Data Preparation and Preprocessing For Aff-Wild2 Dataset

Dhruv Verma
Queen Mary University of London
Mile End Rd, Bethnal Green, London E1 4NS
ec211263@qmul.ac.uk
dhruvverma3098@proton.me

1. Introduction

Besides communicating ideas and thoughts, the face also conveys emotions as one of the most important aspects of human communication. All people of all cultures express anger, disgust, fear, sadness, sadness, and surprise the same way. This is one of the most fascinating aspects of facial expressions. Automatic facial expression analysis/recognition has gained a lot of traction in the past few years [4, 6–11, 13, 14, 20] due to its wide variety of applications. The problem remains challenging due to the subtlety, complexity, and variability of facial expressions [18].

FER or Facial Expression Recognition is applied in several areas such as safety, health, and human-machine interfaces. Researchers in this field are interested in developing techniques to interpret, code facial expressions and extract these features to have a better prediction by computer [16]. Deep Learning has emerged as a means to solve the problems encountered in this field, due to its architecture which mimics the human brain's ability to learn from experience, we can input raw data through the deep neural hierarchy to classify objects on which it has trained or has an experience. In this paper, we will describe how *ImageDataGenerator from keras module* was used to read and preprocess the Aff-Wild2 dataset. The task comes with its inherent complications and challenges involving reading, categorising, data imbalance, splitting, and applying further preprocessing techniques. The 6 main categories provided in the dataset are Anger, Disgust, Fear, Happiness, Sadness, and Surprise. However, the data size is small and it is not audio-visual [13].

The remainder of the paper has been organised as follows. **Section 2** is **Related Work** which refers to existing and ongoing research. **Section 3** is the hypotheses and restrictions faced in this field and how they can be altered, if possible. **Section 4** involves preparation of data which is then preprocessed in **Section 5. Section 6** concludes our paper and mentions future work.

2. Related Work

Facial Expression Recognition has attracted much attention from researchers over the last decade, here we will discuss some of the previous works to put this paper in context.

In [18], LBP features,i.e, Local Binary Pattern are investigated for low-resolution FER. The LBP operators were originally for texture analysis, and now they are employed for facial recognition. Dataset used by the researchers was the Cohn-Kanade database which consists of 100 university students aged from 18 to 30 years who were asked to perform 23 distinct facial expressions which are then categorized into six basic emotions. The research aimed to increase accuracy for descriptors such as the Facial Action Coding System which is a human-observer-based system developed to capture subtle changes in facial expressions. The LBP features were extracted on different resolutions to address a critical problem for real-world applications. A Boosted-LBP was formulated with the obtained LBP histograms with AdaBoost for each expression which resulted in improved recognition performance. The observed performance increase was noted to be 3-5%. Also since the performance of the boosted strong classifier originates in the characteristics of its weak hypothesis space. One of the noticeable limitations of this approach was that the recognition was performed on a static dataset without exploiting temporal behaviours of facial expressions. The same problem can be observed in [19] where similar datasets were used to tackle FACS Action Unit detection and an emotion recognition sub-challenge.

Facial movement features, which include feature position and shape changes, are generally caused by movements of facial elements and muscles during the course of emotional expression [21]. The vast majority of past work in FER doesn't take the dynamics movement features into account, so the authors took advantage of Gabor features followed by a patch matching operation. Patch-based Gabor features have shown excellent performance in overcoming position, scale, and orientation changes, as well as ex-

tracting spatial, frequency, and orientational information. Since real face detectors were taken into account for preprocessing, just cropping was done to a resolution of 48 * 48 pixels. Two databases were used, i.e, JAFFE and Cohn-Kanade to run train the recognition model. JAFFE database yielded an accuracy of 92.93% using four SVMs and four distances with Anger getting the highest recognition of 96.67%. Whereas for the CK database using similar four SVMs and four distances an accuracy of 94.4% was obtained with happiness getting recognized at 98.07%. It was proven from the results that patch-based Gabor features show a better performance over point-based Gabor features in terms of extracting the most useful information. The limitation of the approach originates from the use of static images and the lack of diversity in the database. Further improvements can be made by applying the suggested approach to a video-based FER system that works on multi frames.

A major limitation in the FER field stems from the images used to build such systems captured in a controlled environment such as CK+, MMI, and Oulu-CASIA [15]. The author mentions this problem in comparison to real-world applications where the occlusion of faces will pose a major issue. The occlusions may be caused by hair, glasses, scarf, breathing mask, hands, arms, food, and other objects that may be present in front of faces in daily life. The approach taken to resolve this issue leverages Deep Learning, a Convolution Neural Network (CNN) with an attention mechanism proposed which pays attention to the unblocked patches of a face and gets required information. RAF-DB and AffectNet, which are one of the biggest in-the-wild datasets are used to train the framework. Two versions of ACNN are used, pACNN and gACNN respectively which are then compared to existing state-of-the-art methods. An improvement of 8-9% was observed over the datasets by using the attention mechanism compared to the conventional CNN approach. Further limitations can be removed by generating attention parts in faces without landmarks as it is more suitable for real-life applications.

For [16] author provides us with in-depth insights into the existing and previous works done in the FER field, and states that we can capture verbal and non-verbal information from facial changes, tone of voice, and physiological signals. Face changes during communication are the first signs that transmit emotional state which attracts most researchers. Extraction methods such as Local Binary Patterns LBP, Facial Action Units FACs, Local Directional Patterns LDA, and Gabor Wavelet are mentioned as examples of traditional methods. DL is taken as the new efficient approach which involves automatic extraction of features using Convolution Neural Network CNN and Recurrent Neural Network RNN. All the available datasets are mentioned in detail comprising a different number of images and sizes. A few examples are MultiPie, MMI, CK+, etc. Several CNN/RNN models are discussed and found to have an accuracy of more than 99%. The drawback mentioned is the inaccuracy when dealing with real-life data as emotions cannot be categorized into just six basic emotions. Authors mention that multimodality is one of the conditions necessary to accurately detect human emotion.

3. Hypotheses-Restrictions

Among all the traditional methodologies and research, it can be observed that FER consists of six basic emotions, i.e., Anger, Disgust, Fear, Happiness, Sadness, Surprise and sometimes Neutral [19] but human expressions are much more complex than that. Even though the deployed frameworks using extraction methods ranging from Local Binary Pattern, Facial Action Units, Local Directional Patterns, and Gabor Wavelets, do get more than 90% accuracy but they are only half as accurate when dealing with real-life data. More robust approaches are required to deal with "in-the-wild" data(entirely uncontrolled environment). A common drawback observed in previous works is the lack of variety of data, since FER is an extremely complicated framework to build, it needs a massive dataset of images for just a single emotion to be completely understood. The presence of static images and lack of diversity in datasets also contribute to the same factor.

The drop in accuracy comes from reasons such as occlusion [18], blurring, lack of environmental light, low face detection rate, etc. A major inference in this field is that different emotions have different "salient" areas; however, the majority of these areas are distributed around the mouth and eyes. In addition, these "salient" areas for each emotion seem to not be influenced by the choice of using point-based or using patch-based features [21]. The larger the database, the larger the number of patches required. Now with the developments in the Deep Learning space, there have been advances which combine traditional models with deep neural networks such as Attention Mechanism [18]. Using a framework of Gabor Wavelets in combination with Convolution Neural Networks, a huge problem dealing with "in-the-wild" data was dealt with, which lays the groundwork for multi-frame analysis and video detection. The computational costs increase with every advancement in the field along with complexity which makes calculation times incompatible with real-world applications. Thus the model trained for such frameworks needs to be efficient, quick, reliable and able to deal with unexpected or undesired fluctuations in terms of "in-the-wild" data.

Facial feature detection such as eyes, mouth, nose and lips, plays a crucial role in understanding the large variability of spontaneous expressions and emotions, arising in uncontrolled environments [20]. For applications such as forensic analysis, driver detection in smart cars, surveil-

lance, etc. feature detection is an absolute necessity. Multi-task Cascaded Convolutional Networks (MTCNN) is a framework developed as a solution for both face detection and face alignment [2]. A solution like MTCNN has helped researchers in this field to accelerate and improve the modularity of their FER frameworks. As computing power increases exponentially along with developments in this field we can expect to find newer, more efficient algorithms.

4. Data Preparation

The dataset we are using for our project is a portion of Aff-Wild2 dataset. The provided

dataset has been artificially generated from the Aff-Wild2 database [4, 6–11, 13, 14, 20] using the

methods [15, 16, 18, 19]. Aff-Wild2 is an extension of the Aff-Wild dataset for affect recognition. Improved variability will contribute to the training accuracy of the model framework in future works. It approximately doubles the number of included video frames and the number of subjects; thus, improving the variability of the included behaviors and the involved persons [3, 5, 6, 12, 17]. The dataset comprises images belonging to six basic emotional expressions including Anger, Disgust, Fear, Happiness, Sadness, and Surprise.

The given dataset is in a zip file, so the first step is to extract the files on our server instance, in this case, Jupyter. *Zipfile* library is used for this extraction. Upon extraction six folders are obtained namely, 'ANGER', 'DISGUST', 'FEAR', 'HAPPINESS', 'SADNESS', and 'SURPRISE'. The number of files in each folder and the total is obtained (Table 1).

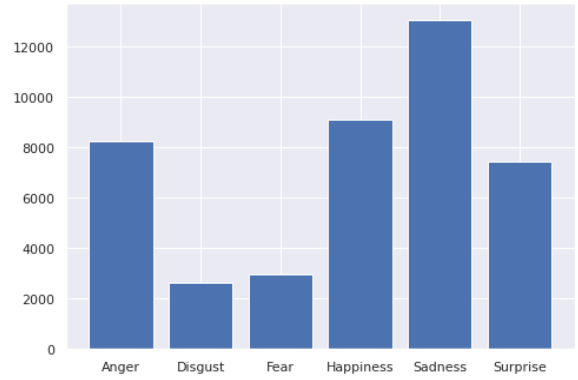
Emotion	Number of Images
Anger:	8228
Sadness:	13040
Happiness	9113
Surprise	7418
Fear	2985
Disgust	2651
Total	43435

Table 1. Dataset Details

The total number of files is 43,435 with SADNESS comprising 13040 images followed by HAPPINESS with 9113, ANGER with 8228, SURPRISE with 7418, FEAR with 2985, and DISGUST with 2651. We can see from the data that it is highly imbalanced(Fig. 1), so to deal with imbalance we will leverage the *class_weights* library from *sklearn* to balance the data during the training phase of our future works.

The next step in the process will be to check the data

Figure 1. Class Balance Visualisation



quality in terms of image size, image dimensions, and duplicates. Each image is observed to be of 4KB in size, 128 * 128 pixels and 3 channel RGB ranging from 0-255 in values. To deal with duplicates, we chose the *hashlib* library with the SHA512 algorithm to generate a hash of every image in the dataset and remove the images with the same images. After running the process it was found that there are 196 duplicates in the dataset with 193 images belonging to the DISGUST class, 2 images to the FEAR class, and 1 image to SURPRISE. After removing the duplicates following data was obtained in Table 2.

Emotion	Number of Images
Anger:	8228
Sadness:	13040
Happiness	9113
Surprise	7417
Fear	2983
Disgust	2458
Total	43239

Table 2. Dataset After Duplication Removal

Final number of total images is 43,239.

5. Data Pre-processing

A modern face recognition pipeline consists of 4 common stages: detect, align, normalize, and represent. At the preprocessing stage of the dataset, we will use the *preprocessing* library from *keras* [21] for the majority of the work. Before leveraging the Keras library we performed face detection and alignment using the *deepface* library. The *deepface* library uses MTCNN for face detection and alignment in real-time. The MTCNN method uses custom CNNs to solve simultaneously the problem of face detection and alignment in real-time. It consists of three sub-networks that process the faces from coarse to fine. Compared with

traditional methods, it has a better performance and faster detection speed, but it may show a low performance on low-quality images[22]. MTCNN detects eyes, nose, and mouth and performs geographic alignment so that eyes are horizontal in the final image. We can observe the alignment in Fig.2 and Fig.3.

Figure 2. Original Image

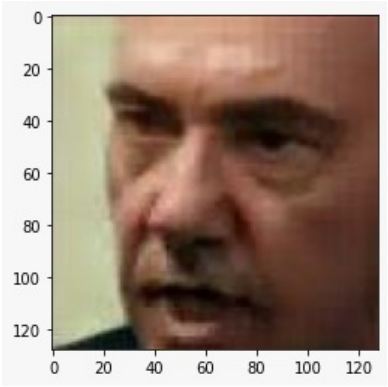
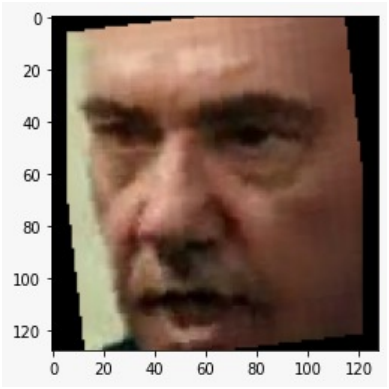


Figure 3. Image After Alignment



After alignment using MTCNN in the *deepface* library, we will employ *keras.preprocessing* to do the next phase of our preprocessing and do most of the heavy lifting. Keras' preprocessing library specializes in preparing 3 kinds of data before training which include images, sequences, and text. Since we are dealing with images only we will have all the utilities necessary to preprocess and augment the data [1]. We used *ImageDataGenerator* along with *flow_from_directory* to perform rescaling, normalization, resizing, random horizontal flipping, shuffling, filling, and 80-20 training-testing split into batches of 64.

Normalization is a technique often applied in the preparation of dataset. The goal of normalization is to change the values of numeric columns in the dataset to use a common scale, without distorting differences in the ranges of values or losing information, in this case from 0-255 to 0-1. This is called Min-Max normalization.

$$x\mathbf{n} = (x - x\mathbf{minimum})/(x\mathbf{maximum} - x\mathbf{minimum})$$

Data augmentation is a technique used to increase the amount of data by adding slightly modified copies of already existing data or newly created synthetic data from existing data. It acts as a regularizer and helps reduce overfitting when training a machine learning model.

Since original images have the value of 0-255 spread over 3 channels, we need to rescale and normalize the images between 0 and 1. This is done by passing the argument *rescale=1./255*. Also since the original images are 128*128 pixels, we need to compress that size to 48*48 to reduce the training time of the model. The argument for resizing is *target_size=48*48*. We also shuffle and flip training data horizontally by passing *shuffle=**true*** and *horizontal_flip=**True*** as part of data augmentation. *fill_mode='nearest'* is passed to fill the blank space in images after alignment. Finally *class_mode='categorical'* are passed to perform stratified splitting of data since we have six different and imbalanced classes. Batches of 64 images are formed with the following details Table 3:

Category	Splitting Division
Training:	34954
Testing:	8645

Table 3. 80:20 Training-Testing Split

The 6 classes found were encoded as follows(Table 4):

Class	Code
Anger:	0
Disgust:	1
Fear:	2
Happiness:	3
Sadness:	4
Surprise:	5

Table 4. Label Encoding

A label_batch and image_batch arrays are created and can be viewed as follows(Fig. 4 and Fig. 5):

Figure 4. Label Batch Sample

```
A random sample in label_batch is [1. 0. 0. 0. 0. 0.]
```

We also verified if the data has been properly normalised(Fig.10):

Figure 5. Image Batch Sample

```
A random image tensor after normalisation and augmentation
 [[[0.7176471  0.7294118  0.7372549  ... 0.34901962 0.1137255  0.05882353]
  [0.72156864 0.74509805 0.7411765  ... 0.16862746 0.08235294 0.11764707]
  [0.7411765  0.77647066 0.7607844  ... 0.14901961 0.10588236 0.09019608]
  ...
  [0.32156864 0.30980393 0.30980393 ... 0.12156864 0.12156864 0.1254902 ]
  [0.29411766 0.26666668 0.25490198 ... 0.13725491 0.1137255  0.19215688]
  [0.23529413 0.21176472 0.20392159 ... 0.1254902  0.21176472 0.29803923]]]
```

Figure 6. Min-Max Pixel Value after Preprocessing

```
Min and max pixel values: 0.0 and 1.0
```

6. Conclusion and Future Work

In the above paper, we have described and performed all the operations necessary to prepare data for a future DL framework. Future work will involve deploying a deep learning framework for FER and performing framework evaluations.

References

[1] Tf.keras.preprocessing.image.imagedatagenerator nbsp;: nbsp; tensorflow core v2.9.1. 4

[2] Deisy Chaves, Eduardo Fidalgo, Enrique Alegre, Rocío Alaiz-Rodríguez, Francisco Jáñez-Martino, and George Azzopardi. Assessment and estimation of face detection performance based on deep learning for forensic applications. *Sensors*, 20(16):4491, 2020. 3

[3] Paul Ekman. Basic emotions. *Handbook of cognition and emotion*, 98(45-60):16, 1999. 3

[4] Dimitrios Kollias. Abaw: Valence-arousal estimation, expression recognition, action unit detection & multi-task learning challenges. *arXiv preprint arXiv:2202.10659*, 2022. 1, 3

[5] Dimitrios Kollias, Shiyang Cheng, Maja Pantic, and Stefanos Zafeiriou. Photorealistic facial synthesis in the dimensional affect space. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018. 3

[6] 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*, 128(5):1455–1484, 2020. 1, 3

[7] Dimitrios Kollias, Mihalis A Nicolaou, Irene Kotsia, Guoying Zhao, and Stefanos Zafeiriou. Recognition of affect in the wild using deep neural networks. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pages 1972–1979. IEEE, 2017. 1, 3

[8] Dimitrios Kollias, Viktoriia Sharmanska, and Stefanos Zafeiriou. Face behavior a la carte: Expressions, affect and action units in a single network. *arXiv preprint arXiv:1910.11111*, 2019. 1, 3

[9] Dimitrios Kollias, Viktoriia Sharmanska, and Stefanos Zafeiriou. Distribution matching for heterogeneous multitask learning: a large-scale face study. *arXiv preprint arXiv:2105.03790*, 2021. 1, 3

[10] Dimitrios Kollias, Panagiotis Tzirakis, Mihalis A Nicolaou, Athanasios Papaioannou, Guoying Zhao, Björn Schuller, Irene Kotsia, and Stefanos Zafeiriou. Deep affect prediction in-the-wild: Aff-wild database and challenge, deep architectures, and beyond. *International Journal of Computer Vision*, pages 1–23, 2019. 1, 3

[11] Dimitrios Kollias and Stefanos Zafeiriou. Expression, affect, action unit recognition: Aff-wild2, multi-task learning and arcface. *arXiv preprint arXiv:1910.04855*, 2019. 1, 3

[12] Dimitrios Kollias and Stefanos Zafeiriou. Va-stargan: Continuous affect generation. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 227–238. Springer, 2020. 3

[13] Dimitrios Kollias and Stefanos Zafeiriou. Affect analysis in-the-wild: Valence-arousal, expressions, action units and a unified framework. *arXiv preprint arXiv:2103.15792*, 2021. 1, 3

[14] Dimitrios Kollias and Stefanos Zafeiriou. Analysing affective behavior in the second abaw2 competition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3652–3660, 2021. 1, 3

[15] Yong Li, Jiabei Zeng, Shiguang Shan, and Xilin Chen. Occlusion aware facial expression recognition using cnn with attention mechanism. *IEEE Transactions on Image Processing*, 28(5):2439–2450, 2018. 2, 3

[16] Wafa Mellouk and Wahida Handouzi. Facial emotion recognition using deep learning: review and insights. *Procedia Computer Science*, 175:689–694, 2020. 1, 2, 3

[17] Andreas Psaroudakis and Dimitrios Kollias. Mixaugment & mixup: Augmentation methods for facial expression recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2367–2375, 2022. 3

[18] Caifeng Shan, Shaogang Gong, and Peter W McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and vision Computing*, 27(6):803–816, 2009. 1, 2, 3

[19] Michel F Valstar, Marc Mehu, Bihan Jiang, Maja Pantic, and Klaus Scherer. Meta-analysis of the first facial expression recognition challenge. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(4):966–979, 2012. 1, 2, 3

[20] Stefanos Zafeiriou, Dimitrios Kollias, Mihalis A Nicolaou, Athanasios Papaioannou, Guoying Zhao, and Irene Kotsia. Aff-wild: Valence and arousal 'in-the-wild'challenge. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pages 1980–1987. IEEE, 2017. 1, 2, 3

[21] Ligang Zhang and Dian Tjondronegoro. Facial expression recognition using facial movement features. *IEEE transactions on affective computing*, 2(4):219–229, 2011. 1, 2