Facial Expression Recognition in the Wild

Damith Chamalke Senadeera
Queen Mary University of London
d.c.senadeera@se21.qmul.ac.uk

1. Introduction

Facial Expression Recognition in uncontrolled real-life environment which is also known as facial expression recognition in the wild has become an important research topic as it plays a pivotal role in intelligent social interaction [31]. Also, according to [31], most of the traditional facial expression recognition systems mainly are focused only on the captured frontal face expressions performed by actors in controlled environments and therefore facial expression recognition in the wild will be more challenging due to factors such as different poses of the face, changes in illuminations in the face and subtle expression variations shown by real people in low resolution images. According to [31] in general, features are extracted from the inputted facial images manually or automatically and using these features these inputted images are classified in to one of the basic six facial expressions (i.e., happiness, surprise, disgust, fear, sadness, anger) as described in [8].

2. Related Work

2.1. Techniques Based on Hand-Crafted Features

Attempts have been made to formulate accurate facial expression recognition systems starting from the pre deep learning era. In 2003, [28] has suggested a mechanism to infer emotions automatically using facial expression recognition in a live video using a face feature tracker. The proposed methodology in [28] has introduced a novel unobtrusive face feature tracker based on a face template. They have used a filter to track the feature positions over subsequent frames to facilitate the calculation of feature displacements by getting the Euclidean distance between a "peak" frame and a neutral frame which had in turn been used as extracted features for the SVM, where these features do not need pre-processing and can be computed real-time which could be seen as another novelty at that time [28].

Cohn-Kanade Facial Expression Database which consisted of face images of 100 university students displaying 23 facial emotions including the 6 basic emotions [8] have been used in [28] to train and test their algorithm for still images and image frames custom generated from videos from expert and ad-hoc users have been used to evaluate on video data. [28] has reported 87.9% as the best average accuracy for Cohn-Kanade database and for the performance evaluation with frames extracted from videos, [28] obtained an average accuracy of 87.5% for data from an expert user and 60.7% for data from ad-hoc interactions. As constraints seen in [28], it's worth to note that they have used an extremely small number of training-testing datasets which might not help in generalization and also inaccuracies might be introduced due to the head motion and by normalizing all features with regards to only one root feature which also didn't take into account the rotational motions of the head.

In 2005, [30] has attempted to introduce a new algorithm to recognize facial expressions using Local Binary Patterns (LBP) as a feature extraction mechanism also evaluating the effect of these chosen features on the performance of facial expression recognition on low resolution face images. The authors in [30] have introduced LBP as a low-cost novel discriminative feature extraction mechanism with a SVM in their algorithm compared with the state-of-the-art facial feature extraction mechanism at that time using Gabor-wavelet appearance feature which had been computationally expensive. LBP operator has been used to extract facial features in a reasonable time representing a good amount of facial information.

They also have used Cohn-Kanade Facial Expression Database to train and test their algorithm and have obtained the best accuracy of 88.4%. Further they have investigated the LBP features on facial images with low resolution up to 14x19, and have achieved an accuracy of 75.8% for the smallest resolution. However, on the downside LBP may be slow in processing large data sets and it tends to miss the local structure under some circumstances due to not considering the centre pixel in LBP histogram and also it is sensitive to noise [9].

In the paper [9], they have addressed the limitations of the work of [30] in facial expression recognition by introducing a novel Centralized Binary Pattern (CBP) operator which reduces the dimensionality of the feature histogram while reducing the effect of the noise and improving dis-

crimination of the features. They have also embedded image Euclidean distance in CBP to improve robustness for small deformations in face images.

They have used the JAFFE database which includes facial expression images from 10 Japanese female expressors for 7 expressions (6 basic facial expression and neutral face) and Cohn-Kanade database which was discussed earlier for training and testing their model in [9] and obtained a 94% recognition accuracy for the testing with JAFFE database while 94.86% accuracy has been reported based on Cohn-Kanade database. They also only have tested their model on grey scale images and feature extraction is manually crafted similar to the methods discussed earlier.

2.2. Techniques Based on Deep Learning

With the discovery of deep learning techniques, many researchers have deviated their focus from manual feature extraction techniques such as discussed in the work of [28], [30] and [9], towards automated feature extraction for facial expression recognition. In the work of [13], in 2015 they have introduced a novel deep neural network architecture for facial expression recognition with the novelty of automatic facial feature extraction using a Convolutional Neural Network (CNN) compared to the above discussed manual feature extraction methodologies. They have used 2 deep neural networks in their architecture, first one being the CNN to extract the temporal appearance features from a face image sequence and the second one being another deep neural network to extract features from facial landmark points. Both these models are fine-tuned together to minimize the classification loss [13].

In [13] they have used 3 databases for training and testing their model namely, CK+ database which consist of 327 images representing 7 emotions (6 basic facial expressions + contempt expression) where they have reported an overall best accuracy of 97.25%, Oulu-CASIA database which consist of 480 images captured under normal lighting conditions representing 6 basic facial expressions where the best overall accuracy has been reported as 81.46% and the MMI data base consisting of 205 images of frontal faces for 6 basic emotions where the reported overall best accuracy is 70.24%. Even they train their models and report the performance only on grey scale images in contrary to using colour images and real world conditions are not taken into consideration specifically as well.

In 2017 [27] has introduced an approach using a CNN architecture to compensate the difficulties that the facial expression recognition systems face in real world conditions due to variations in different face subjects by capturing both expression related and identity related features. The authors have proposed the novel identity aware convolutional neural network (IACNN) which reduces the effects introduced by personal attributes by the use of an expression sensitive contrastive loss during the training of the network to attain identity expression recognition which is invariant to identity using two identical component Convolutional Neural Networks where weights were shared.

Authors of [27] have conducted extensive experimentation on 3 different facial expression image databases namely, CK+, MMI and SFEW database which consists of 1766 images depicting 7 facial expressions (6 basic facial expressions + neutral face) with their algorithm and have reported a 95.37% overall accuracy on CK+ database, 71.55% average accuracy on MMI database and 50.98% validation accuracy and 54.3% testing accuracy on the SFEW database. Even though the authors in [27] try to reduce the effect of a real world issue faced during facial expression recognition in the wild, this algorithm only works on still images and it is not quite useful for facial expression recognition in video frames as temporal information are not used.

In 2020 the authors of the paper [32] came up with a novel deep learning solution to address the issues of occlusion and pose variations seen in facial expression recognition task in the wild, by the use of a novel Region Attention Network (RAN) to capture the relevance of facial regions for occlusion and position variations in facial expression recognition in an adaptive manner. A numerous number of region-features which are processed by a backbone CNN are collected and embedded by a RAN using a self-attention and a relation attention module by making use of a Region Biased Loss (RB-Loss) which tries to learn a high attention weight for the region with highest importance in turn producing a model robust for occlusion and pose variations in facial images.

The authors of [32] have built custom datasets using the public facial expression databases of FERPlus which consist of 35887 real world facial images depicting 8 facial expressions (6 basic expressions + neutral face + contempt), AffectNet with around 1 million images where around 450000 images are manually annotated for 8 facial expressions same as of the FERPlus and RAF-DB database which contains 30000 images with basic or compound expressions with occlusion and pose annotations for training the model. They have evaluated the model on FERPlus, AffectNet, RAF-DB and SFEW databases to obtain accuracies of 89.16%, 59.5%, 86.9% and 56.4% respectively.

After the introduction of Vision Transformer Architecture [6] which became a base SOTA deep learning architecture for the image classification problem in the recent past, authors of [33] had tried to incorporate this novel Visual Transformers (ViT) for facial expression recognition (FER) which could learn local representations which are relation aware using 3 components Multi Attention Dropping(MAD) which is used to randomly drop an attention map, ViT-FER which is the ViT used in FER and Multi-head Self-Attention Dropping (MSAD) which is used to

randomly drop a single self-attention module. In [33], they have used RAF-DB and AffectNet databases to train and test their algorithm and at the time of the publication of their results they reported the best SOTA accuracies for these 2 databases which stood at 90.91% and 66.23% respectively.

3. Hypotheses-Restrictions

Some of the restrictions faced during the facial expression recognition in the wild based on the 6 basic facial expressions discussed in [8] are

1. Substantial inter class similarity between facial expression images in wild belonging to different basic facial expression classes. For example- the mouth regions of the images selected from two different facial expression labels "surprise" and "anger" from our dataset, are almost similar as seen in figure 1. So, this means sometimes, the facial expressions tagged into different class labels might only show some subtle differences at given scenarios [33].

2. Little intra class similarity between facial expression images in wild belonging to one particular basic facial expression class. Given the demographics of the person, same facial expression between different people with different demographics such as gender, race or age might highly differ [33].

3. Facial expression representation might highly differ or may not be properly visible due to occlusion and pose variations of the face image [32] or illumination changes or due to unconstrained backgrounds present in real life scenarios while capturing the face images as well. [33].

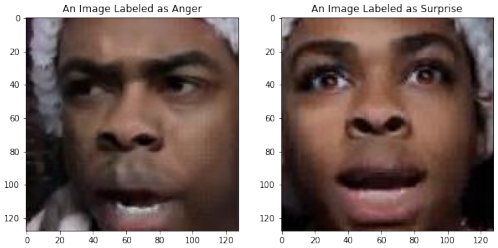

Figure 1. Two Similar Images from Two Class Labels

In order to address these restrictions, we can use more and more facial images captured in the wild by general public with different demographic factors compared to facial images captured in constrained laboratory settings where facial expressions are performed by experts. More the real-world training data used in training face recognition systems, better the model will generalize to learn the real-world facial expression settings [35].

Also, as explained in the paper [24], if we have access to facial images annotated with additional data such as occlusions, head poses and illumination conditions which can directly affect the facial expression recognition in the wild, we can make use of these additional annotations to help steer our recognition models in the right direction. Another important aspect which is also discussed in [24] is that, if we have access to auxiliary information such as audio data related to the specific facial expression images, we can make use of this to devise a multimodal system which is trained using images and the related voice information to properly recognize the displayed facial expression.

4. Data Preparation

For this task of Facial Expression Recognition in the wild, the data set used has been generated synthetically using the methods mentioned in [16], [23], [15] and [29] based on an image set obtained from Aff-Wild2 database [14], [25], [18], [20], [24], [22], [19], [21], [34], [17] where images are captures as frames from the videos in this database.

All the images are labelled according to the 6 basic emotions [8]. The data set is read using the opencv python package [2] and 2 lists are created, one being a list of all the images and the other one being the list of labels corresponding to each image and the distribution of images among the 6 labels were analysed in figure 2.

We can observe that the data is not distributed in a balanced manner among classes. Therefore, during training and validation data splitting, pre-processing and during training the model to recognize the facial expressions, we have to address this class imbalance issue.

To avoid data leakage from the training set to the validation set, before splitting the data, the exact duplicates among the images were searched using hash comparison with the image hashes calculated with sha1 algorithm [7] using the hashlib python package [1] to get the unique representation of the images. A total of 196 duplicates were detected where 1 image from "Surprise" label, 2 from "Fear" label and 193 from "Disgust" label were reported. After removing the duplicates, final distribution of the image set is shown in the table 1.

Afterwards, text labels are converted into label encoding using the "labelencoder" function of sklearn python package [4] and then the whole data set is split into training and validation sets in 80% and 20% ratio to obtain generalized results avoiding overfitting according to the empirical study [10], preserving the same ratio for each class label as a stratified split after a random shuffle to ensure that the original data is split representatively among the train and validation sets as reported in table 2. The random seed was set into 0 for the sake of reproducibility of this training and validation split.

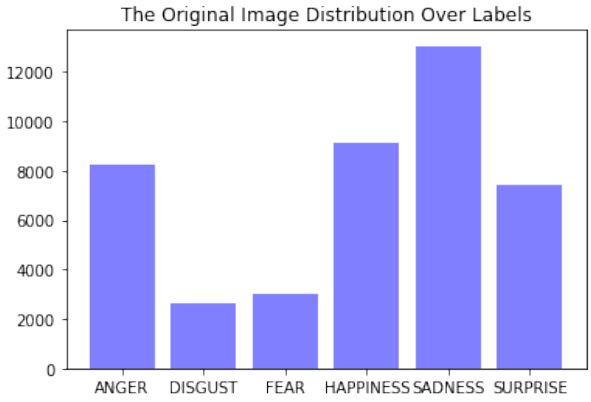

Figure 2. Original Data Distribution Histogram

Class Label	No. of Images before duplicate removal	No. of Images after duplicate removal
Anger	8228	8228
Disgust	2651	2458
Fear	2985	2983
Happiness	9113	9113
Sadness	13040	13040
Surprise	7418	7417
Total	**43435**	**43239**

Table 1. Original Image Distribution

Class Label	Label Encoded Class Label	No. of Images in training set	No. of Images in validation set
Anger	0	6582	1646
Disgust	1	1967	491
Fear	2	2386	597
Happiness	3	7290	1823
Sadness	4	10432	2608
Surprise	5	5934	1483
Total		**34591**	**8648**

Table 2. Training and Validation Image Distribution

5. Data Pre-processing

Afterwards, the data is pre-processed in order to increase the consistency among the training data under a specific label and to make the data smooth in order to make it easier for interpretation.

As the initial data quality assessment, the source of the data was investigated where the data set has been synthetically created using techniques mentioned in [16], [23], [15] and [29] by inputting a set of video frames obtained from the Aff-Wild2 database discussed earlier. All the images are JPEG RGB – 3 channel images with a resolution of 128 x 128 pixels for each image. Also, all the images contained only the face images which have been cropped.

During data preparation step, data duplicates were already searched for and those found duplicates were removed to stop data leakage from training set to the validation set. So, at this stage all the exact duplicates are removed from the data set.

Since all the faces are detected and cropped in this dataset, we investigated other pre-processing techniques which would be useful for our task of facial expression recognition. As mentioned in [5] if the faces are aligned using the facial landmarks such as the position of the left eye, right eye, nose, mouth, etc. of the face, the accuracy of facial expression recognition has significantly improved. Since it was evident that the face images in the data set are not aligned during data quality assessment, a pipeline was devised to align the face images both in training and validation data sets by detecting facial landmarks of left eye, right eye and the nose.

To detect these facial landmarks, the algorithm of Multi task Cascaded Convolutional Networks (MTCNN) [36] which uses 3 convolutional neural networks to detect the facial landmarks was chosen as it has shown superior performance although the time taken to process images is somewhat higher than the techniques used in trivial python packages such as opencv or dlib. Once the facial landmarks of the left eye, right eye and the nose are extracted, an affine transformation is calculated using the position of the left eye and the right eye to bring those 2 points into a horizontal line and this transformation is applied to the whole image to rotate it so that the 2 eyes are aligned. The python package MTCNN [12] is used to detect these facial landmarks and then the whole image is aligned using those landmarks. If the algorithm is unable to detect the landmarks in an image, that particular image is not aligned. Statistics for the alignment are shown in the table 3.

After alignment, for both the training and validation sets, the images are resized to 96 x 96 for computational efficiency to process images during training of the model in GPUs in batch wise and then the images are normalized by dividing each of the pixels in the images by 255, to bring the values of the pixels into the range of 0 to 1 as the original values were ranging from 0 to 255 as deep neural networks have shown to work better with normalized pixel values according to [26]. An example is displayed in figure 3

After all these pre-processing steps, to address the problem of class imbalance data in the training data set, images

Class Label	Training Aligned No. of Images	Training Non-Aligned No. of Images	Testing Aligned No. of Images	Testing Non-Aligned No. of Images
Anger	6353	229	1596	50
Disgust	1879	88	460	31
Fear	2314	72	573	24
Happiness	7106	184	1789	34
Sadness	10213	219	2583	69
Surprise	5674	260	1430	53
Total	**33539**	**1052**	**8431**	**261**

Table 3. Training and Validation Image Distribution

Figure 3. An Example of a Pre-processed Image

are augmented to make the number of images for each class label almost equal only in the training set. For this augmentation of data in training set, only perturbation method of adding noise which is also label preserving is used as the other methods such as transformations such as rotations are not applied because we have already aligned the images. Also, according to [11] since a little amount of noise in images tested for inference can lead for complete wrong predictions from the model, by adding noise as mentioned in [11] during the training step, can help make the models more robust to noise and also can help to identify the places where the wrong predictions can be made.

Therefore, to address this class imbalance problem and also to help train a robust deep neural network model for this problem we added augmented data using the function "random_noise" in python package skimage [3] based on "Poisson Noise" which is an additive noise sampled from the Poisson distribution, "Gaussian Noise" which is an additive noise sampled from Gaussian Distribution and "Speckle Noise" which is a multiplicative noise sampled from gaussian distribution according to the combination given in the

Class Label	Augmentation Methodology Used
Anger	Data with each Poison Noise, Speckle Noise and Gaussian Noise for 20% of random pre-augmented data is added as new data.
Disgust	Data with each Poison Noise, Speckle Noise and Gaussian Noise for the whole pre-augmented data is added as new data.
Fear	Data with each Poison Noise, Speckle Noise and Gaussian Noise for the whole pre-augmented data is added as new data.
Happiness	Data with each Poison Noise, Speckle Noise and Gaussian Noise for 10% of random pre-augmented data is added as new data.
Sadness	Data with each Poison Noise, Speckle Noise and Gaussian Noise for 10% of random pre-augmented data is replaced.
Surprise	Data with each Poison Noise, Speckle Noise and Gaussian Noise for 20% of random pre-augmented data is added as new data.

Table 4. Training Data Augmentation Methodology

Class Label	No. of Images in pre-augmented training set	No. of Images in post-augmented training set
Anger	6582	9543
Disgust	1967	7868
Fear	2386	9544
Happiness	7290	9477
Sadness	10432	10432
Surprise	5934	9492
Total	**34591**	**56356**

Table 5. Training and Validation Image Distribution

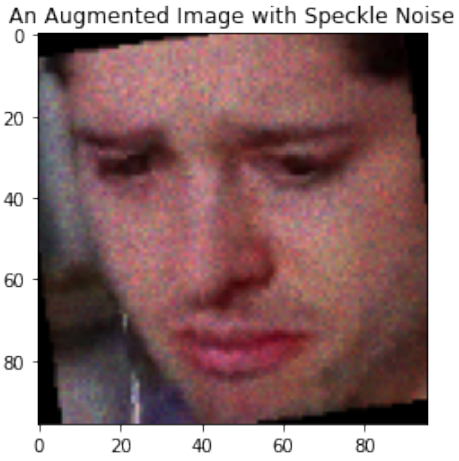

Figure 4. An Example of an Augmented Image

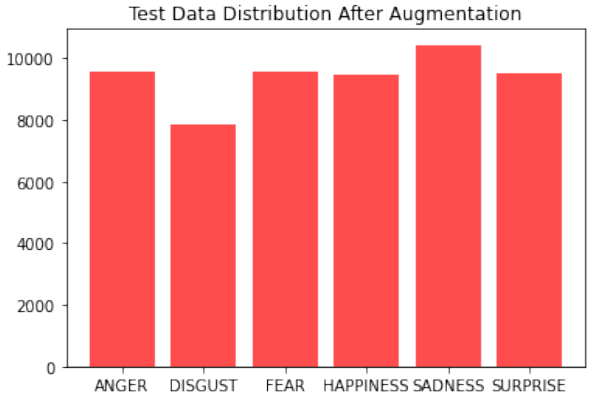

Figure 5. Augmented Data Distribution Histogram

table 4 using the random seed as 0 for the sake of reproducibility and the final augmented statistics for each class label are reported in the table 5. For the class "Sadness" which contained the most number of training samples, some data were retrieved and augmented with noise and replaced in the training data set so as to preserve generality of noise among all the 6 classes. An example of an augmented image is displayed in figure 4. After augmenting the training data, the images for a specific class label were again reshuffled to promote representativeness keeping the random seed as 0 for reproducibility.

As seen in the histogram in figure 5, the class imbalanced issue has been minimized and to further address this issue, class weights will be used while training the model in the loss function and also F1 score will be also used during evaluation as another measure to mitigate the impact from this problem.

At the end all the labels were converted to one-hot encoded labels and the training and the validation sets were saved.

References

[1] Hashlib - secure hashes and message digests¶. 3

[2] Opencv modules. 3

[3] scikit-image: Image processing in python. 5

[4] scikit-learn machine learning in python. 3

[5] Romain Belmonte, Benjamin Allaert, Pierre Tirilly, Ioan Marius Bilasco, Chaabane Djeraba, and Niculae Sebe. Impact of facial landmark localization on facial expression recognition. *IEEE Transactions on Affective Computing*, 2021. 4

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 2

[7] D Eastlake 3rd and Paul Jones. Us secure hash algorithm 1 (sha1). Technical report, 2001. 3

[8] Paul Ekman. Basic emotions. *Handbook of cognition and emotion*, 98(45-60):16, 1999. 1, 3

[9] Xiaofeng Fu and Wei Wei. Centralized binary patterns embedded with image euclidean distance for facial expression recognition. In *2008 Fourth International Conference on Natural Computation*, volume 4, pages 115–119. IEEE, 2008. 1, 2

[10] Afshin Gholamy, Vladik Kreinovich, and Olga Kosheleva. Why 70/30 or 80/20 relation between training and testing sets: a pedagogical explanation. 2018. 3

[11] Zhezhi He, Adnan Siraj Rakin, and Deliang Fan. Parametric noise injection: Trainable randomness to improve deep neural network robustness against adversarial attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 588–597, 2019. 5

[12] Ipazc. Ipazc/mtcnn: Mtcnn face detection implementation for tensorflow, as a pip package. 4

[13] Heechul Jung, Sihaeng Lee, Junho Yim, Sunjeong Park, and Junmo Kim. Joint fine-tuning in deep neural networks for facial expression recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 2983–2991, 2015. 2

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

[15] 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, 4

[16] 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. 3, 4

[17] 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. 3

[18] D Kollias, A Schulc, E Hajiyev, and S Zafeiriou. Analysing affective behavior in the first abaw 2020 competition. In *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)(FG)*, pages 794–800. 3

[19] 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. 3

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

[21] 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. 3

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

[23] 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, 4

[24] 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. 3

[25] 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. 3

[26] Shan Li and Weihong Deng. Deep facial expression recognition: A survey. *IEEE transactions on affective computing*, 2020. 4

[27] Zibo Meng, Ping Liu, Jie Cai, Shizhong Han, and Yan Tong. Identity-aware convolutional neural network for facial expression recognition. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 558–565. IEEE, 2017. 2

[28] Philipp Michel and Rana El Kaliouby. Real time facial expression recognition in video using support vector machines. In *Proceedings of the 5th international conference on Multimodal interfaces*, pages 258–264, 2003. 1, 2

[29] 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, 4

[30] Caifeng Shan, Shaogang Gong, and Peter W McOwan. Robust facial expression recognition using local binary patterns. In *IEEE International Conference on Image Processing 2005*, volume 2, pages II–370. IEEE, 2005. 1, 2

[31] Jie Shao and Yongsheng Qian. Three convolutional neural network models for facial expression recognition in the wild. *Neurocomputing*, 355:82–92, 2019. 1

[32] 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*, 29:4057–4069, 2020. 2, 3

[33] Fanglei Xue, Qiangchang Wang, and Guodong Guo. Transfer: Learning relation-aware facial expression representations with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3601–3610, 2021. 2, 3

[34] 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. 3

[35] Feifei Zhang, Tianzhu Zhang, Qirong Mao, Lingyu Duan, and Changsheng Xu. Facial expression recognition in the wild: A cycle-consistent adversarial attention transfer approach. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 126–135, 2018. 3

[36] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE signal processing letters*, 23(10):1499–1503, 2016. 4