVA	0.346	0.185

Table 3.2. MSE evaluation of VGGFace on eight test sets labeled with different basic emotions.

Augmented by	Neutral	Happy	Sad	Surprise	Fear	Disgust	Anger	Contempt
None	0.276	0.274	0.279	0.276	0.278	0.281	0.278	0.280
GANimation	0.271	0.270	0.274	0.271	0.273	0.276	0.273	0.275
VA-StarGAN	0.268	0.266	0.270	0.268	0.270	0.272	0.270	0.272
StarGAN-EgVA	0.264	0.263	0.266	0.265	0.266	0.268	0.266	0.267

within the sequence and also with the input image. Figure 3.7 shows some example sequences generated by VA-StarGAN and StarGAN-EgVA. Here, we decide not to include the results of GANimation because they are not comparable to the results of the two StarGAN-based models. Each of the seven sets of sequences represents six points on a linear path in the VA plane, starting from neutral (0,0) to a domain emotion category: happy (0.8,0.48), surprise (0.3,0.9), sad (-0.7,-0.28), fear (-0.1,1.0), disgust (-0.7,0.42), anger (-0.45,0.9), and contempt (-0.65,0.715). As seen in Figure 3.7, the face expression sequence generated by VA-StarGAN either starts badly (b, e, g - sub-figures) or ends badly (c, d - sub-figures). Our results, on the other hand, are more consistent throughout the sequence and agree well with the input image (due to domain emotion guidance). Again, we would attribute this consistency to our implementation of emotion-guided VA regression, and also the strategy of inferring emotion labels during testing, which ensures a stable and smooth transition from one VA point to the other. To further validate our qualitative results we also present quantitative results in the next subsection. This will assure the superiority of our model both visually as well as quantitatively.

3.2.7 Quantitative Results

We demonstrate the superiority of the proposed model over two baselines by utilizing them as a strategy of data augmentation in training a separate emotion recognition network VGGFace [103]. We modify VGGFace to predict valence and arousal intensities for face images and train it on four different settings: a) AffectNet dataset (a subset of

200,000 images), b) AffectNet dataset and 100,000 images generated by GANimation given random VA inputs, c) AffectNet dataset and 100,000 images generated by VA-StarGAN, and d) AffectNet dataset and 100,000 images generated by the proposed StarGAN-EgVA. To ensure that data augmentation does not introduce noise to the original dataset, we sample VA inputs from existing VA points in the AffectNet training set. For each input image, we randomly shift its VA values within the range of $[-0.1, 0.1]$ to get the target VA. It makes the target emotion not deviate too far away from source emotion and thus minimizes possible artifacts. The input images for augmentation are also randomly selected from AffectNet at a fixed distribution over eight basic emotions for three models to make the three sets of generated data mutually comparable and serve as a complementary to original AffectNet data. A common test set with roughly 2,600 images randomly selected from the original AffectNet dataset is used to evaluate the performance of these trained VGGFace models. Table 3.1 shows the MSE regression loss values on both valence and arousal. It can be seen that all three generative approaches contribute to the learning of the dimensional affect recognition model, especially on the recognition of valence. AffectNet dataset together with augmented images from StarGAN-EgVA yields the lowest regression loss over other approaches. The reason why MSE loss on arousal does not improve much is that most emotions (except neutral and sad) in the dataset are high-arousal emotions, and the values of arousal highly overlap among them. Take happy and angry as an example, though they have a very different valence, they usually share the same arousal intensities ranging in $[0, 1]$. This makes it harder for the learning model to improve its recognition over arousal. On the other hand, the fact that the distribution of arousal intensities are more concentrated than valence limits the range of prediction for VGGFace, which leads to a smaller MSE loss.

We also evaluate the MSE scores averaged on valence and arousal, on sub testing sets originally labeled with different basic emotions, to check if our proposed model can generate photo-realistic facial expressions with various emotions. Table 3.2 displays average MSE scores on eight separate test sets. Our proposed model yields the lowest MSE scores among all eight emotion categories, demonstrating its effectiveness of representing diverse emotion states in the 2D VA plane. Although the difference in the MSE might not seem too much, however the small margin of improvement is attributed to the stable and smooth transition from one VA point to another which was missing in other baselines and could be a bottleneck for all future works. Our work is the first to do so by incorporating emotion classification (exploiting additional information) on the VA regression training of generative models.

Chapter 4
Results

4.1 Dataset

In this section, we first describe the datasets that we use in our experiments, followed by the pre-processing pipeline.

4.1.1 Labelled Sets

AFEW 8.0 (Acted Facial Expression in the Wild) [18] contains videos with seven emotion labels, i.e. anger (197 samples), neutral (207 samples), sad (179 samples), fear (127 samples), surprise (120 samples), happiness (212 samples), and disgust (114 samples) from different movies. The train set consists of 773 video samples (46,080 frames), and the validation set consists of 383 video samples (21,157 frames). The results are reported on the validation set since the test set labels are only available to EmotiW challenge [22] participants. Some of the example frames are shown in Fig. 4.1.

Affect-Net-Vid is a synthetically created dataset using StarGAN-EgVA. The theory for generating continuous image sequences is already explained in Figure 3.7. We create a video by choosing the initial frame randomly from a linear path starting from neutral (0,0) to a domain emotion category. A biased random function is used that chooses a number from the far end to ensure that the generated video does not contain high number of neutral frames. For the neutral category, we use a length threshold of 0.1 from the center for the linear path. The frames are generated with a distance gap of 0.005 to emulate the continuous effect of a video. The training dataset contains 1546 video sequences with controlled variable length (20-100 frames each randomly) divided

into seven categories, i.e anger (394 samples), neutral (414 samples), sad (358 samples), fear (254 samples), surprise (240 samples), happiness (424 samples), and disgust (228 samples). We use 70% of the dataset for training and the rest of the dataset for training and the rest is used for testing.

4.1.2 Unlabelled Set

BoLD (Body Language Dataset) [104] contains videos selected from the publicly available AVA dataset [105], which contains a list of YouTube movie IDs. While the gathered videos are annotated based on body language, the videos having a close shot of the face instead of the whole or partially-occluded body are unlabelled. To create an AFEW-like subset from the BoLD dataset, we impose two conditions to automatically validate a video. Firstly, a video should have $f\,(\geq 30)$ such consecutive frames where only one actor's face is detected by MTCNN (Multi-task Cascaded Convolutional Networks) [2]. Secondly, the bounding box of the face detected using MTCNN should exceed an occupied area threshold for the majority of those f frames. If the video satisfies the above two conditions, a smaller video with those f frames is added to the unlabelled dataset. Using this procedure, we create a subset of 3450 videos (224,258 frames) from the original BoLD dataset. Some of the examples gathered are shown in Fig. 4.1.

4.1.3 Pre-Processing

Previous work [13, 20] have used CNN-based detector provided by dlib [106] for face alignment. However, the alignment of faces is highly dependent on accurate detection of facial landmarks and CNN-based detector provided by dlib is not reliable for 'in-the-wild' conditions (especially non-frontal faces). We use MTCNN [2] for face detection and alignment. If MTCNN detects multiple faces in a frame, the face with the largest bounding box is selected. After obtaining the facial landmarks, its alignment is corrected using the angle between the line connecting the landmark points of the eyes and the horizontal line. After detection and alignment, the cropped face is resized to 224*224 pixels, which is the input size of our model.

We use the landmarks given by MTCNN to isolate the mouth (lower face) and eyes (upper face) region. The upper face is isolated using the eyes landmarks with the desired left eye normalized co-ordinates being (0.2, 0.6) and right-eye co-ordinates being (0.8,

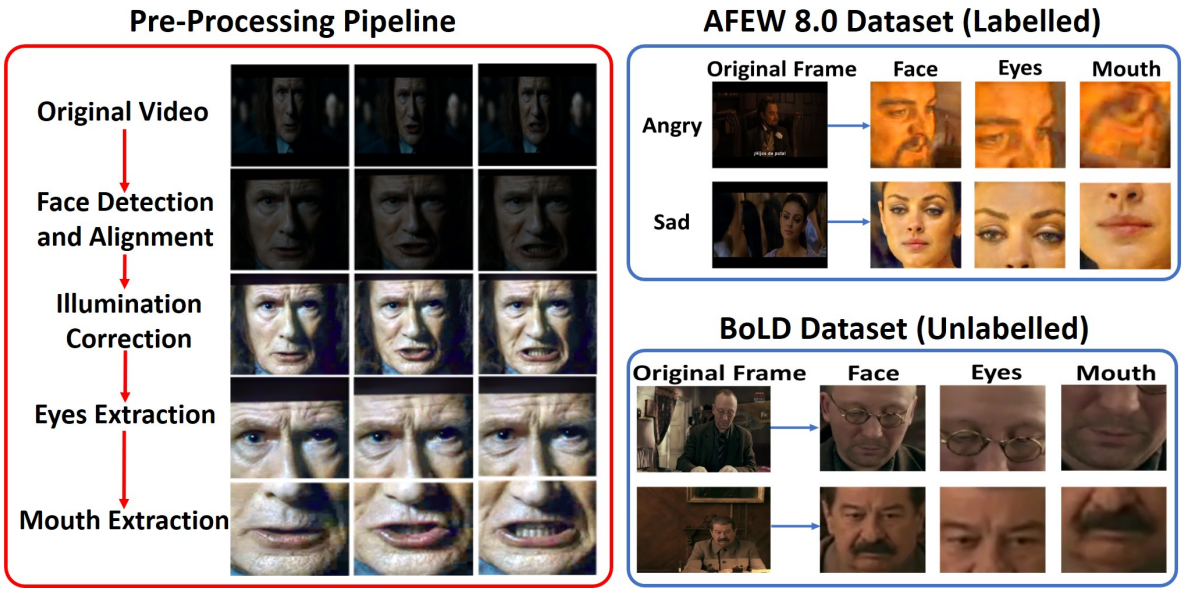

Figure 4.1. The pre-processing steps mainly include face detection and alignment (MTCNN [2]), illumination correction (Enlighten-GAN [3]) and landmark-based cropping. Examples from labelled dataset (AFEW 8.0) and unlabelled dataset (BoLD dataset) are shown. As seen in the figure, only videos with a close shot of the face are selected from the BoLD dataset.

0.6) in the new frame, which is enough to occlude the lower-half of the face in almost all frames (Fig. 4.1). A similar procedure is used for occluding the upper-half of the face and isolating the mouth region using left-mouth and right-mouth landmarks. All landmark-based crops are again resized to 224*224 pixels.

As addressed earlier, some of the categories of emotions are often captured in the dark in movies, which requires an illumination correction step. Several methods have been suggested for illumination normalization such as gamma correction [107,108], Difference of Gaussians (DoG) [109] and histogram equalization [110,111] which are effective for facial expression recognition. However, these methods tend to amplify noise, tone distortion, and other artefacts. Hence, we use a state-of-the-art pre-trained deep learning model, i.e. Enlighten-GAN [3] (U-Net [112] as generator) which provides appropriate results (Fig. 4.1) with uniform illumination and suppressed noise.

4.2 Evaluation on AFEW 8.0 Dataset

Fig. 4.2 shows the results of processing individual regions (without group convolution and channel attention) using only the AFEW 8.0 dataset (for training and testing), along with the proposed methodology. Our objective is to explore a) if upper face region and

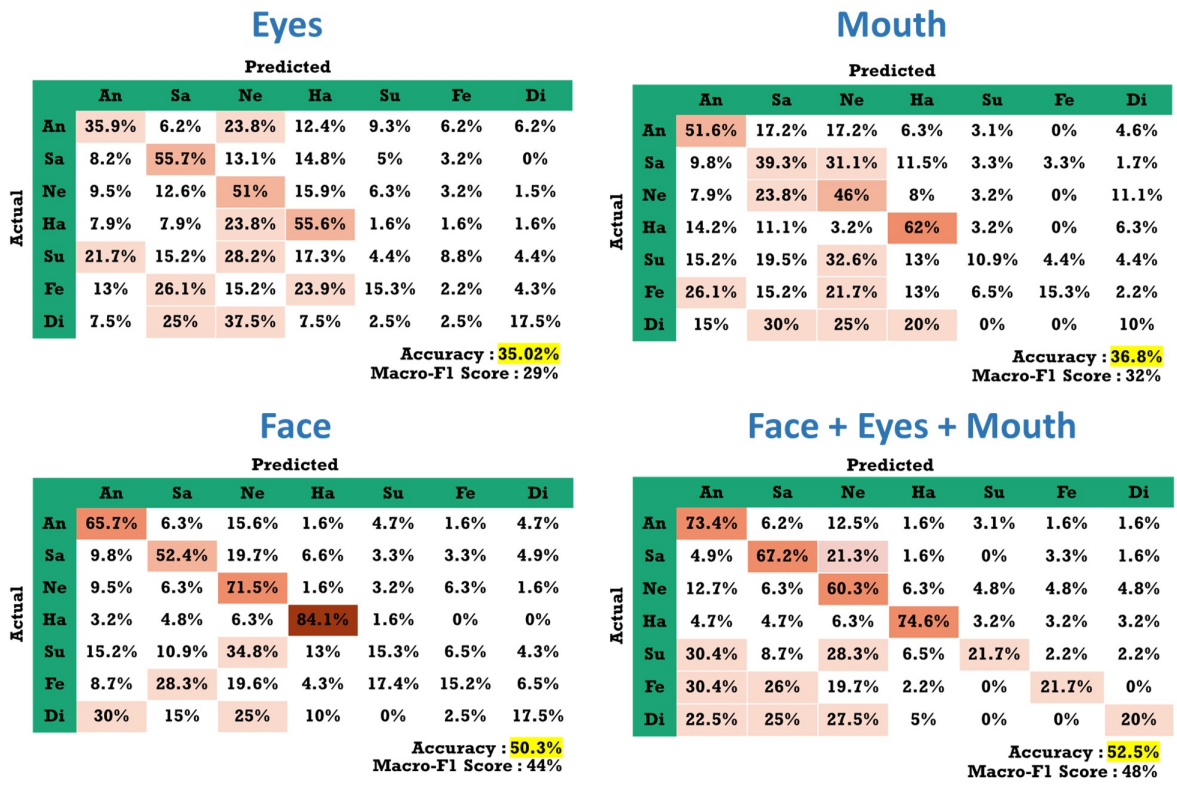

Figure 4.2. This figure shows the confusion matrices, the accuracies, and the macro f1 scores achieved on the AFEW 8.0 dataset using different regions of the face. The proposed model (Face + Eyes + Mouth) achieves the highest accuracy. An=Angry, Sa=Sad, Ne=Neutral, Ha=Happy, Su=Surprise, Fe=Fear, Di=Disgust.

lower face regions have different feedback signals that dominate different categories of emotions, and b) if isolating the regions and processing them independently leads to an increase of accuracy. As seen in the confusion matrix (Fig. 4.2), the eyes region is better than the mouth region in the prediction of sadness and disgust categories. Intuitively, the squinted eyes expression in disgust and the droopy eyelids or furrowed eyebrows expression in sadness makes the eyes region pronounced. On the other hand, the mouth region is comparatively better with categories that require lip movements like happiness, anger, and surprise. Overall, 52.50% accuracy is achieved using the proposed model, which is slightly better than the model that only uses faces. Furthermore, we see a significant increase in the macro f1 score when we include the eyes and mouth region along with faces indicating that the predictions are comparatively more unbiased for the seven categories (an advantage for noisy student training). The proposed model is still biased against fear, surprise, and disgust categories, but performs better than several existing methods [20, 27, 113] where the reported accuracies for these categories are close

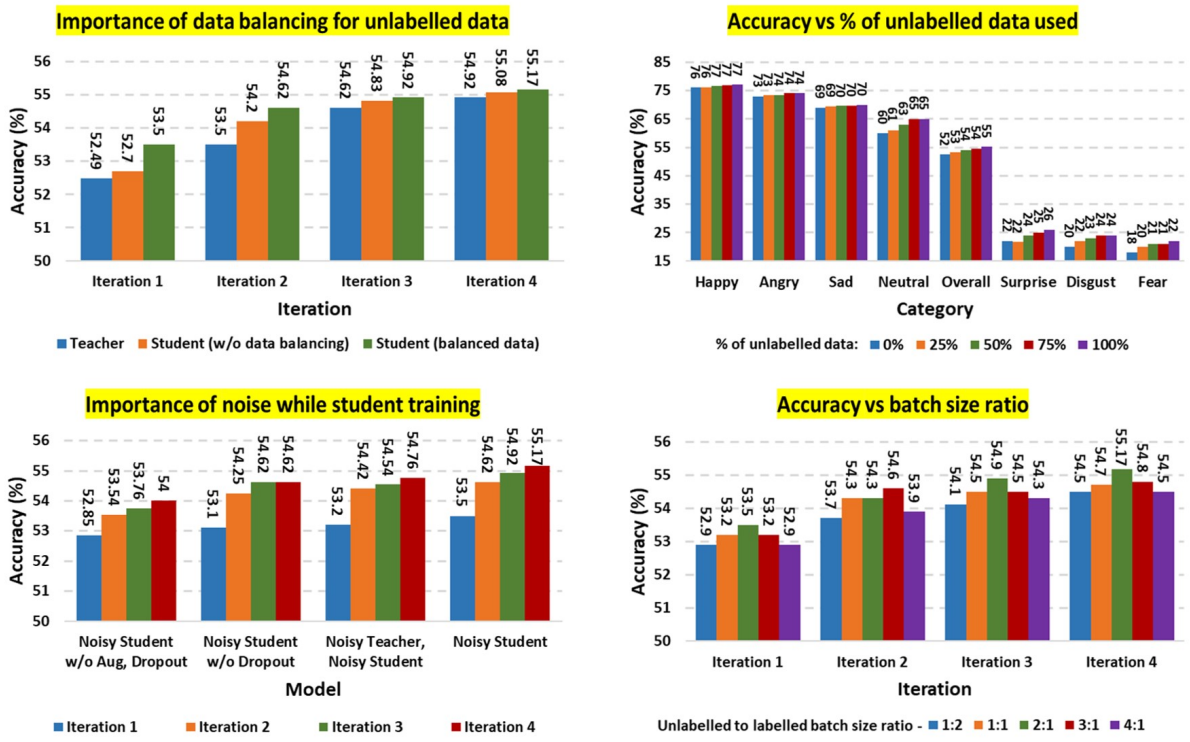

Figure 4.3. This figure shows the experimental results of noisy student training for four iterations using AFEW 8.0 and BoLD datset.

to 0%.

4.3 Evaluation on All Datasets

We report our experimental results on the AFEW 8.0 dataset and on the Affect-Net-Vid dataset after training on all datasets combined. The model is trained first using AFEW 8.0 dataset and Affect-Net-Vid dataset combined using semi-supervised learning and, later is trained using all datasets combined using noisy student training.

4.3.1 Data Balancing

Since the model is biased, the number of pseudo-labels in the unlabelled dataset for some categories is smaller than in other categories. We try to match the distribution of the training set by duplicating images of fear, disgust, and surprise categories. Additionally, images of angry, happy, and neutral classes are filtered out based on confidence scores. Fig. 4.3 shows that balancing the pseudo-labels leads to better accuracy in each iteration compared to the student model without data balancing.

4.3.2 Unlabelled Dataset Size

As stated in the original paper [1], using a large amount of unlabelled data leads to better accuracy. After data balancing, we use a fraction of the BoLD dataset and report the accuracy after several iterations of training until the performance saturates (see Fig. 4.3). For AFEW 8.0 dataset, we observe that using the whole unlabelled training set is better as opposed to using just a fraction of the dataset. Fig. 4.3 shows a steady increase in all categories and overall accuracy with an increase in data size after four iterations of noisy-student training.

4.3.3 Importance of Noise

Noise helps the student to be more robust than the teacher, as addressed in Sec. **??**. The accuracy only reaches 54% on the AFEW 8.0 dataset without noise in student training. However, we achieve an accuracy of 55.17% after noisy training, which shows that input and model perturbations are vital while training the student. Additionally, Fig. 4.3 shows that it is better when the pseudo-labels are generated without noise, i.e. the teacher remains as powerful as possible.

4.3.4 Batch Size Ratio

When training on combined data, a batch of labelled images and a batch of unlabelled images are concatenated for each training step. If the batch sizes of labelled and unlabelled sets are equal, the model will complete several epochs of training on labelled data before completing one epoch of training on the BoLD dataset due to its larger size. To balance the number of epochs of training on both datasets, the batch size of the unlabelled set is kept higher than the labelled set. Fig. 3.3 shows that a batch size ratio of 2:1 or 3:1 is ideal for noisy-student training.

4.3.5 Comparison With Other Methods

We evaluate our model on the labelled datasets (AFEW 8.0 and Affect-Net-Vid) and show a comparison with the existing state-of-the art-methods (Table 4.1). The results are shown for four versions of our model: a) trained and evaluated only on AFEW 8.0 dataset, b) trained and evaluated only on Affect-Net-Vid dataset, c) trained and evaluated on both datasets (AFEW 8.0 and Affect-Net-Vid) without iterative training and, d) trained on all datasets (AFEW 8.0 + Affect-Net-Vid + BoLD) using noisy student training

Table 4.1. We compare our results to the top-performing *single* models evaluated on the AFEW 8.0 dataset and on the Affect-Net-Vid dataset.

AFEW 8.0		Affect-Net-Vid	
Models	**Acc.**	**Models**	**Acc.**
CNN-RNN (2016) [57]	45.43%	-	-
DSN-HoloNet (2017) [114]	46.47%	-	-
DSN-VGGFace (2018) [19]	48.04%	-	-
VGG-Face + LSTM (2017) [21]	48.60%	-	-
VGG-Face (2019) [60]	49.00%	-	-
ResNet-18 (2018) [115]	49.70%	-	-
FAN (2019) [13]	51.18%	-	-
DenseNet-161 (2018) [56]	51.44%	VGG-Face + BLSTM (2018) [20]	74.35%
Our Model (AFEW 8.0)	52.49%	FAN (2019) [13]	74.56%
VGG-Face + BLSTM (2018) [20]	53.91%	Our Model (Affect-Net-Vid)	76.15%
Our Model (w/o iter.training)	57.75%	Our Model (w/o iter.training)	78.77%
Our Model (iter. training)	**59.25%**	**Our Model (iter. learning)**	**79.5%**

and evaluated on both datasets. On the AFEW 8.0 dataset, we achieve an accuracy of 52.5% by just training on the AFEW 8.0 dataset. This shows that the model is efficient compared to other models without using additional datasets. After combined training on AFEW 8.0 and Affect-Net-Vid, our model performs significantly better than all the other models by a big margin. Lastly, we train the model using noisy student training by combining the labelled and the unlabelled datasets which results in an additional increase in accuracy on the validation set. When comparing to existing best single models, our proposed method improves upon the current state-of-the-art [20] by 5.34%. Compared to static-based CNN methods that aim to combine frame scores for video-level recognition, we achieve a significant improvement of 8.07% over the previous baseline [56].

For Affect-Net-Vid, we retrain the two best state-of-the-art [13, 56] models and evaluate their performance on the validation set. All the three versions of our model 4.1 beats the previous work in terms of accuracy.

Chapter 5
Discussion and Conclusion

We propose a multi-level attention model for video-based facial expression recognition. Our contribution is a cost-effective single model that achieves better performance with state-of-the-art models using three broad strategies. Firstly, we use attention with multiple sources of information to capture spatially and temporally important features, which is a computationally economical alternative to the fusion of multiple learning models. Secondly, we use self-training to overcome the lack of labelled video datasets for facial expression recognition. Lastly, we present the emotion-guided training of StarGAN on valence and arousal intensities and its application on continuous affect synthesis. We use the StarGan-EgVA model to generate continuous 3D affective sequences which leads to a new synthetic in-the-wild facial expression dataset.

We conduct a comparison of performance and speed of the existing state-of-the-art models including fusion methods (only visual modality) with our proposed model. Several methods that show higher validation accuracy have significantly higher computational demand which may be impractical for real-time world applications. For instance, [115] uses an ensemble of 50 models with the same architecture and yet attains a 52.2% validation accuracy. Similarly, [20, 56] use a combination of multiple deep learning models where each model has a higher computational cost than ours. We measure the computational complexity of state-of-the-art methods using FLOPS (Floating point operations) and results show that our method is the most optimal based on performance and speed (Fig. 5.1).

Our baseline model is ResNet-18 where the video-level feature vector is an unweighted average of all the frame-level feature vectors. Without sophisticated pre-processing, the baseline achieves an accuracy of 47.5%. To better understand the significance of each component, we record our results after every change to the baseline model (Table 5.1). Significant improvements are observed when features are concatenated from multiple

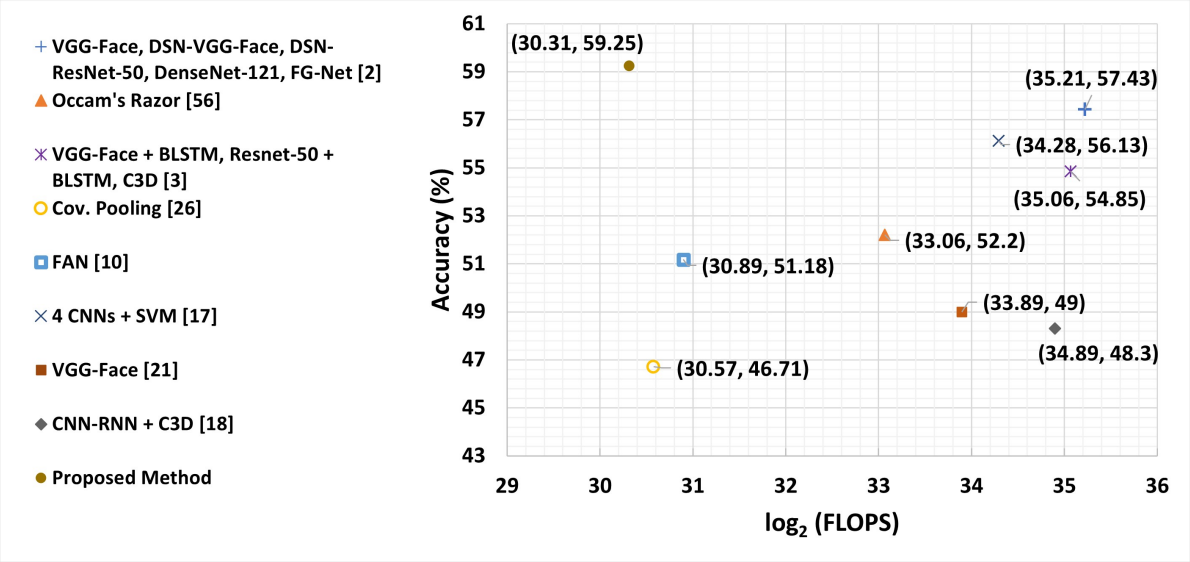

Figure 5.1. Comparison of performance (in accuracy) vs computational cost (in FLOPS - Floating point operations) of state-of-the-art models evaluated on AFEW 8.0 dataset. FLOPS for the models are estimated values based on the backbone network unless explicitly specified by the authors. Most optimal models will be closer to the top-left corner.

Table 5.1. This table shows the ablation studies conducted with AFEW 8.0 dataset. *Component Importance* shows the increase in accuracy with the addition of each component separately. *Noisy Student Training* shows the increase in accuracy with each loop of iterative learning and the effect of using a larger student.

Component Importance		**Noisy Student Training**		
Component	**Acc.**	**Iteration**	**Student**	**Acc.**
ResNet-18 (Baseline)	47.5%	0	-	57.75%
+ MTCNN, Enlighten-GAN (Sec. 4.1.3)	48.3%	1	ResNet-18	58.35%
+ Features from all blocks (Sec. 3.1.1.1)	49.3%		ResNet-34	58.35%
+ Spatial-Attention (Sec. 3.1.1.2)	50.3%	2	ResNet-18	58.6%
+ Multiple Regions (Sec. 3.1.1.1)	51.2%		ResNet-34	58.4%
+ Channel-Attention (Sec. 3.1.1.3)	51.7%	3	ResNet-18	58.92%
+ Frame-Attention (Sec. 3.1.1.4)	52.5%		ResNet-34	58.7%
+ Affect-Net-Vid Training (Sec. 4.1.1)	57.75%	4	ResNet-18	59.25%
+ Iteration 1 - Self-training (Sec. 3.1.2)	58.35%		ResNet-34	58.92%
+ Iteration 2 - Self-training (Sec. 3.1.2)	58.6%	5	ResNet-18	59.25%
+ Iteration 3 - Self-training (Sec. 3.1.2)	58.92%		ResNet-34	59.25%
+ Iteration 4 - Self-training (Sec. 3.1.2)	59.25%			

residual blocks using spatial-attention, and when frame features are combined from multiple regions using group convolution and channel-attention.

Additionally, Table 5.1 shows the increase in validation accuracy with each loop of

iterative learning. As suggested by [1], noisy student learning may perform better if the student is larger in size than the teacher. Since ResNet-34 [97] has a comparatively larger capacity, we report its results besides ResNet-18 as the student model for each iteration. As seen in Table 5.1, our results do not show improvement when ResNet-18 in our student model is replaced with a larger backbone. A possible explanation is that the unlabelled dataset used by [1] is a hundred times larger than the labelled dataset and using a student with higher capacity may have resulted in better performance. On the contrary, our unlabelled dataset is only two times larger than the labelled dataset. Gathering additional unlabelled samples and using a larger student may result in a further increase in accuracy on the AFEW 8.0 dataset.

Bibliography

[1] Xie, Q., E. Hovy, M.-T. Luong, and Q. V. Le (2019) "Self-training with Noisy Student improves ImageNet classification," *arXiv preprint arXiv:1911.04252*.

[2] Zhang, K., Z. Zhang, Z. Li, and Y. Qiao (2016) "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, **23**(10), pp. 1499–1503.

[3] Jiang, Y., X. Gong, D. Liu, Y. Cheng, C. Fang, X. Shen, J. Yang, P. Zhou, and Z. Wang (2019) "Enlightengan: Deep light enhancement without paired supervision," *arXiv preprint arXiv:1906.06972*.

[4] Jain, A. K. and S. Z. Li (2011) *Handbook of face recognition*, vol. 1, Springer.

[5] Darwin, C. and P. Prodger (1998) *The expression of the emotions in man and animals*, Oxford University Press, USA.

[6] Freiwald, W. A., D. Y. Tsao, and M. S. Livingstone (2009) "A face feature space in the macaque temporal lobe," *Nature neuroscience*, **12**(9), pp. 1187–1196.

[7] Ekman, R. (1997) *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*, Oxford University Press, USA.

[8] Huang, D., C. Shan, M. Ardabilian, Y. Wang, and L. Chen (2011) "Local binary patterns and its application to facial image analysis: a survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, **41**(6), pp. 765–781.

[9] Morik, K., P. Brockhausen, and T. Joachims (1999) *Combining statistical learning with a knowledge-based approach: a case study in intensive care monitoring*, *Tech. rep.*, Technical Report.

[10] Kluckner, S., G. Pacher, H. Grabner, H. Bischof, and J. Bauer (2007) "A 3D teacher for car detection in aerial images," in *2007 IEEE 11th International Conference on Computer Vision*, IEEE, pp. 1–8.

[11] Zhang, L. and D. Tjondronegoro (2011) "Facial expression recognition using facial movement features," *IEEE transactions on affective computing*, **2**(4), pp. 219–229.

[12] Fasel, B. and J. Luettin (2003) "Automatic facial expression analysis: a survey," *Pattern recognition*, **36**(1), pp. 259–275.

[13] Meng, D., X. Peng, K. Wang, and Y. Qiao (2019) "frame attention networks for facial expression recognition in videos," in *2019 IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 3866–3870.

[14] Lucey, P., J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews (2010) "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *2010 ieee computer society conference on computer vision and pattern recognition-workshops*, IEEE, pp. 94–101.

[15] Lyons, M., S. Akamatsu, M. Kamachi, and J. Gyoba (1998) "Coding facial expressions with gabor wavelets," in *Proceedings Third IEEE international conference on automatic face and gesture recognition*, IEEE, pp. 200–205.

[16] Pantic, M., M. Valstar, R. Rademaker, and L. Maat (2005) "Web-based database for facial expression analysis," in *2005 IEEE international conference on multimedia and Expo*, IEEE, pp. 5–pp.

[17] Langner, O., R. Dotsch, G. Bijlstra, D. H. Wigboldus, S. T. Hawk, and A. Van Knippenberg (2010) "Presentation and validation of the Radboud Faces Database," *Cognition and Emotion*, **24**(8), pp. 1377–1388.

[18] Dhall, A., R. Goecke, S. Lucey, and T. Gedeon (2012) "Collecting large, richly annotated facial-expression databases from movies," *IEEE multimedia*, (3), pp. 34–41.

[19] Fan, Y., J. C. Lam, and V. O. Li (2018) "Video-based emotion recognition using deeply-supervised neural networks," in *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pp. 584–588.

[20] Lu, C., W. Zheng, C. Li, C. Tang, S. Liu, S. Yan, and Y. Zong (2018) "Multiple spatio-temporal feature learning for video-based emotion recognition in the wild," in *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pp. 646–652.

[21] Vielzeuf, V., S. Pateux, and F. Jurie (2017) "Temporal multimodal fusion for video emotion classification in the wild," in *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pp. 569–576.

[22] Dhall, A. (2019) "EmotiW 2019: Automatic Emotion, Engagement and Cohesion Prediction Tasks," in *2019 International Conference on Multimodal Interaction*, pp. 546–550.

[23] Littlewort, G., M. S. Bartlett, I. Fasel, J. Susskind, and J. Movellan (2004) "Dynamics of facial expression extracted automatically from video," in *2004 Conference on Computer Vision and Pattern Recognition Workshop*, IEEE, pp. 80–80.

[24] Shan, C., S. Gong, and P. W. McOwan (2009) "Facial expression recognition based on local binary patterns: A comprehensive study," *Image and vision Computing*, **27**(6), pp. 803–816.

[25] Knyazev, B., R. Shvetsov, N. Efremova, and A. Kuharenko (2017) "Convolutional neural networks pretrained on large face recognition datasets for emotion classification from video," *arXiv preprint arXiv:1711.04598*.

[26] Tang, Y. (2013) "Deep learning using linear support vector machines," *arXiv preprint arXiv:1306.0239*.

[27] Acharya, D., Z. Huang, D. Pani Paudel, and L. Van Gool (2018) "Covariance pooling for facial expression recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 367–374.

[28] Mollahosseini, A., B. Hasani, and M. H. Mahoor (2017) "Affectnet: A database for facial expression, valence, and arousal computing in the wild," *IEEE Transactions on Affective Computing*, **10**(1), pp. 18–31.

[29] Choi, Y., M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo (2018) "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8789–8797.

[30] Lopes, A. T., E. de Aguiar, A. F. De Souza, and T. Oliveira-Santos (2017) "Facial expression recognition with convolutional neural networks: coping with few data and the training sample order," *Pattern Recognition*, **61**, pp. 610–628.

[31] Kumar, P., S. Happy, and A. Routray (2016) "A real-time robust facial expression recognition system using HOG features," in *2016 International Conference on Computing, Analytics and Security Trends (CAST)*, IEEE, pp. 289–293.

[32] Islam, R., K. Ahuja, S. Karmakar, and F. Barbhuiya (2016) "SenTion: A framework for sensing facial expressions," *arXiv preprint arXiv:1608.04489*.

[33] Bartlett, M. S., B. Braathen, G. Littlewort-Ford, J. Hershey, I. Fasel, T. Marks, E. Smith, T. J. Sejnowski, and J. R. Movellan (2001) *Automatic analysis of spontaneous facial behavior: A final project report*,

Tech. rep., Technical Report UCSD MPLab TR 2001.08, University of California, San Diego.

[34] Essa, I. A. and A. P. Pentland (1997) "Coding, analysis, interpretation, and recognition of facial expressions," *IEEE transactions on pattern analysis and machine intelligence*, **19**(7), pp. 757–763.

[35] Sariyanidi, E., H. Gunes, and A. Cavallaro (2014) "Automatic analysis of facial affect: A survey of registration, representation, and recognition," *IEEE transactions on pattern analysis and machine intelligence*, **37**(6), pp. 1113–1133.

[36] Ekman, P. and W. V. Friesen (2003) *Unmasking the face: A guide to recognizing emotions from facial clues*, vol. 10, Ishk.

[37] Friesen, E. and P. Ekman (1978) "Facial action coding system: a technique for the measurement of facial movement," *Palo Alto*, **3**(2), p. 5.

[38] Viola, P. and M. J. Jones (2004) "Robust real-time face detection," *International journal of computer vision*, **57**(2), pp. 137–154.

[39] Yang, B., J. Yan, Z. Lei, and S. Z. Li (2014) "Aggregate channel features for multi-view face detection," in *IEEE international joint conference on biometrics*, IEEE, pp. 1–8.

[40] Pham, M.-T., Y. Gao, V.-D. D. Hoang, and T.-J. Cham (2010) "Fast polygonal integration and its application in extending haar-like features to improve object detection," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 942–949.

[41] Zhu, Q., M.-C. Yeh, K.-T. Cheng, and S. Avidan (2006) "Fast human detection using a cascade of histograms of oriented gradients," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, IEEE, pp. 1491–1498.

[42] Yan, J., Z. Lei, L. Wen, and S. Z. Li (2014) "The fastest deformable part model for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2497–2504.

[43] Krizhevsky, A., I. Sutskever, and G. E. Hinton (2012) "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105.

[44] Sun, Y., X. Wang, and X. Tang (2014) "Deep learning face representation by joint identification-verification," *arXiv preprint arXiv:1406.4773*.

[45] Yang, S., P. Luo, C.-C. Loy, and X. Tang (2015) "From facial parts responses to face detection: A deep learning approach," in *Proceedings of the IEEE international conference on computer vision*, pp. 3676–3684.

[46] Li, H., Z. Lin, X. Shen, J. Brandt, and G. Hua (2015) "A convolutional neural network cascade for face detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5325–5334.

[47] Burgos-Artizzu, X. P., P. Perona, and P. Dollár (2013) "Robust face landmark estimation under occlusion," in *Proceedings of the IEEE international conference on computer vision*, pp. 1513–1520.

[48] Cao, X., Y. Wei, F. Wen, and J. Sun (2014) "Face alignment by explicit shape regression," *International Journal of Computer Vision*, **107**(2), pp. 177–190.

[49] Cootes, T. F., G. J. Edwards, and C. J. Taylor (1998) "Active appearance models," in *European conference on computer vision*, Springer, pp. 484–498.

[50] Zhang, Z., P. Luo, C. C. Loy, and X. Tang (2014) "Facial landmark detection by deep multi-task learning," in *European conference on computer vision*, Springer, pp. 94–108.

[51] Chen, D., S. Ren, Y. Wei, X. Cao, and J. Sun (2014) "Joint cascade face detection and alignment," in *European conference on computer vision*, Springer, pp. 109–122.

[52] Dhall, A., A. Kaur, R. Goecke, and T. Gedeon (2018) "Emotiw 2018: Audio-video, student engagement and group-level affect prediction," in *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pp. 653–656.

[53] Sikka, K., K. Dykstra, S. Sathyanarayana, G. Littlewort, and M. Bartlett (2013) "Multiple kernel learning for emotion recognition in the wild," in *Proceedings of the 15th ACM on International conference on multimodal interaction*, pp. 517–524.

[54] Liu, M., R. Wang, Z. Huang, S. Shan, and X. Chen (2013) "Partial least squares regression on grassmannian manifold for emotion recognition," in *Proceedings of the 15th ACM on International conference on multimodal interaction*, pp. 525–530.

[55] Chen, J., Z. Chen, Z. Chi, and H. Fu (2014) "Emotion recognition in the wild with feature fusion and multiple kernel learning," in *Proceedings of the 16th International Conference on Multimodal Interaction*, pp. 508–513.

[56] Liu, C., T. Tang, K. Lv, and M. Wang (2018) "Multi-feature based emotion recognition for video clips," in *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pp. 630–634.

[57] Fan, Y., X. Lu, D. Li, and Y. Liu (2016) "Video-based emotion recognition using CNN-RNN and C3D hybrid networks," in *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pp. 445–450.

[58] Tran, D., L. Bourdev, R. Fergus, L. Torresani, and M. Paluri (2015) "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 4489–4497.

[59] Hochreiter, S. and J. Schmidhuber (1997) "Long short-term memory," *Neural computation*, **9**(8), pp. 1735–1780.

[60] Aminbeidokhti, M., M. Pedersoli, P. Cardinal, and E. Granger (2019) "Emotion recognition with spatial attention and temporal softmax pooling," in *International Conference on Image Analysis and Recognition*, Springer, pp. 323–331.

[61] Fang, Y., J. Gao, C. Huang, H. Peng, and R. Wu (2019) "Self Multi-Head Attention-based Convolutional Neural Networks for fake news detection," *PloS one*, **14**(9).

[62] Lin, Z., M. Feng, C. N. d. Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio (2017) "A structured self-attentive sentence embedding," *arXiv preprint arXiv:1703.03130*.

[63] Wang, K., X. Peng, J. Yang, D. Meng, and Y. Qiao (2020) "Region attention networks for pose and occlusion robust facial expression recognition," *IEEE Transactions on Image Processing*, **29**, pp. 4057–4069.

[64] Zeng, X., Q. Wu, S. Zhang, Z. Liu, Q. Zhou, and M. Zhang (2018) "A false trail to follow: differential effects of the facial feedback signals from the upper and lower face on the recognition of micro-expressions," *Frontiers in psychology*, **9**, p. 2015.

[65] Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio (2014) "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, pp. 2672–2680.

[66] Pumarola, A., A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer (2018) "Ganimation: Anatomically-aware facial animation from a single image," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 818–833.

[67] Kollias, D. and S. Zafeiriou (2020) "VA-StarGAN: Continuous Affect Generation," in *International Conference on Advanced Concepts for Intelligent Vision Systems*, Springer, pp. 227–238.

[68] Russell, J. A. (1980) "A circumplex model of affect." *Journal of Personality and Social Psychology*, **39**(6), p. 1161.

[69] Kossaifi, J., G. Tzimiropoulos, S. Todorovic, and M. Pantic (2017) "AFEW-VA database for valence and arousal estimation in-the-wild," *Image and Vision Computing*, **65**, pp. 23–36.

[70] Kumar, V., S. Rao, and L. Yu (2020) "Noisy Student Training using Body Language Dataset Improves Facial Expression Recognition," *arXiv preprint arXiv:2008.02655*.

[71] Oliveira, A. M., M. P. Teixeira, I. B. Fonseca, and M. Oliveira (2006) "JOINT MODEL-PARAMETER VALIDATION OF SELF-ESTIMATES OF VALENCE AND AROUSAL: PROBING A DIFFERENTIAL-WEIGHTING MODEL OF AFFECTIVE INTENSITY." *Proceedings of Fechner Day*, **22**, pp. 245–250.

[72] Lewis, P. A., H. Critchley, P. Rotshtein, and R. J. Dolan (2007) "Neural correlates of processing valence and arousal in affective words," *Cerebral cortex*, **17**(3), pp. 742–748.

[73] Nicolaou, M. A., H. Gunes, and M. Pantic (2011) "Continuous prediction of spontaneous affect from multiple cues and modalities in valence-arousal space," *IEEE Transactions on Affective Computing*, **2**(2), pp. 92–105.

[74] Huang, X., Y. Li, O. Poursaeed, J. Hopcroft, and S. Belongie (2017) "Stacked generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5077–5086.

[75] Radford, A., L. Metz, and S. Chintala (2015) "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*.

[76] Mathieu, M. F., J. J. Zhao, J. Zhao, A. Ramesh, P. Sprechmann, and Y. LeCun (2016) "Disentangling factors of variation in deep representation using adversarial training," in *Advances in Neural Information Processing Systems*, pp. 5040–5048.

[77] Karras, T., T. Aila, S. Laine, and J. Lehtinen (2017) "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*.

[78] Arjovsky, M., S. Chintala, and L. Bottou (2017) "Wasserstein GAN," *arXiv preprint arXiv:1701.07875*.

[79] Gulrajani, I., F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville (2017) "Improved training of wasserstein gans," in *Advances In Neural Information Processing Systems*, pp. 5767–5777.

[80] Mirza, M. and S. Osindero (2014) "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*.

[81] Ledig, C., L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. (2017) "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4681–4690.

[82] Isola, P., J.-Y. Zhu, T. Zhou, and A. A. Efros (2017) "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1125–1134.

[83] Tesei, G. (2019) "Generating Realistic Facial Expressions through Conditional Cycle-Consistent Generative Adversarial Networks (CCycleGAN)," .

[84] Liu, M.-Y., T. Breuel, and J. Kautz (2017) "Unsupervised image-to-image translation networks," in *Advances in Neural Information Processing Systems*, pp. 700–708.

[85] Zhu, J.-Y., T. Park, P. Isola, and A. A. Efros (2017) "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2223–2232.

[86] Kim, T., M. Cha, H. Kim, J. K. Lee, and J. Kim (2017) "Learning to discover cross-domain relations with generative adversarial networks," in *Proceedings of the International Conference on Machine Learning-Volume 70*, JMLR. org, pp. 1857–1865.

[87] Liu, M.-Y. and O. Tuzel (2016) "Coupled generative adversarial networks," in *Advances in Neural Information Processing Systems*, pp. 469–477.

[88] Kingma, D. P. and M. Welling (2013) "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*.

[89] Zhou, Y. and B. E. Shi (2017) "Photorealistic facial expression synthesis by the conditional difference adversarial autoencoder," in *Proceedings of the Seventh International Conference on Affective Computing and Intelligent Interaction (ACII)*, IEEE, pp. 370–376.

[90] Kollias, D., S. Cheng, E. Ververas, I. Kotsia, and S. Zafeiriou (2020) "Deep neural network augmentation: Generating faces for affect analysis," *International Journal of Computer Vision*, pp. 1–30.

[91] Cheng, S., I. Kotsia, M. Pantic, and S. Zafeiriou (2018) "4dfab: A large scale 4d database for facial expression analysis and biometric applications," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5117–5126.

[92] Yarowsky, D. (1995) "Unsupervised word sense disambiguation rivaling supervised methods," in *33rd annual meeting of the association for computational linguistics*, pp. 189–196.

[93] Riloff, E. (1996) "Automatically generating extraction patterns from untagged text," in *Proceedings of the national conference on artificial intelligence*, pp. 1044–1049.

[94] Radosavovic, I., P. Dollár, R. Girshick, G. Gkioxari, and K. He (2018) "Data distillation: Towards omni-supervised learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4119–4128.

[95] Bachman, P., O. Alsharif, and D. Precup (2014) "Learning with pseudo-ensembles," in *Advances in neural information processing systems*, pp. 3365–3373.

[96] Rasmus, A., M. Berglund, M. Honkala, H. Valpola, and T. Raiko (2015) "Semi-supervised learning with ladder networks," in *Advances in neural information processing systems*, pp. 3546–3554.

[97] He, K., X. Zhang, S. Ren, and J. Sun (2016) "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778.

[98] Barsoum, E., C. Zhang, C. C. Ferrer, and Z. Zhang (2016) "Training deep networks for facial expression recognition with crowd-sourced label distribution," in *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pp. 279–283.

[99] Cubuk, E. D., B. Zoph, J. Shlens, and Q. V. Le (2019) "RandAugment: Practical data augmentation with no separate search," *arXiv preprint arXiv:1909.13719*.

[100] Ulyanov, D., A. Vedaldi, and V. Lempitsky (2016) "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*.

[101] Martin Arjovsky, S. and L. Bottou (2017) "Wasserstein Generative Adversarial Networks," in *Proceedings of the International Conference on Machine Learning*.

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

[103] Parkhi, O. M., A. Vedaldi, and A. Zisserman (2015) "Deep Face Recognition," in *Proceedings of the British Machine Vision Conference*.

[104] Luo, Y., J. Ye, R. B. Adams, J. Li, M. G. Newman, and J. Z. Wang (2020) "Arbee: Towards automated recognition of bodily expression of emotion in the wild," *International Journal of Computer Vision*, **128**(1), pp. 1–25.

[105] Gu, C., C. Sun, D. A. Ross, C. Vondrick, C. Pantofaru, Y. Li, S. Vijayanarasimhan, G. Toderici, S. Ricco, R. Sukthankar, et al. (2018) "Ava: A video dataset of spatio-temporally localized atomic visual actions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6047–6056.

[106] King, D. E. (2009) "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, **10**(Jul), pp. 1755–1758.

[107] Anila, S. and N. Devarajan (2012) "Preprocessing technique for face recognition applications under varying illumination conditions," *Global Journal of Computer Science and Technology*.

[108] Liu, Y., Y. Li, X. Ma, and R. Song (2017) "Facial expression recognition with fusion features extracted from salient facial areas," *Sensors*, **17**(4), p. 712.

[109] Wang, S., W. Li, Y. Wang, Y. Jiang, S. Jiang, and R. Zhao (2012) "An Improved Difference of Gaussian Filter in Face Recognition." *Journal of Multimedia*, **7**(6), pp. 429–433.

[110] Bendjillali, R. I., M. Beladgham, K. Merit, and A. Taleb-Ahmed (2019) "Improved Facial Expression Recognition Based on DWT Feature for Deep CNN," *Electronics*, **8**(3), p. 324.

[111] Karthigayan, M., M. R. M. Juhari, R. Nagarajan, M. Sugisaka, S. Yaacob, M. R. Mamat, and H. Desa (2007) "Development of a personified face emotion recognition technique using fitness function," *Artificial Life and Robotics*, **11**(2), pp. 197–203.

[112] Ronneberger, O., P. Fischer, and T. Brox (2015) "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, Springer, pp. 234–241.

[113] Yan, J., W. Zheng, Z. Cui, C. Tang, T. Zhang, and Y. Zong (2018) "Multi-cue fusion for emotion recognition in the wild," *Neurocomputing*, **309**, pp. 27–35.

[114] Hu, P., D. Cai, S. Wang, A. Yao, and Y. Chen (2017) "Learning supervised scoring ensemble for emotion recognition in the wild," in *Proceedings of the 19th ACM international conference on multimodal interaction*, pp. 553–560.

[115] Vielzeuf, V., C. Kervadec, S. Pateux, A. Lechervy, and F. Jurie (2018) "An occam's razor view on learning audiovisual emotion recognition with small training sets," in *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pp. 589–593.