Technische Universiteit
Eindhoven
University of Technology

Department of Mathematics and Computer Science
Data Mining Research Group

Deep facial expression recognition in dynamic, resource constrained environments

Bachelor Thesis

Blazej Manczak

Supervisors:
Dr. Vlado Menkovski, Eindhoven Univeristy of Technology
Dr. Laura Astola, Accenture

Final Draft

Eindhoven, June 2020

Abstract

Fast, accurate, and automatic facial expression recognition (FER) is crucial for natural human-robot interaction. In this report, we propose a convolutional neural network (CNN) based approach for classifying facial expressions into 7 discrete emotion categories. We experiment with two models pre-trained in the facial recognition, VGG-Face, and SE-ResNet-50, and adapt them to the task of discerning emotions using a 3-stage training procedure that involves 2 biggest, in-the-wild datasets, namely AffectNet and RAF-DB. The empirical results on these datasets yield results on par with the current state-of-the-art, achieving 64.08% and 87.8% respectively. The system is then intuitively evaluated via the Grad-Cam algorithm that produces a coarse localization map of the important regions. Additionally, an extensive yet efficient deployment data preprocessing pipeline is presented. Finally, the whole system is evaluated for speed, achieving a real-time performance of 4.5 FPS on a dual-core laptop CPU and 2.5-3 FPS on a Coral Dev Board and Raspberry Pi.

Chapter 1

Introduction

Facial expressions allow one to convey the current emotional state, intentions, and frame one's actions [35]. Hence, the ability to recognize them is crucial for natural, non-verbal communication between humans. Given the importance of the field, numerous studies on facial affect have been conducted. One of the pioneering works in the field is the one by Ekman and Friesen [8] who identified 6 basic emotions that humans perceive in the same way regardless of culture. The proposed facial expressions are anger, disgust, fear, happiness, sadness, and surprise. Interestingly, recent research in the fields of neuroscience and psychology has challenged the hypothesis of culture-universal expressions [16]. Moreover, the expressive power of the model has been questioned [23]. More intricate models of measuring emotions have been proposed [23] such as the Facial Action Coding System (FACS) or continuous affect dimension model. However, no model is perfect. The new, proposed measures to capture emotions, vary widely, are not straight-froward in interpretation, and suffer from low agreeableness between the annotators. That is why mainly due to its simplicity, pioneering roots and intuitive interpretation, the simple discrete model with 6 basic emotions extended by a neutral expression dominates the facial expression recognition (FER) research today. In this paper, the categorical model is used.

Given the amount of information that can be derived from human facial expressions, the use-cases for automatic FER systems abound in a wide variety of fields. For instance, the facial expression recognition system is considered an important feedback mechanism in e-learning platforms to monitor's students' understanding and engagement [9]. Similarly, it can be a commercial tool for companies to gain more insight into how the ads affect peoples' emotions. However, the most noticeable use-case of an automatic FER system is human-robot interaction. As the field of social robotics advances, the need for accurate, fast, and automatic solutions for reading and interpreting human emotional cues is profusely important [28]. By making the interaction more passive and pervasive one can achieve a higher level of engagement as the robot can interact more naturally and accurately with the user.

Even though interpreting facial expressions for humans is natural and trivial, this task is not that easy for a machine. Due to the importance of emotion recognition, many approaches throughout the years have been proposed to solve it. The traditional approaches such as non-negative matrix-factorization (NMF) or local binary patterns (LBP) dominated the literature until 2013 [23]. These solutions were susceptible to small changes in expression yielding opposing outcomes. Moreover, not all features derived through these methods were effective and positive, which resulted in pure generalization to previously unseen examples [29] [15]. The undertaken approaches have shifted towards deep learning in 2013 when two in-the-wild challenges were conducted: EmotiW and FER2013. There were two main axes of change. Firstly, the datasets contained facial expressions in the wild contrary to laboratory-controlled poses used before. Secondly, the amount of samples has increased drastically: from a few hundreds to tens of thousands. Combined with a rapid increase in chip processing power such as GPU units and the inability to create universal handcrafted features, the field turned to convolution neural networks (CNN's) which had already enabled a breakthrough in image classification. This transition spurred many novel approaches

that significantly improved upon the state of the art at the time.

Although many excellent solutions have been proposed in the FER literature, they are almost exclusively not deployment-oriented. The available off-the-shelf solutions are mostly trained on posed expressions datasets that do not fare well in in-the-wild settings. Secondly, many state-of-the-art approaches use an ensemble of multiple deep networks or make the final prediction based on multiple transformations of the queried image. Both of these aspects significantly slow down the inference procedure which makes it impractical in dynamic deployment environments. The problem of deployment preprocessing pipeline is also often neglected. It is needed to parse the raw input to match the training data that is usually preprocessed already at the data collection stage. Making sure that the distributions of the training and real-world samples match is crucial for effective deployment. The few real-time solutions that one can find in the literature have two main shortcomings. Firstly, they are often trained on lab-posed datasets. Secondly, the real-time performance metrics such as frame rate are reported on big, stationary, and expensive hardware that is neither suitable nor scalable to dynamic scenarios. Lastly, many works limit the model evaluation to analyzing metrics on the test set. However, for validation purposes, extra measures that provide insight into why the classification was made should be undertaken to make sure the models do not utilize a specific dataset bias.

The objective of this paper is to build an algorithm that in-real time is capable of detecting a face and classifying the corresponding expression into one of the discrete emotions in the context of human-drone interaction. The scope of this work is restricted to CNN's as a solution for feature learning. In summary, my contribution consists of (i) creating a model capable of real-time, in-the-wild facial emotion detection under resource constraint that (ii) achieves close to the state of the art performance on benchmark datasets, (iii) creating an efficient but extensive deployment data preprocessing pipeline and (iv) providing an interpretable validation of the model by visualizing the network activations. The structure of the paper is the following: we firstly review the related work in Chapter II and list the challenges in building such a solution. Subsequently, in Chapter III, I propose, combine, and reason about the choices made for each of the challenges. I evaluate the system in Chapter IV. Finally, I provide a conclusion in Chapter V.

Chapter 2

Related work

In the literature about deep learning-based approaches concerning FER, one can find reoccurring challenges that need to be addressed for the method to perform well. This section lists these challenges and reviews a subset of possible solutions. In particular, I focus on the following issues:

- datasets suitable for the given classification problem
- agreeableness between the annotators
- preprocessing techniques
- class imbalance
- feature learning

Please note that each of the listed challenges is often a vibrant research area on its own and it would be impractical to try to list all approaches here. There exist comprehensive articles such as [23] [35] that attempt to do that.

2.1 Dataset

Supervised deep learning approaches investigated in this paper require a lot of labeled examples to perform well. Moreover, the application domain treated in this paper regards the in-the-wild setting which increases the need for a big dataset containing many environments and poses. In this section, I provide an overview of the most notable, big-scale in-the-wild image datasets. Some of the dataset descriptions are excerpts from Li and Deng [23].

FER2013 [11]: The FER2013 database was the first of its kind, introduced during the ICML 2013 Challenges in Representation Learning. FER2013 is a large-scale and unconstrained database collected automatically by the Google image search API. All images have been registered and resized to 48*48 pixels after rejecting wrongfully labeled frames and adjusting the cropped region. FER2013 contains 28,709 training images, 3,589 validation images, and 3,589 test images with seven expression labels (anger, disgust, fear, happiness, sadness, surprise, and neutral).

EmotioNet[3] : EmotioNet is a large-scale database with one million facial expression images collected from the Internet. A total of 950,000 images were annotated by the automatic action unit (AU) detection model, and the remaining 25,000 images were manually annotated with 11 AUs.

RAF-DB [24]: The Real-world Affective Face Database (RAF-DB) is a real-world database that contains 29,672 highly diverse facial images downloaded from the Internet. With manually crowd-sourced annotation (40 annotators per image) each image being annotated by 40 annotators)and reliable estimation, seven basic and eleven compound emotion labels are provided for the samples. Specifically, 15,339 images from the basic emotion set are divided into two groups (12,271 training samples and 3,068 testing samples) for evaluation.

AffectNet [30]: AffectNet contains more than one million images from the Internet that were obtained by querying different search engines using emotion-related tags. It is by far the largest database that provides facial expressions in two different emotion models (categorical model and dimensional model), of which 450,000 images have manually annotated labels for ten different classes basic expressions.

2.2 Agreeableness between the annotators

Annotation bias is inevitable in facial expression tasks due to the subjective nature of the annotation task. Looking at Figure 2.1 one sees that similar expressions have different labels for AffectNet and RAF datasets. This in turn affects the inference stage as CNN learns the labels. A valid question to ask at this point is what the ground truth is.

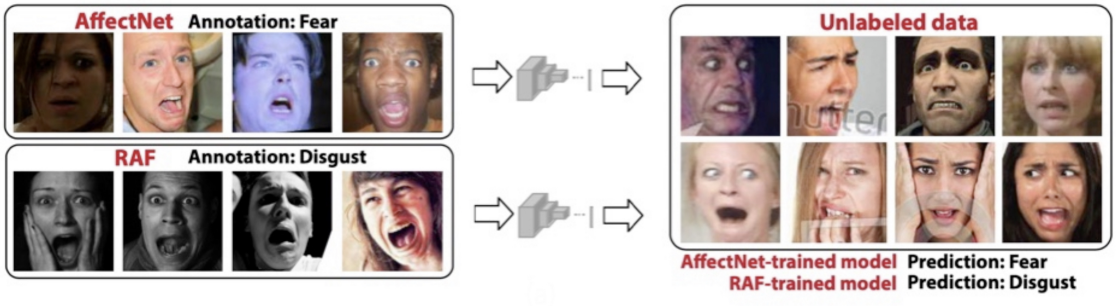

Figure 2.1: Noise/Inconsistency annotations between RAF and AffectNet datasets. [47]

Moreover, if two datasets differ in classification, does the difference stem from inconsistency in labeling procedure, or does one dataset have more noise than the other? In other words, were the semantic instructions given to human annotators different across datasets (inconsistency), or are there more/more careful annotators across datasets (noise)?. When the general agreeableness in labeling decreases, so does the performance gain from the models trained on the conjunction of datasets as gradient descent based algorithms cannot converge [17]. It is often impossible to establish whether the noise or the inconsistency is the root of the problem. Often it is the combination of the two. However, the proposed remedies usually make explicit assumptions about source disagreement.

The issue of noise is much more prevalent in the literature as it also concerns the cases where there is no dispute about what the ground truth is. This approach most often assumes that one posses a small set of clean data and a larger set of noisy data. The straightforward approach is to first train the model on the noisy subset and then fine-tune on the clean data. The clean data can also be used to assess the quality of the labels in course of training [41] [26]. For example Veit et al. [41] trained two CNNs on clean and noisy subsets in parallel. The networks shared the feature extraction layers. One CNN used the clean dataset to learn to clean the noisy dataset, which was then used by the other CNN to learn the main classification task. Other techniques use clean data to train feature extractors to assess the quality of the noisy label and re-assign or weight the sample [22] [7]. Yet another approach is to estimate the distribution of the noisy labels and adapt output probabilities to better match the noisy labels [39].

There is significantly less research about inconstant annotations between datasets as such a problem is less prevalent. Moreover, most of them assume to have access to multiple annotations per image. If that is the case, the most straightforward way of addressing the issue is to use soft labels [14] during the training process, i.e. not assign a specific class to a sample but rather provide a probability distribution over all classes. This approach ignores the fact that some annotations are less reliable than others. To address that, one might use an EM-based algorithm to filter out unreliable annotations, as proposed by [6]. This technique has empirical success in estimating

labels from crowd-sourcing [47]. It has been used in compiling the labels of the RAF-DB dataset. If one does not have multiple annotations per image, solutions such as [47] propose to use multiple datasets, say A and B, and get those extra annotations by training the models on dataset A and making predictions on dataset B.

2.3 Preprocessing techniques

Variation invariant to facial expressions such as illumination, background or head pose are abundant in in-the-wild scenarios. These features constitute noise which is undesired for convergence of neural networks. To address these problems, the preprocessing pipeline usually consists of face detection, face alignment and illumination correction [23]. The result of such pipeline is portrayed in Figure 2.2.

Figure 2.2: Example of a preprocessing pipeline that includes detection of a face and facial landmarks, illumination normalization and face alignment

The first step in preprocessing the image is to register the position of the face. There are many algorithms to do that. Perhaps the most notable one is the Viola-Jones (VJ) face detector [42]. Moreover, it has been repeatedly shown that further face alignment can yield significant improvements by reducing variation in face scale and in-plane rotation [46] [18]. The face alignment detectors usually also encapsulate face detection and capture the location of landmarks such as eyes, nose, mouth corners, and often more. One of the most popular methods for obtaining the landmarks are discriminative models that use a cascade of regression functions to map the image appearance to landmark locations. [45]. There are also deep learning-based approaches such as Tasks-Constrained Deep Convolutional Network (TCDCN) [49] and Multi-task CNN (MTCNN) [48] that further leverage multi-task learning by utilizing the learning capacity of neural networks. Based on the facial landmarks the pose is normalized by affine or rigid transformations. It is noteworthy that due to partial face occlusions and peculiar poses it is not always possible to reliably estimate these landmarks [33]. One way to alleviate the problem is to only used landmarks given high confidence of the classifier.

Many existing methods also perform some sort of illumination normalization to decrease the intra-class variance. in the image. The most popular way to perform illumination normalization is via histogram equalization [23]. It manages to increase the global contrast well given that the brightness of the background and foreground are similar. Otherwise, parts of the images might be too bright which results in losing important features. A solution to this problem is contrast limited adaptive histogram equalization (CLAHE) [34] that first divides the image into tiles withing which it applies histogram normalization, clipping the amplification factor.

2.4 Class imbalance

The class imbalance is prevalent in in-the-wild facial expression datasets. This stems from the fact that the samples are most often queried from the internet where people tend to share certain emotions such as happiness over others, such as disgust. This imbalance poses a problem for neural networks. If no countermeasures are taken, the model will overemphasize the dominant classes. That is because, on average, the loss incurred from misclassifying an image as a class with big support is lower than the misclassification of the class with small support.

There are plenty of methods to deal with the issue. [30] investigates the performance on the balanced test set of AffectNet baseline CNN using 4 approaches: down-sampling, up-sampling, weighted loss, and regular imbalanced dataset. Weighted loss function with class weights proportional to $\frac{f_i}{f_{min}}$, where f_i is the number of samples of the i^{th} class and f_{min} is the number of samples in the most underrepresented class. Moreover, the issue is often remedied with extensive data augmentation [15] [23] [33]. The augmentation usually entails random perturbations such as rotations, shifting, skewing, scaling, noise contrast, and color jittering. Some approaches go a step further and generate additional artificial data using general adversarial models (GANs) [1].

2.5 CNN architecture, training and inference

CNN's dominated the FER research area mainly due to being invariant to face location changes and scale variations, which is a big advantage over the traditional methods. Since the debut of CNN's in FER in 2013, many architectures have been utilized. The review of the literature reveals a trend of using deeper and deeper models over time, mainly due to leaps in deep learning field and an increase in computational resources. An overview of used architectures and their characteristics are shown in Table 2.1. The

	AlexNet [21]	VGG [37]	GoogleNet [40]	ResNet [12]
#layers	$5+3$	$13/16+3$	$21+1$	(17 to 151) + 1
Kernel size	$11, 5, 3$	3	$7, 1, 3, 5$	$7, 1, 3, 5$
Inception module	✗	✗	✓	✗
Batch norm	✗	✗	✗	✓
Used in	[31]	[10] [44]	[50] [5]	[15] [47]

Table 2.1: Comparison of CNN architectures.

Many state of the art approaches utilize several architectures by encompassing them in an ensemble [15] [33] [38]. This strategy is effective due to different architectures learning distinctive features of different emotions. For example, one architecture might perform significantly better at recognizing surprise whereas another at recognizing fear. The predictions are then integrated via different forms of averaging.

Another approach to boost the performance is to transfer the initial parameters of the network from a closely related problem [10] [20], which is usually a facial recognition task. Moreover, especially the works before the AffectNet dataset used to perform training on the conjunction of many datasets [33] to compensate for the relatively small size of FER datasets. This trend is not prominent in recent works, as the number of samples has dramatically increased in the datasets such as AffectNet or RAF-DB.

As mentioned in Section 2, one can hardly find the approaches that do not employ data augmentation in course of training. Some works go a step further and use augmentation for inference. For instance, [15] predicts eight images that are generated by translations and flips of the original image.

Chapter 3

Methods

The proposed method consists of steps that must be taken in sequence to maximize performance. In this section, I address these steps and elaborate on their implementation details. The employed solutions to problems raised in Related work are listed here.

3.1 Method overview

Two in-the-wild facial expressions datasets are used: AffectNet and RAF-DB. AffectNet dataset is used as a set of noisy labels, whereas RAF-DB is treated as a set of clean labels. By pre-training CNN on the set of noisy labels first and then fine-tuning on a set with clean labels both the scale of AffectNet and the quality of RAF-DB are utilized. A loss function weighted by the inverse of class support is used to address the problem of class imbalance that is present in both datasets. Additionally, data augmentation is used to artificially increase the size and diversity of the datasets. Before training, all images are scaled down to the size of 224×224 and are normalized to have zero mean and unit variance. Then VGG-Face and SE-ResNet-50 models are fine-tuned to the task of discerning emotions from facial expressions. For model deployment, the quantization procedure is carried out to accommodate faster inference for resource constrained environments. The deployment pipeline consisting of face detection, alignment, and normalization is created to parse the raw camera inputs.

3.2 Data handling

The experiments in this paper are conducted on the AffectNet [30] and RAF-DB [24] datasets. Both datasets contain in-the-wild facial expressions to match the deployment target environment. The images in these datasets are already cropped to include the face only. No additional offline preprocessing is performed on either dataset. Additionally, in both datasets, one can distinguish the same subset of 7 classes: angry, disgust, fear, happy, sad, surprise, and neutral.
AffectNet is currently the biggest dataset annotated by humans with 265 thousand images of the aforementioned labels. Images have been annotated by one annotator. A subset of 36 thousand images has been annotated by two annotators to check the consistency of annotations. The consistency for 11 classes is 60.7%, which assuming uniform redistribution amounts to approximately 69% consistency for the 7 class problem.
RAF-DB dataset is a much smaller dataset, containing 15 thousand images of the aforementioned classes. The strength of this datasets lies in each image being annotated by 40 annotators. Additionally, the Expectation-Maximization (EM) framework was used to assess each labeler's reliability and removal of inconsistent annotations. The usage of the framework resulted in a high consistency score with Cronbach's Alpha score of 0.966.

Both datasets suffer from a class imbalance in the training sets as can be seen in figure 3.1. In both datasets the class with the highest support is happy. In the case of AffectNet, the ratio of

the class with the biggest support to the one with the lowest support (disgust) is 0.03. For RAF-DB this ratio is 0.06. Two techniques are applied to combat the issue: firstly, the loss function is altered to penalize the miss-classification of under-populated classes. The loss function is re-weighted by the inverse class frequency. Secondly, the online data augmentation using random image perturbations such as flipping, rotating, and scaling is applied. A sample of transformation can be seen in Figure 3.2.

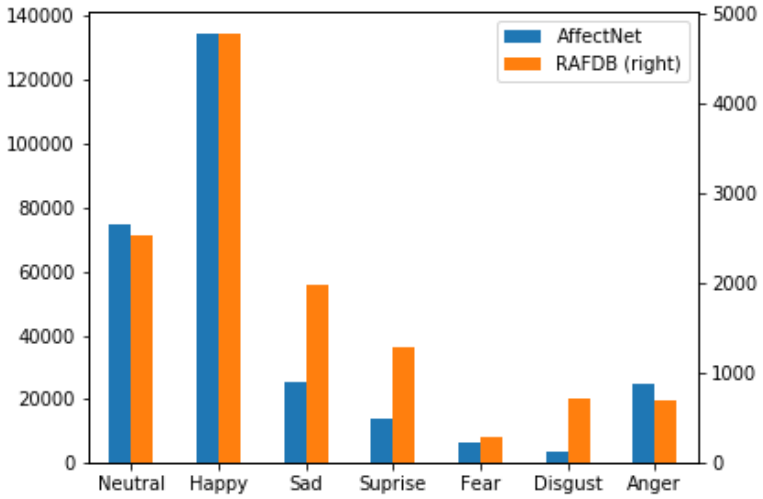

Figure 3.1: Number of samples in AffectNet (blue) and RAF-DB dataset(orange). Please note the separate y axes.

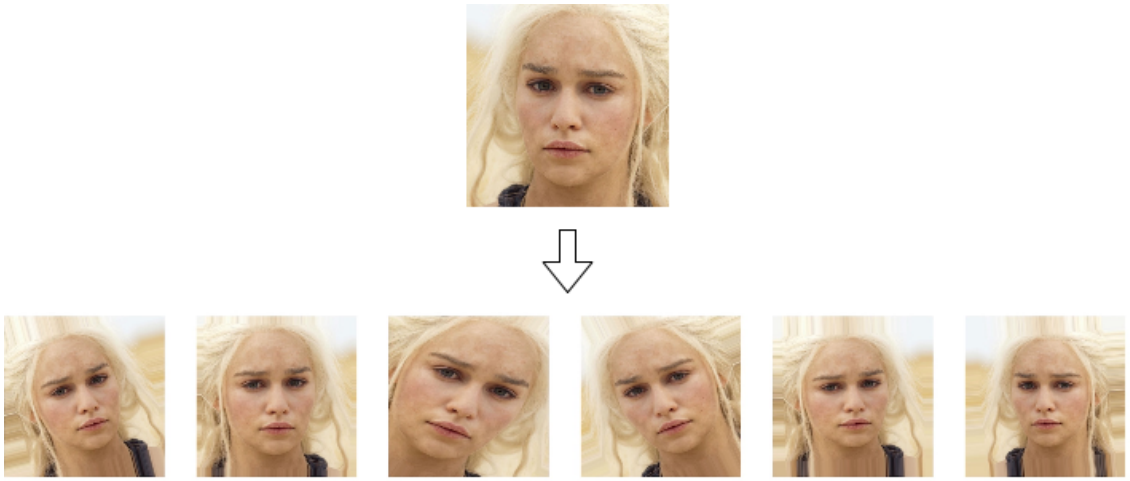

Figure 3.2: Example of applied data augmentation

As mentioned in Section 2.2 Agreeableness between the annotators, there is a significant issue of consistency in annotations. To mitigate this problem and use all of the available data at the same time a decision has been made to first pre-train a network on the AffectNet dataset and then fine-tune it on the Raf-DB dataset. This procedure assumes that the differences in annotation stem from noise rather than inconsistent annotation procedure.

3.3 CNN architecture

The procedure of choosing the architecture was dictated by two metrics: deployability and performance. The first aspect precludes solutions like ensemble learning, meaning that one cannot utilize multiple architectures. Moreover, the loss accrued due to the quantization loss of similar models was investigated. The best performing stand-alone architectures for face related tasks were found via the literature review. That being said, two models are investigated: VGG-face [32] and SE-ResNet-50 [4] (SENet for short). Neither architecture is trained from scratch. The pre-trained weights for a closely related task of face recognition are used. Both architectures have a record of successfully being quantized on other tasks without much performance loss and they allow for a relatively fast inference on a resource-constrained devices without sacrificing the performance.

3.3.1 VGG-Face

VGG16 [37] is an architecture with 16 layers: 13 convolution layers and 3 fully connected layers. It is characterized by using a small receptive field with a size of 3×3. The input image is 224×224 which is consecutively passed through blocks consisting of convolutional (conv) and max-pooling layers. Finally, there are 3 fully connected (fc) layers attached on top of the network; first two with 4096 units and the third outputs classification into originally one of 1000 categories using softmax activation. Other layers use ReLU as an activation function. The network totals 138 million parameters. The architecture is portrayed in figure 3.3.

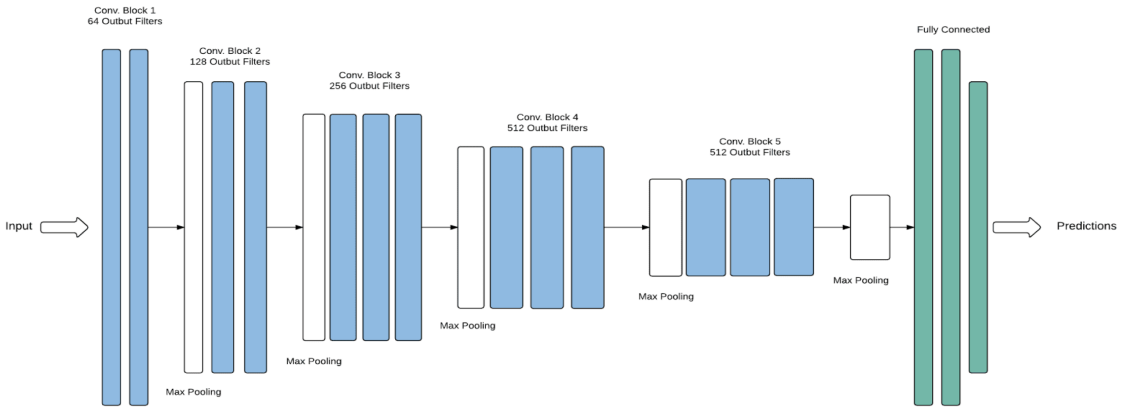

Figure 3.3: Architecture of VGG16 [27]

The VGGFace network [32] uses VGG16 architecture, with the output layer having 2622 units. VGGFace network is trained for face recognition task on the VGGFace that consists of 2.6 million face images with 2622 unique subjects. Although it is not a very deep architecture it has achieved the state of the art results in its original task as well as in facial affect recognition after fine-tuning. [10].

3.3.2 SE-ResNet-50

ResNet [12] is a deep architecture with residual connections that alleviate the risk of vanishing gradient problem by introducing skip connections. The layers within which the skip connection originates and ends is called the residual block, see Figure 3.4. The skip connection simply adds the input x to the head of the residual block. ResNet architectures differ in size based on how big

each block is and how many blocks there are. Here ResNet-50 is used, with the architecture as shown in Table 3.1.

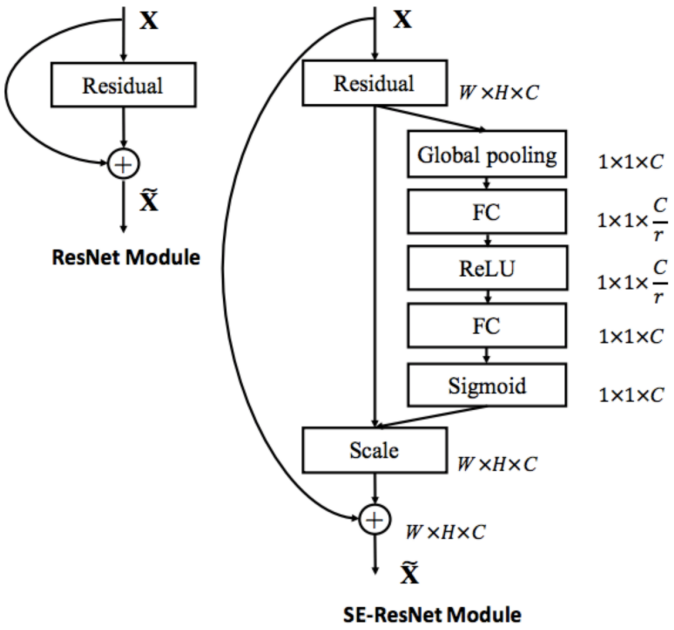

Figure 3.4: The residual block [left] and a residual block with Squeeze-and-Excitation component [right]. [12]

layer name	ResNet50
conv1	$7 \times 7, 64, stride 2$
conv2_x	3×3 max pool , stride 2 $\begin{bmatrix} 1 \times 1, & 64 \\ 3 \times 3, & 64 \\ 1 \times 1, & 256 \end{bmatrix} \times 3$
conv3_x	$\begin{bmatrix} 1 \times 1, & 128 \\ 3 \times 3, & 128 \\ 1 \times 1, & 512 \end{bmatrix} \times 4$
conv4_x	$\begin{bmatrix} 1 \times 1, & 256 \\ 3 \times 3, & 256 \\ 1 \times 1, & 1024 \end{bmatrix} \times 6$
conv5_x	$\begin{bmatrix} 1 \times 1, & 512 \\ 3 \times 3, & 512 \\ 1 \times 1, & 2048 \end{bmatrix} \times 3$
	average pool, 1000-d fc, softmax

Table 3.1: ResNet50 Architecture. n×n represents the conv filter size, after which is the number of filters. The number after array structure means how many times this structure will be replicated [27]

Additionally, the Squeeze-and-Excitation (SE) [13] blocks are added to the architecture. SE blocks can be integrated with modern architectures to improve their representational power. They improve channel interdependencies at a very little computational cost. It is achieved by explicitly modeling channel relationships with a small neural networks.

3.4 Training procedure

The training procedure for both networks follows the same structure and contains 3 phases. It is portrayed in Figure 3.5. The top part of the SENet architecture has been modified to have one dense layer with 1000 units, followed by a dropout rate and a softmax classification layer corresponding to the 7 facial expression. The softmax layer in VGG-Face has been modified in the same manner. Moreover, for regularization purposes, dropout layer has been added after the first two fc layers with the dropout rate of 0.5.

While training the model on the AffecNet dataset, two phases of fine-tuning are carried out. Firstly the weights in conv layers or residual blocks are frozen and only the top part with fc layers are trained to adapt the network for the task of facial expression recognition. In this first phase of fine-tuning, Adam optimizer is used with a learning rate equal to 10^{-3} and the default hyperparameters as prescribed in the paper by Kingma et al. [19]. In the second round of fine-tuning the model obtained in the first round is used. All the layers are then unfrozen but this time a significantly smaller learning rate is used, namely 10^{-5}. After the second phase model has converged, it is used in the final step of training on the RAF-DB dataset. The training procedure on RAF-DB is the same as the second phase on AffectNet: all layers have trainable parameters and training takes place with a learning rate of 10^{-5}.

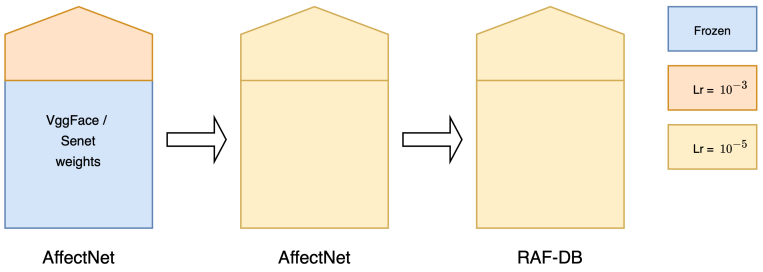

Figure 3.5: The training procedure. The arrow implies the transfer of parameters from model on the left to model on the right. The colors illustrate used learning rate.

3.5 Deployment specific methods

Before deploying the algorithm on a resource constraint device, the model is quantized to INT8 format. A quantized model executes some or all of the operations on tensors with integers rather than floating-point values. This allows for more compact model representation and the use of high performance vectorized operations on many hardware platforms. Once the quantized algorithm is deployed on the device, one needs to parse the input like a camera feed to be compatible with the model. It is of utmost importance that the input to the algorithm is as similar as possible to the training data. Even a slight difference in data distribution can significantly degrade performance. Moreover, the parsing process must enable inference of multiple faces. To ensure that there are as few discrepancies as possible the following steps are undertaken: face detection, face extraction, face alignment, and (in low resolution) illumination normalization. This process is shown in Figure 2.2.

The first, optional step in the deployment pipeline is illumination normalization. If the lighting conditions are undesirable, contrast limited adaptive histogram normalization (CLAHE) can be applied. However, as this method is expensive, the image resolution must be significantly lowered. The next step is face detection. The MTCNN [48] algorithm is used due to is excellent reliability and relatively fast detection. The output of this classifier contains the bounding box together with facial landmarks and prediction confidence for each detected face. The detections below 0.9 are discarded. Subsequently, for each detected the position of the eyes is utilized to derive the

rotation angle needed to ensure that in the transformed output the eyes lie on a horizontal line. The transformation is applied and the faces are extracted. Each face blob is the resized to 224×224 and feature-wise centering and standard deviation normalization are applied with parameters from the original training models [32] [4].

Chapter 4

Experiments and performance analysis

The evaluation of the system is performed at multiple levels. Firstly, the results of the models in the course of consecutive stages of the training process are evaluated on respective validation sets before and after the quantization procedure. The Gradient-weighted Class Activation Mapping (Grad-Cam) [36] visualization technique is used to affirm that the network's neurons do not activate on unrelated features. Lastly, an overview of on-drone deployment results is provided.

4.1 Quantitative results

The accuracy of the system is analyzed at each stage of the training process as described in Figure 3.5. Two architectures are compared, namely VGG-Face and Senet-50. Their performance is portrayed in Table 4.1. Additionally, the accuracy of several of the state-of-the-art approaches is shown in Table 4.2.

	VGG-Face	SENet-50
1st round (AffectNet)	56.20%	-
2nd round (AffectNet)	63.80%	64.06%
3rd round (RAF-DB)	86.01%	87.80%

Table 4.1: Accuracy from second (AffectNet) and third (RAF-DB) stage of training, before quantization.

	Accuracy		
AffectNet (7-class)	55.33% [25]	62.11% [15]	63.31% [10]
RAF-DB	86.9% [44]	87.00% [2]	88.14% [43]

Table 4.2: State of the art results on the corresponding validation sets.

Noticeably, the SENet-50 model could not be trained in the first phase. The attempt of unfreezing the top layers resulted in the validation accuracy not changing. Only stage 2 was conducted with top layers' parameters being randomly initialized from the standard normal distribution. We see that both VGG-Face and SENet-50 perform similarly after on the AffectNet dataset. However, in the target, the RAF-DB dataset SENet-50 is better by almost 2%. The performance on both datasets is close to the state of art results and even better in the case of the

AffectNet dataset. Subsequently, the confusion matrices are used to gain more insight into the strengths and shortcomings of the models and to establish where they differ, see Figure 4.1.

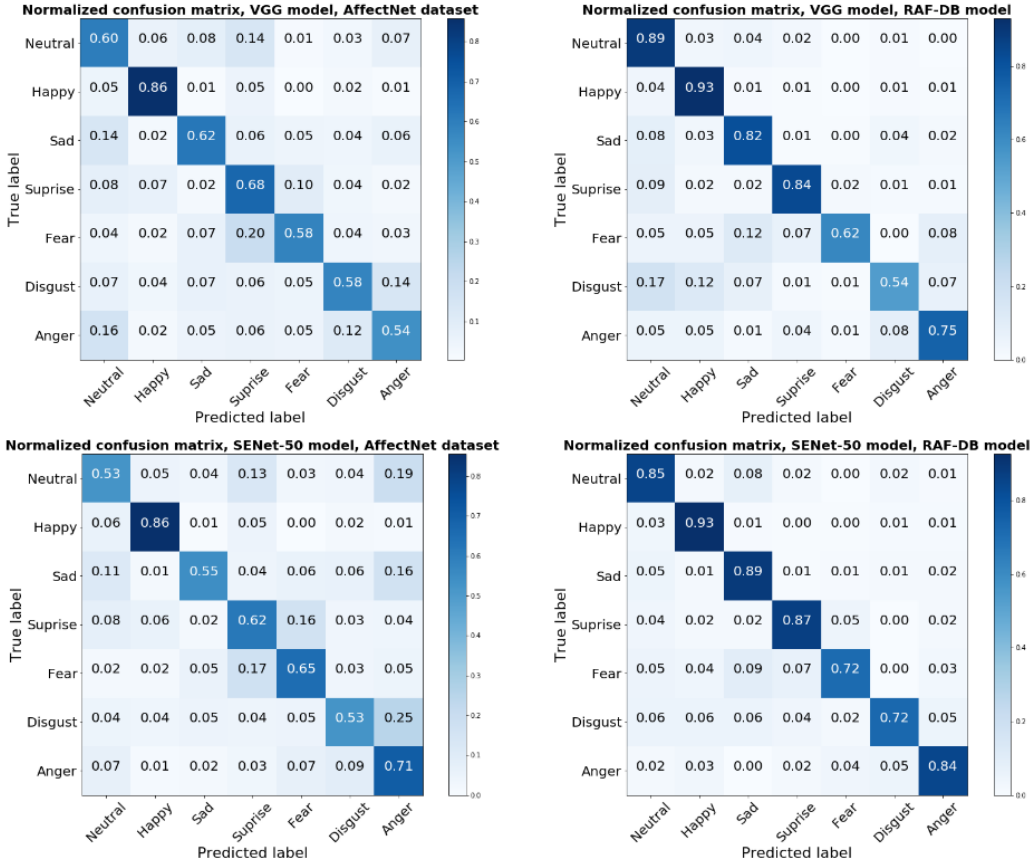

Figure 4.1: Confusion matrices for VGG-Face (first row) and SENet-50 (second row) models from second (AffectNet) and third stage(RAF-DB) of training

Universally, models performed well in the happy class (86% - 93% accuracy). Surprisingly, the accuracy in other classes is highly dependent on the used model. That is especially the case on the AffectNet dataset, where the classes ordered by performance are almost reversed. For instance, anger is the class with the second-best accuracy (71%) using the SENet-50 model but the class with worst accuracy (54%) using the VGG model on the AffectNet dataset. The difference most likely stems from significantly different architectures. The applied filters might be more prone to capture relevant features of certain emotions. On the RAF-DB dataset, which is a dataset with higher consistency, this phenomenon is less severe. For both architectures, fear and disgust are the classes with the lowest accuracy. However, the SENet model has less volatile error across the classes with the standard deviation of diagonal of 0.075 as compared to 0.133 for the VGG model. Moreover, as the test set of RAF-DB is imbalanced, an average prediction accuracy per class is useful to consider. The upper hand of SENet is even more prominent, with the average prediction accuracy of 83.1% compared to 77% for the VGG model.

After the 3rd stage of training, the model is quantized to make it ready for deployment. The results of quantization are portrayed in Table 4.3.

Both models undergo quantization well, not losing much accuracy. The FP32 to INT8 quantization allows for a 4x reduction in the model size and a 4x reduction in memory bandwidth requirements.

	VGG-Face	SENet-50
Before	86.01%	87.80%
After	85.88%	87.64 %

Table 4.3: Quantization results of the 3rd stage models.

4.2 Qualitative results

Figure 4.2 shows a misclassification made by the Senet-50 model. The problem of complexity and ambiguity of facial expressions comes to prominence. The mistakes of the model do not strike as obvious and plain wrong but rather show how intricate human expressions are and that the discrete set of emotions cannot capture the compound emotional load of some expressions.

Figure 4.2: Example misclassifications by the Senet-50 model.

Furthermore, Grad-Cam algorithm is employed as an intuitive check whether the facial features that matter to the network are indeed related to the predicted expressions. Grad-CAM uses the gradients flowing into the final conv layer to produce a coarse localization map highlighting the important regions in the image for predicting the concept. The results for different expressions can be seen in Figure 4.3. One sees that the neurons of the last convolutional layer activate most in the area around the eyes, which is not an unreasonable choice.

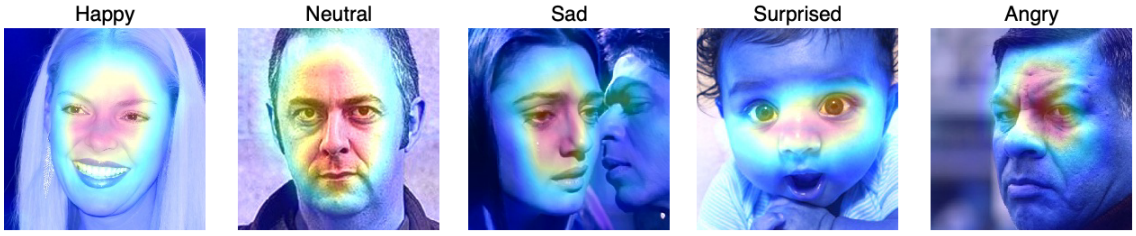

Figure 4.3: Example outputs of Grad-Cam algorithm.

4.3 On-device testing

The whole system featuring the quantized Senet-50 model is deployed on the BlueJay drone. The drone is equipped with two computation units, namely Google Dev Board and Rasberry PI model b+. The former one is used for inference of the emotion classification model as it has TPU support enabling faster inference. The quantization of the MTCNN model was unsuccessful so it runs on the CPU together with the rest of the preprocessing pipeline. The rest of the operations is performed on the Pi's CPU. Prior to applying the preprocessing deployment pipeline, the input from the camera is resized to a size 600×600 pixels. As the drone had technical problems before submission date, no footage from the drone can be presented in this report.

The whole system has been tested for speed. Firstly, the system has been executed on a 2016, baseline 13" inch MacBookPro with Skylake 2.9 GHz Intel Core i5" processor (6267U). The frame rate has been consistent around 4.5 FPS. A sample of outputs from the computer testing is portrayed in Figure 4.4. With the computational units that will be mounted on the drone the method achieves 2.5 - 3 FPS depending on the auxiliary computations performed on the drone. Taking into account that none of the computations are outsourced, this is satisfactory performance.

Figure 4.4: A sample of outputs from testing the whole system.

Chapter 5

Conclusion and discussion

This paper proposes a fast and accurate deep learning-based approach for automatic, in-the-wild facial expression recognition that can be deployed in resource constraint environments such as drones. Two state-of-the-art models used in the facial recognition were adopted for tasks of discerning emotions using a training procedure that involves two biggest, in-the-wild datasets, namely AffectNet and RAF. The post-quantization results are comparable to the state of the art scores in the literature, showing that to achieve great performance one does not need an ensemble of many networks or intricate inference pipelines. Finally, I provide additional, intuitive validation of the approach by using the Grad-Cam algorithm followed by example predictions of the entire system.

However, there is room for improvement. Firstly, one could use the additional 450 thousand automatically labeled images offered by the AffectNet dataset utilizing noisy labels or semi-supervised based approaches. Moreover, more experimentation could be done with the problem of data imbalance, such as employing a different loss function that takes into account that not every new sample of a certain class is equally important. It is also noteworthy that the results obtained on this and other FER datasets are only indicative of real-world FER performance due to dataset bias. This limitation applies not only to this study but to FER research in general. To fix this problem, a new approach to labeling data where the ground truth is unclear should be undertaken. In this paper, I used one of the simplest approaches to address dataset bias. Even though it worked well, there is room for a lot of improvement. For the better deployment results, one could use a quantized model for face detection that could run on the GPU/TPU. Additionally, one could increase performance by using a small ensemble of networks using parallel computing if supported by hardware.

Bibliography

[1] Iman Abbasnejad, Sridha Sridharan, Dung Nguyen, Simon Denman, Clinton Fookes, and Simon Lucey. Using Synthetic Data to Improve Facial Expression Analysis with 3D Convolutional Networks. In *Proceedings - 2017 IEEE International Conference on Computer Vision Workshops, ICCVW 2017*, 2017. 6

[2] Dinesh Acharya, Zhiwu Huang, Danda Pani Paudel, and Luc Van Gool. Covariance pooling for facial expression recognition. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2018. 13

[3] C. Fabian Benitez-Quiroz, Ramprakash Srinivasan, and Aleix M. Martinez. EmotioNet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016. 3

[4] Qiong Cao, Li Shen, Weidi Xie, Omkar M. Parkhi, and Andrew Zisserman. VGGFace2: A dataset for recognising faces across pose and age. In *Proceedings - 13th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2018*, 2018. 9, 12

[5] Ciprian Adrian Corneanu, Marc Oliu Simón, Jeffrey F. Cohn, and Sergio Escalera Guerrero. Survey on RGB, 3D, Thermal, and Multimodal Approaches for Facial Expression Recognition: History, Trends, and Affect-Related Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016. 6

[6] A. P. Dawid and A. M. Skene. Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. *Applied Statistics*, 1979. 4

[7] Mostafa Dehghani, Aliaksei Severyn, Sascha Rothe, and Jaap Kamps. Avoiding your teacher's mistakes: Training neural networks with controlled weak supervision. *arXiv preprint arXiv:1711.00313*, 2017. 4

[8] Paul Ekman and Wallace V. Friesen. Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology*, 1971. 1

[9] Zixiang Fei, Erfu Yang, David Day Uei Li, Stephen Butler, Winifred Ijomah, Xia Li, and Huiyu Zhou. Deep convolution network based emotion analysis towards mental health care. *Neurocomputing*, 2020. 1

[10] Mariana-Iuliana Georgescu, Radu Tudor Ionescu, and Marius Popescu. Local learning with deep and handcrafted features for facial expression recognition. *IEEE Access*, 7:64827–64836, 2019. 6, 9, 13

[11] Ian J. Goodfellow, Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski, Yichuan Tang, David Thaler, Dong Hyun Lee, Yingbo Zhou, Chetan Ramaiah, Fangxiang Feng, Ruifan Li, Xiaojie Wang, Dimitris Athanasakis, John Shawe-Taylor, Maxim Milakov, John Park, Radu Ionescu, Marius Popescu, Cristian Grozea, James Bergstra, Jingjing Xie, Lukasz Romaszko, Bing Xu, Zhang Chuang, and Yoshua Bengio.

Challenges in representation learning: A report on three machine learning contests. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2013. 3

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. ResNet. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016. 6, 9, 10

[13] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-Excitation Networks. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018. 10

[14] Ninghang Hu, Gwenn Englebienne, Zhongyu Lou, and Ben Krose. Learning to Recognize Human Activities Using Soft Labels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 4

[15] Wentao Hua, Fei Dai, Liya Huang, Jian Xiong, and Guan Gui. HERO: Human Emotions Recognition for Realizing Intelligent Internet of Things. *IEEE Access*, 2019. 1, 6, 13

[16] Rachael E. Jack, Oliver G.B. Garrod, Hui Yu, Roberto Caldara, and Philippe G. Schyns. Facial expressions of emotion are not culturally universal. *Proceedings of the National Academy of Sciences of the United States of America*, 2012. 1

[17] Davood Karimi, Haoran Dou, Simon K Warfield, and Ali Gholipour. Deep learning with noisy labels: exploring techniques and remedies in medical image analysis. *arXiv preprint arXiv:1912.02911*, 2019. 4

[18] Dae Hoe Kim, Wissam J. Baddar, Jinhyeok Jang, and Yong Man Ro. Multi-objective based spatio-temporal feature representation learning robust to expression intensity variations for facial expression recognition. *IEEE Transactions on Affective Computing*, 2019. 5

[19] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015. 11

[20] Dimitrios Kollias, Shiyang Cheng, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Deep Neural Network Augmentation: Generating Faces for Affect Analysis. *International Journal of Computer Vision*, 2020. 6

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012 AlexNet. *Advances In Neural Information Processing Systems*, 2012. 6

[22] Kuang Huei Lee, Xiaodong He, Lei Zhang, and Linjun Yang. CleanNet: Transfer Learning for Scalable Image Classifier Training with Label Noise. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018. 4

[23] Shan Li and Weihong Deng. Deep facial expression recognition: A survey. *arXiv preprint arXiv:1804.08348*, 2018. 1, 3, 5, 6

[24] Shan Li, Weihong Deng, and Jun Ping Du. Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017. 3, 7

[25] Yong Li, Jiabei Zeng, Shiguang Shan, and Xilin Chen. Patch-Gated CNN for Occlusion-aware Facial Expression Recognition. In *Proceedings - International Conference on Pattern Recognition*, 2018. 13

[26] Yuncheng Li, Jianchao Yang, Yale Song, Liangliang Cao, Jiebo Luo, and Li Jia Li. Learning from Noisy Labels with Distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017. 4

[27] Mengyao Liu and Dimitrios Kollias. Aff-Wild Database and AffWildNet. *arXiv preprint arXiv:1910.05318*, 2019. 9, 10

[28] Brais Martinez and Michel F. Valstar. Advances, challenges, and opportunities in automatic facial expression recognition. In *Advances in Face Detection and Facial Image Analysis*. 2016. 1

[29] Ali Mollahosseini, David Chan, and Mohammad H. Mahoor. Going deeper in facial expression recognition using deep neural networks. In *2016 IEEE Winter Conference on Applications of Computer Vision, WACV 2016*, 2016. 1

[30] Ali Mollahosseini, Behzad Hasani, and Mohammad H. Mahoor. AffectNet: A Database for Facial Expression, Valence, and Arousal Computing in the Wild. *IEEE Transactions on Affective Computing*, 2019. 4, 6, 7

[31] Sébastien Ouellet. Real-time emotion recognition for gaming using deep convolutional network features. *arXiv preprint arXiv:1408.3750*, 2014. 6

[32] Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep Face Recognition. In *Deep Face Recognition*, 2015. 9, 12

[33] Christopher Pramerdorfer and Martin Kampel. Facial expression recognition using convolutional neural networks: state of the art. *Facial expression recognition using convolutional neural networks: state of the art*, 2016. 5, 6

[34] Ali M. Reza. Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement. *Journal of VLSI Signal Processing Systems for Signal, Image, and Video Technology*, 2004. 5

[35] Evangelos Sariyanidi, Hatice Gunes, and Andrea Cavallaro. Automatic analysis of facial affect: A survey of registration, representation, and recognition, 2015. 1, 3

[36] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017. 13

[37] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2014. 6, 9

[38] Henrique Siqueira, Sven Magg, and Stefan Wermter. Efficient Facial Feature Learning with Wide Ensemble-based Convolutional Neural Networks. *arXiv preprint arXiv:2001.06338*, 2020. 6

[39] Sainbayar Sukhbaatar and Rob Fergus. Learning from Noisy Labels with Deep Neural Networks. *arXiv preprint*, 2014. 4

[40] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014. 6

[41] Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge Belongie. Learning from noisy large-scale datasets with minimal supervision. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017. 4

[42] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001. 5

[43] Kai Wang, Xiaojiang Peng, Jianfei Yang, Shijian Lu, and Yu Qiao. Suppressing uncertainties for large-scale facial expression recognition. *arXiv preprint arXiv:2002.10392*, 2020. 13

[44] Kai Wang, Xiaojiang Peng, Jianfei Yang, Debin Meng, and Yu Qiao. Region Attention Networks for Pose and Occlusion Robust Facial Expression Recognition. *IEEE Transactions on Image Processing*, 2020. 6, 13

[45] Xuehan Xiong and Fernando De La Torre. Supervised descent method and its applications to face alignment. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2013. 5

[46] Zhenbo Yu, Guangcan Liu, Qingshan Liu, and Jiankang Deng. Spatio-temporal convolutional features with nested LSTM for facial expression recognition. *Neurocomputing*, 2018. 5

[47] Jiabei Zeng, Shiguang Shan, and Xilin Chen. Facial expression recognition with inconsistently annotated datasets. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018. 4, 5, 6

[48] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. *IEEE Signal Processing Letters*, 2016. 5, 11

[49] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2014. 5

[50] Xiangyun Zhao, Xiaodan Liang, Luoqi Liu, Teng Li, Yugang Han, Nuno Vasconcelos, and Shuicheng Yan. Peak-piloted deep network for facial expression recognition. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016. 6