IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

PyFER: A Facial Expression Recognizer based on Convolutional Neural Networks

Abdullah Talha Kabakus

Department of Computer Engineering, Faculty of Engineering, Duzce University, Duzce, 81620, TURKEY

Corresponding author: Abdullah Talha Kabakus (e-mail: talhakabakus@duzce.edu.tr).

ABSTRACT Facial expression recognition (FER), one of the most trending research areas of the Human-Machine Interaction, is the task of detecting emotions by analyzing facial expressions and this analysis plays a critical role as it conveys the clearest information regarding the emotions of people. Despite the fact that the traditional machine learning algorithms produce high accuracies for similar tasks, they lack to detect emotions of faces, which are captured in a spontaneous manner (*a.k.a.* "in the wild") or in different poses or environmental conditions. In this paper, a novel convolutional neural network architecture, namely, *PyFER*, is proposed to address the FER problem, of which the efficiency was revealed thanks to the experiments conducted on a widely-used benchmark dataset. According to the experimental results, the accuracy of *PyFER* was calculated to be as high as 96.3% on a de-facto standard dataset, namely, *CK+*, and all facial expressions, except for *happiness*, were correctly detected by *PyFER*, which is encouraging for future studies. 16.67% of the images that actually represented the facial expression happiness were misdetected as the facial expression fear. The experimental results confirmed that the proposed neural network architecture is fast enough to be integrated into real-time FER applications as it was able to complete the analysis of a given photo for an average of 12.8 milliseconds, which is in the tolerable limit to latency for real-time applications.

INDEX TERMS Artificial intelligence, Artificial neural networks, Backpropagation, Multi-layer neural network, Neural networks, Supervised Learning

I. INTRODUCTION

Emotion detection plays an important role in many areas including but not limited to intelligent security [1], robotics manufacturing [2], clinical psychology [3], multimedia [4], and automotive security [5]. Facial expression recognition (FER), which is an important research area of Human-Machine Interaction (HMI), is the task of detecting emotions by analyzing facial expressions that play a key role in social interaction [6] and convey meaningful and clear information about the emotions of people [7]. As a natural consequence of that, various computer vision systems based on machine learning algorithms have proposed FER where they were trained using annotated face datasets. Despite the fact that traditional machine learning algorithms produce high accuracies, they lack to detect emotions of faces captured in a spontaneous manner (*a.k.a.* "in the wild") or when they are applied to a dataset that they were not trained on [6]–[8]. In addition to that, when the fact that FER datasets are constructed in highly controlled pose conditions is considered, the proposed approach specifically designed to not utilize traditional machine learning algorithms. Instead, deep neural networks, which are more capable of extracting features from the training data [9] and reportedly being able to produce high accuracy even for the face poses in the wild [10], are utilized within the proposed approach. Being able to self-learn important features makes deep neural networks a common choice for classification problems in which spatial information plays a critical role [9]. In the case of FER, being able to extract spatial features is a key aspect considering the fact that classification highly depends on the shape of facial features such as the eyes, mouth, and eyebrows [9]. Another advantage of using a deep neural network instead of a traditional machine learning algorithm is that when deep neural networks are trained on large datasets, they extract generalized features, which could be applied on datasets that the network has not been trained in [6]. Various datasets have used in the literature. For the proposed system, a de-facto standard well-known dataset, namely, *CK+*, was utilized.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3012703, IEEE Access

Some sample photos from the *CK+* dataset [11], [12] with their representative facial expression labels are presented in Figure 1.

FIGURE 1. Some sample photos from the *CK+* dataset with their representative facial expression labels.

FER applications take the photos of subjects as the input and produce the detected emotions through various analyses as the output. The target emotions vary through the proposed approach which could be happiness, sadness, surprise, anger, disgust, fear, contempt, and neutral. The general architecture of FER approaches contains three phases, namely, (1) preprocessing, (2) feature extraction, and (3) classification phase. In the first phase, the preprocessing phase, the quality of the input images is enhanced and the redundant information is removed. In the feature extraction phase, the preprocessed input data are transformed into the best representative features in order to lead a sensitive and flexible classification technique that could perform the right predictions in terms of emotions. In the last phase, the classification phase, mapping of input data to the target emotions takes place by virtue of the utilized classification algorithm. This paper presents a novel convolutional neural network architecture, namely, *PyFER*, for the FER problem, which was applied to a widely-used benchmark dataset in order to reveal its effectiveness on the detection of facial expression in various situations such as the angles of poses, the human races and genders, the way the emotion is expressed, and the environmental effects (i.e. light intensity, background, etc.). The main contributions of this study are listed as follows:

- A novel CNN architecture, which was specifically designed to be able to detect the facial expression for a given photo in the tolerable limit to latency for real-time applications, was proposed whose accuracy was calculated to be as high as 96.3% on a de-facto standard benchmark dataset.
- The proposed architecture (including all the preprocessing tasks) was implemented using open-source software (e.g. various Python libraries, *OpenCV*), which makes the extensibility of the model possible. Thanks to the implemented automatized tasks, the effect of various hyper-parameters on the proposed deep convolutional neural network experimented, and the ones would that provide the best accuracy were revealed.
- The experiences which were experimented during the optimization of the proposed architecture on the validation set were discussed in order to share the 'learned lessons' as a great contribution to the field.

The rest of the paper is structured as follows: Section 2 briefly presents the related work. Section 3 describes the proposed deep convolutional neural network architecture, namely, *PyFER*, to handle the FER problem. Section 4 presents the experimental result and discussion. Finally, Section 5 concludes the paper with future directions.

II. RELATED WORK

Szegedy et al. [13] proposed a deep neural network architecture based on convolutional neural networks, namely, *GoogLeNet*, which consisted of 22 layers (27 layers when the pooling layers are considered, too). It was introduced during the *ImageNet Larger Scale Visual Recognition Challenge 2014 (ILSVRC 2014)*[1]. *Mollahossein et al.* [6] proposed a deep neural network architecture that was inspired by *GoogLeNet* and *AlexNet* [14]. This approach classifies the input facial photos into either of seven emotions namely, $anger$, $disgust$, $fear$, $happiness$, $sadness$, $surprise$, and $neutral$. *Zhang et al.* [15] proposed a feature learning model based on a novel deep neural network with the SIFT (Scale Invariant Feature Transform) features. *Jung et al.* [16] proposed a deep neural network that was based on two different models. While the first deep neural network extracted temporal appearance features from the photos, the other deep neural network extracted temporal geometry features from temporal facial landmark points. The experiments conducted proved that a combination of these two models boosted the performance of the FER task. *Lopes et al.* [17] proposed an approach that was a combination of a convolutional neural network and specific preprocessing steps. The conducted experiments proved that the combination of the normalization procedures improved the accuracy significantly. *Majumder et al.* [18] proposed a FER system, namely, *AFERS*, which consisted of four steps as follows: (1) geometric feature extraction, (2) regional local binary pattern (LBP) extraction, (3) fusion of both of the features using auto-encoders, and (4) classification using a classifier based on Kohonen self-organizing map. *Hamester et al.* [19] proposed a new deep neural network architecture based on the Multi-Channel Convolutional Neural Network (MCCNN). *Alizadeh and Fazel* [20] proposed various convolutional neural networks and evaluated their performance. According to the experimental result, they claimed that hybrid feature sets did not improve the model's accuracy, which implies that convolutional neural networks substantially learn the

[1] http://www.image-net.org/challenges/LSVRC/2014/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3012703, IEEE Access

necessary features through the raw pixel data. *Burkert et al.* [21] proposed an emotion recognition model, namely, *DeXpression*, which consisted of a pair of $convolutional$, $pooling$, and $Rectified\ Linear\ Unit\ (ReLU)$ layers. *Anderson* and *McOwan* [22] proposed a face expression system that was capable of recognizing six basic emotions, namely, $happiness$, $sadness$, $disgust$, $surprise$, $fear$, and $anger$. The proposed system consists of three components as follows: (1) a face tracker which is responsible to detect the location of the face, (2) an optical flow algorithm to track the motions on the face, and (3) the recognition engine which is based on *SVM* and *Multilayer Perceptron* algorithms. *Kotsia* and *Pitas* [23] proposed a FER approach based on geometric deformation features such as Candide grids, and *SVM* in order to detect six basic facial expressions, namely, $anger$, $disgust$, $fear$, $happiness$, $sadness$, and $surprise$. *Shan et al.* [24] proposed a FER approach based on *Local Binary Patterns (LBP)*, which were calculated over the facial region. According to the experimental results, the best recognition performance was achieved through *SVM* classifiers with *Boosted-LBP* features. *Zafer et al.* [25] proposed an algorithm, namely, *Robust Normalized Cross-correlation Coefficient (RNCC)* for face recognition with expression variation where the outliner pixel effects the template matching too strong or too weak were excluded. *Song et al.* [26] proposed a deep neural network architecture that consisted of 5 layers with a total of $65K$ neurons. $Convolutional$, $pooling$, local filter layers, and fully connected layers (*a.k.a.* $dense$ layers) are utilized within the proposed network alongside both $Dropout$ and data augmentation in order to avoid overfitting, which is a serious issue when the size of the network gets larger [26]. *Garcia-Ramirez et al.* [27] proposed a FER system based on three stages: The pre-processing stage is responsible for mouth and eyebrow segmentation. The second stage is feature extraction where polynomials utilized as features. And, the third stage is responsible for classification where different supervised learners such as neural networks, *K-Nearest Neighbors*, and *C4.5* decision trees were utilized.

III. MATERIAL AND METHOD

In this section, both of the dataset which was used to evaluate the proposed convolutional neural network, and the detail of the proposed deep neural network architecture are described.

A. DATASET

The *Extended Cohn-Kanade Dataset (CK+)* is a standard FER benchmark dataset that consists of 210 subjects at the ages of 18 to 50 years. The dataset contains a great variety of photos of both genders with different ages having different backgrounds. The sizes of the photos in this dataset are $640x480$ and $640x490$ pixels. The dataset contains both colored and grayscale photos and consists of 327 sequences from 123 subjects. Each sequence is categorized into one of the seven facial emotions which are (1) $anger$, (2) $disgust$, (3) $fear$, (4) $happiness$, (5) $sadness$, (6) $surprise$, and (7) $contempt$. The number of frames in a sequence varies from 10 to 60 as each subject performs the target facial expression transitions from the neutral phase. The distribution of target classes (facial expressions) in the *CK+* dataset is presented in Figure 2.

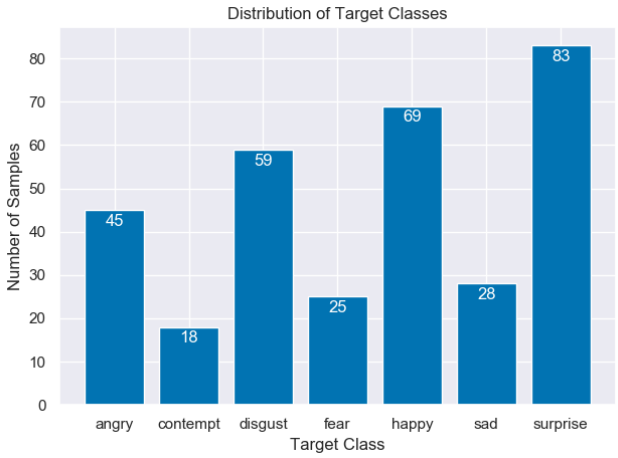

FIGURE 2. The distribution of target classes (facial expressions) in the *CK+* dataset.

B. PROPOSED ARCHITECTURE

The proposed novel deep convolutional neural network architecture, namely, *PyFER*, was consciously designed (1) to efficiently classify given images into six facial expressions, namely (1) $anger$, (2) $disgust$, (3) $fear$, (4) $happiness$, (5) $sadness$, and (6) $surprise$, and (2) to be lightweight in order to make it possible to be used on machines with limited computational power. *PyFER* was implemented using *PyTorch* [28], which is an open-source deep neural network framework written in Python programming language and is based on the *Torch* scientific computing framework [29]. *PyFER* consists of three phases: (1) In the first phase, the images are preprocessed, (2) the second phase contains the traditional CNN modules such as $convolutional$ and $pooling$ layers to perform convolutions over the input, and (3) the third phase is where the classification occurs by virtue of the fully connected layers that make predictions for a given image. Each phase is described in the detail as follows: In the preprocessing phase, the given input images are converted to grayscale form and scaled to $64x64$ pixels in order to reduce the complexity of the data as well as the computational power required. In the second phase, each preprocessed image is processed through the convolutional layers of *PyFER*. Each convolutional layer was succeeded by three components: (1) $ReLU$ activation function that was employed to avoid the vanishing gradient problem as a result of some other activation functions [6] (more details are available in [14]), (2) $Batch\ Normalization$ [30], which is a method that can be applied to any set of activations in the neural network in order to normalize the output of each activation by the mean and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3012703, IEEE Access

standard of deviation of the outputs calculated over the samples in the minibatch [31], (3) *Dropout* [32], which is a technique that deals with the well-known problem of deep neural networks, namely, the *overfitting* problem, as a result of increased depth and complexity of deep neural networks [6]. *Dropout* randomly drops units (neurons) from the neural network during training in order to prevent co-adapting. The datasets, which contain only hundreds of sequences such as the one used within this study, *CK+*, easily overfit since a typical network has many parameters [16]. Therefore, *Dropout* was utilized in each layer. *Max Pooling*, which is a sub-sampling procedure, was utilized to reduce the input size of images by applying the maximum function over the input/pooling window [33]–[36], which not only reduces the required (1) time to train the CNN model and (2) hardware to store available space to store tensors as a natural consequence of the reduced size of the images but also improves the performance [37]. For the *Max Pooling*, both of *kernel size* and *stride* parameters were set to 2. The *kernel size*, *stride*, and *padding* parameters of all convolutional layers were set to 3, 1, and 0, respectively. *CrossEntropyLoss* is used as the loss function (*a.k.a.* criterion) of PyFER which is a combination of *LogSoftMax* and *NLLLoss* functions [38]. *Stochastic Gradient Descent (SGD)* was utilized as the optimization function of *PyFER*, and the *learning rate*, *momentum*, and *weight decay* parameters of *SGD* were set to 0.001, 0.9, and 0.01, respectively. *Flattening* was used as a connector between CNN and Fully Connected layers in order to reshape the input into a vector [39]. All of these hyper-parameters were set following a number of experiments on the validation set of *PyFER*. Finally, in the third phase, the classification of images into facial expressions was applied by virtue of the fully connected layers that mapped all neurons of the previous layer to each neuron of the current one. Each layer of *PyFER* is presented in Figure 3 with the output sizes of the produced images.

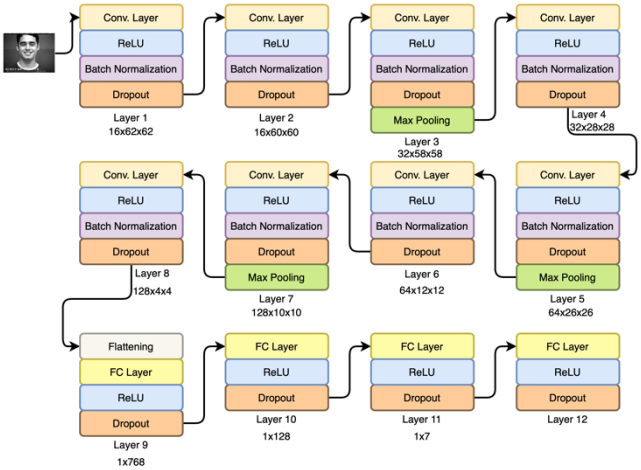

FIGURE 3. The block representation of the proposed convolutional neural network architecture.

The first convolutional layers extract the low-level edge features as shown in the samples of a specific image presented in Figure 4. The deep features are extracted from the deeper convolutional layers as shown in the samples of the image used in Figure 4, which are presented in Figure 5. It can be deduced that the deeper it is, the more abstract the output features are [40].

FIGURE 4. Low-level edge features of a sample image.

FIGURE 5. Deep features of a sample image.

IV. EXPERIMENTAL RESULT AND DISCUSSION

The proposed deep convolutional neural network architecture, *PyFER*, was evaluated by the *CK+* dataset in order to reveal its efficiency. Before training the proposed deep convolutional neural network, all the images in the *CK+* dataset were cropped in order to remove the redundant parts of images such as background thanks to the *OpenCV* face detection module, namely, *opencv-python* [41]. The dataset used to measure the efficacy of *PyFER*, *CK+*, originally contains 593 sequences from 123 subjects. Each sequence starts with a photo of the subject with a neutral facial expression and ends with a photo of the subject with the target facial expression. For each sequence in the *CK+*, there is only one image (the last one in each sequence which is the one that describes the related facial expression the best) that is mapped with a facial expression. Therefore, a Python script was implemented in order to exclude all the images other than the last one, a similar way to that of *Mollahosseini et al.* [6]. Consequently, the pre-processed dataset contained 327 images, and 80%, 10%, and 10% of these images were used for training, validation, and testing purposes, respectively. The images in the dataset were randomly distributed. An overview of the images in a sample sequence from the *CK+* dataset that the subject performs the surprise facial expression is presented in Figure 6.

FIGURE 6. An overview of the images in a sample sequence from the *CK+* dataset that the subject performs the facial expression "*surprise*".

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3012703, IEEE Access

The proposed deep convolutional neural network, *PyFER*, was trained until the point the loss of training was saturated, which occurred around the epoch #200. Then, the training was stopped, and the testing phase was started. The calculated losses for both the validation and training phases are presented in Figure 7.

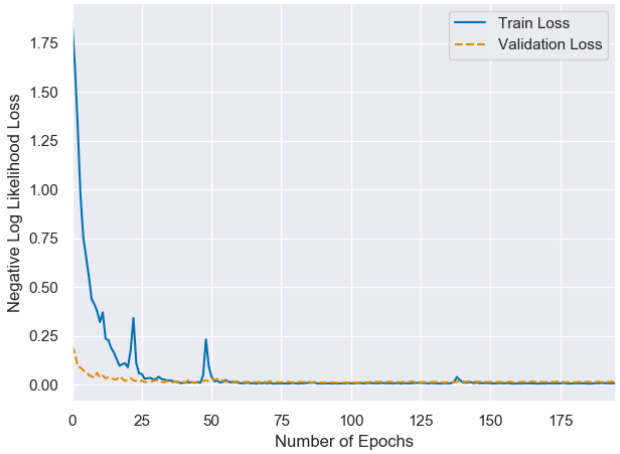

FIGURE 7. The calculated losses for both the validation and training phases.

The confusion matrix is the de-facto standard of measuring the efficiency of classifiers in terms of the proposed system's ability to classify given samples to the classes they actually belong to [42]–[44]. The performance criteria of *PyFER* is defined as follows: (1) The accuracy for detecting facial expressions on given photos, which equals to the ratio of the number of samples correctly classified to the number of all samples, and (2) the fast analysis time since FER applications need a quick response as some of them are used by real-time systems. According to the experimental results, the accuracy of *PyFER* was calculated to be as high as 96.3%. As the confusion matrix of the test result is presented in Figure 8, all facial expressions except for the facial expression "*happiness*" were correctly detected by *PyFER*. 16.67% of images that actually represented the facial expression "*happiness*" were misdetected as the facial expression "*fear*". When the facial expression "*contempt*" included like some of the related work does, the accuracy of *PyFER* was decreased to 85.71%.

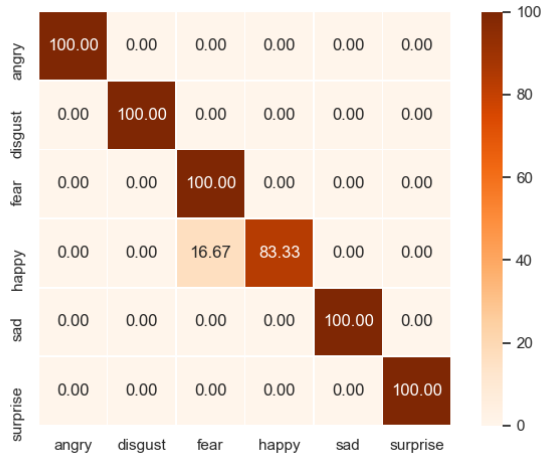

FIGURE 8. The confusion matrix of the test result.

It was experienced that increasing the depth (the number of layers) of the proposed neural network did not improve the accuracy any further, and the hyper-parameters were all set through the conducted experiments. In addition to this, setting other commonly used values to hyper-parameters such as setting the *kernel size* to 1, and *stride* to 0 for each CNN layer also experimented but they were not preferred as both of these experiments decreased the accuracy of the proposed network to 67.86%. Another experiment that was conducted was the arrangement of the layers as some deep neural network models use *Batch Normalization* before the activation function. When *Batch Normalization* used before the activation function (which is *ReLU* for *PyFER*), the accuracy of *PyFER* was decreased to 67.76%. The effect of the activation function on the accuracy also experimented. When *sigmoid* was used as the activation function instead of *ReLU*, the accuracy of *PyFER* was decreased to 77.78%. A comparison of the related work, which was evaluated on the *CK+* dataset, is listed in Table 1. Even though *Jung et al.* [16] had achieved a better accuracy compared to PyFER, our architecture is a lot more lightweight than theirs as their architecture considers features based on temporal appearance and temporal geometry alongside CNN layers. Proposing a lightweight architecture in order to complete the analysis in the tolerable limit to latency for real-time applications is one of the design principles of *PyFER* as this principle not only accelerates training and validation phases but also makes employing the model in lightweight devices (e.g. smartphones) possible. According to the conducted experiments, *PyFER* was able to detect the facial expression of a given photo for an average of 12.8 milliseconds on a 3-years old notebook[2], which is in the tolerable limit to latency for real-time (*a.k.a.* live) applications, which is less than 20 milliseconds [45], [46].

[2] The experiment was carried out on a notebook with the following hardware specifications: Quad-core Intel Core i7-7700HQ CPU 2.80GHz, 32GB memory, 1TB 7200 RPM hard drive.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3012703, IEEE Access

TABLE I
A COMPARISON OF THE RELATED WORK, WHICH WAS EVALUATED ON THE *CK+* DATASET

Related Work	Number of Targeted Facial Expressions	Accuracy (%)
Huang et al. [47]	7	93.89
Kumar et al. [48]	7	95
Shao and Quian [40]	7	92.86, 95.29 (pre-trained)
Poursaberi et al. [49]	6	92.02
Ghimire et al. [50]	6	94.1
Jung et al. [16]	7	97.25
Liu et al. [51]	7	92.05
Mollahosseini et al. [6]	6	93.2 (Top-1)
Lopes et al. [17]	6	91.64
AlexNet [14]	7	92.2
Liu et al. [52]	7	92.4
Zhang et al. [53]	7	95.37
PyFER	**6**	**96.3**

V. CONCLUSION

Through deep learning, handling the problems, which could not be handled by traditional machine learning methods or the ones that would not yield high accuracy even handled by them, becomes possible. Convolutional neural networks, which are a class of deep feedforward neural networks, have yielded remarkable performances in many problems in computer vision in the past years. The FER problem, which is an important research area of Human-Machine Interaction (HMI), is the task of detecting emotions by analyzing facial expressions by virtue of the advances in both computer software and hardware. To this end, a novel CNN architecture, namely, *PyFER*, was proposed in this paper whose efficiency was revealed by virtue of the experiments conducted on a widely-used benchmark dataset. According to the experimental results, the accuracy of *PyFER* was calculated to be as high as 96.3%, and all facial expressions except for the happiness were correctly detected by *PyFER*, which is encouraging for future studies. Also, the experimental result confirms that the average analysis duration of *PyFER* for a given photo is in the tolerable limit to latency for real-time applications.

In future studies, more efficient hand-crafted features can be integrated into the proposed architecture. Also, cross-database training network parameters can be used in order to get better generalization capability. Finally, more facial expressions can be targeted thanks to the above-mentioned features.

REFERENCES

[1] R. Wang and B. Fang, "Affective computing and biometrics based HCI surveillance system," in *2008 International Symposium on Information Science and Engineering (ISISE 2008)*, 2008, pp. 192–195, doi: 10.1109/ISISE.2008.317.

[2] W. Wu, Q. Men, and Y. Wang, "Development of the humanoid head portrait robot system with flexible face and expression," in *Proceedings of 2004 IEEE International Conference on Robotics and Biomimetics (IEEE ROBIO 2004)*, 2004, pp. 757–762.

[3] M. H. Su, C. H. Wu, K. Y. Huang, Q. B. Hong, and H. M. Wang, "Exploring microscopic fluctuation of facial expression for mood disorder classification," in *Proceedings of the 2017 International Conference on Orange Technologies (ICOT 2017)*, 2018, pp. 65–69, doi: 10.1109/ICOT.2017.8336090.

[4] M. B. Mariappan, M. Suk, and B. Prabhakaran, "FaceFetch: A User Emotion Driven Multimedia Content Recommendation System Based on Facial Expression Recognition," in *Proceedings of 2012 IEEE International Symposium on Multimedia (ISM 2012)*, 2012, pp. 84–87, doi: 10.1109/ISM.2012.24.

[5] S. A. Patil and P. J. Deore, "Local Binary Pattern based face recognition system for automotive security," in *2015 International Conference on Signal Processing, Computing and Control (ISPCC)*, 2015, pp. 13–17, doi: 10.1109/ISPCC.2015.7374990.

[6] A. Mollahosseini, D. Chan, and M. 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, pp. 1–10, doi: 10.1109/WACV.2016.7477450.

[7] O. Ekundayo and S. Viriri, "Facial Expression Recognition: A Review of Methods, Performances and Limitations," in *2019 Conference on Information Communications Technology and Society (ICTAS 2019)*, 2019, pp. 1–6, doi: 10.1109/ICTAS.2019.8703619.

[8] C. Mayer, M. Eggers, and B. Radig, "Cross-database evaluation for facial expression recognition," *Pattern Recognit. Image Anal.*, vol. 24, no. 1, pp. 124–132, 2014, doi: 10.1134/S1054661814010106.

[9] A. Ruiz-Garcia, M. Elshaw, A. Altahhan, and V. Palade, "Stacked deep convolutional auto-encoders for emotion recognition from facial expressions," in *Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN 2017)*, 2017, pp. 1586–1593, doi: 10.1109/IJCNN.2017.7966040.

[10] S. E. Kahou *et al.*, "Combining modality specific deep neural networks for emotion recognition in video," in *Proceedings of the 2013 ACM International Conference on Multimodal Interaction (ICMI 2013)*, 2013, pp. 543–550, doi: 10.1145/2522848.2531745.

[11] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops (CVPRW 2010)*, 2010, pp. 94–101, doi: 10.1109/CVPRW.2010.5543262.

[12] T. Kanade, J. F. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," in *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition (FG' 00)*, 2000, pp. 46–53, doi: 10.1109/AFGR.2000.840611.

[13] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1 (NIPS'12)*, 2012, pp. 1097–1105.

[15] T. Zhang, W. Zheng, Z. Cui, Y. Zong, J. Yan, and K. Yan, "A Deep Neural Network-Driven Feature Learning Method for Multi-view Facial Expression Recognition," *IEEE Trans. Multimed.*, vol. 18, no. 12, pp. 2528–2536, 2016, doi: 10.1109/TMM.2016.2598092.

[16] H. Jung, S. Lee, J. Yim, S. Park, and J. Kim, "Joint Fine-Tuning in Deep Neural Networks for Facial Expression Recognition," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV 2015)*, 2015, pp. 2983–2991, doi: 10.1109/ICCV.2015.341.

[17] A. T. Lopes, E. de Aguiar, A. F. De Souza, and T. Oliveira-Santos, "Facial expression recognition with Convolutional Neural Networks: Coping with few data and the training sample order," *Pattern Recognit.*, vol. 61, pp. 610–628, 2017, doi: 10.1016/j.patcog.2016.07.026.

[18] A. Majumder, L. Behera, and V. K. Subramanian, "Automatic Facial Expression Recognition System Using Deep Network-Based Data Fusion," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 103–114, 2018, doi: 10.1109/TCYB.2016.2625419.

[19] D. Hamester, P. Barros, and S. Wermter, "Face Expression Recognition with a 2-channel Convolutional Neural Network," in *Proceedings of the 2015 International Joint Conference on Neural Networks (IJCNN)*, 2015, pp. 1–8, doi: 10.1109/IJCNN.2015.7280539.

[20] S. Alizadeh and A. Fazel, "Convolutional Neural Networks for Facial Expression Recognition," *arXiv Prepr.*, vol. 1704.06756, pp. 1–8, 2017, doi: 10.21629/JSEE.2017.04.18.

[21] P. Burkert, F. Trier, M. Z. Afzal, A. Dengel, and M. Liwicki, "DeXpression: Deep Convolutional Neural Network for Expression Recognition," *arXiv Prepr.*, vol. 1509.05371, pp. 1–8, 2015.

[22] K. Anderson and P. W. McOwan, "A Real-time Automated System for the Recognition of Human Facial Expressions," *IEEE Trans. Syst. Man, Cybern. Part B*, vol. 36, no. 1, pp. 96–105, 2006, doi: 10.1109/TSMCB.2005.854502.

[23] I. Kotsia and I. Pitas, "Facial Expression Recognition in Image Sequences Using Geometric Deformation Features and Support Vector Machines," *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 172–187, 2007, doi: 10.1109/TIP.2006.884954.

[24] C. Shan, S. Gong, and P. W. McOwan, "Robust facial expression recognition using local binary patterns," in *Proceedings of the 2005 International Conference on Image Processing (ICIP 2015)*, 2005, pp. 1–4, doi: 10.1109/ICIP.2005.1530069.

[25] A. Zafer, R. Nawaz, and J. Iqbal, "Face recognition with expression variation via robust NCC," in *ICET 2013 - 2013 IEEE 9th International Conference on Emerging Technologies*, 2013, pp. 1–5, doi: 10.1109/ICET.2013.6743520.

[26] I. Song, H. J. Kim, and P. B. Jeon, "Deep Learning for Real-Time Robust Facial Expression Recognition on a Smartphone," in *Proceedings of 2014 IEEE International Conference on Consumer Electronics (ICCE)*, 2014, pp. 564–567, doi: 10.1109/ICCE.2014.6776135.

[27] J. García-Ramírez, J. A. Olvera-López, I. Olmos-Pineda, and M. Martín-Ortíz, "Mouth and eyebrow segmentation for emotion recognition using interpolated polynomials," *J. Intell. Fuzzy Syst.*, vol. 34, no. 5, pp. 3119–3131, 2018, doi: 10.3233/JIFS-169496.

[28] "PyTorch," 2020. https://pytorch.org (accessed Jul. 06, 2020).

[29] "Torch | Scientific computing for LuaJIT," 2020. http://torch.ch (accessed Jul. 06, 2020).

[30] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *32nd International Conference on Machine Learning (ICML 2015)*, 2015, pp. 448–456.

[31] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," in *Advances in Neural Information Processing Systems 29 (NIPS 2016)*, 2016, pp. 1–9.

[32] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[33] Y. Zheng, B. K. Iwana, and S. Uchida, "Discovering Class-Wise Trends of Max-Pooling in Subspace," in *Proceedings of 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, 2018, pp. 98–103, doi: 10.1109/ICFHR-2018.2018.00026.

[34] Y. Chen, L. Xu, K. Liu, D. Zeng, and J. Zhao, "Event Extraction via Dynamic Multi-Pooling Convolutional Neural Networks," in *Proceedings of the Conference 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL-IJCNLP 2015)*, 2015, pp. 167–176.

[35] M. Sun, Z. Song, X. Jiang, J. Pan, and Y. Pang, "Learning Pooling for Convolutional Neural Network," *Neurocomputing*, vol. 224, pp. 96–104, 2017, doi: 10.1016/J.NEUCOM.2016.10.049.

[36] V. Christlein, L. Spranger, M. Seuret, A. Nicolaou, P. Král, and A. Maier, "Deep Generalized Max Pooling," *arXiv Prepr.*, vol. 1908.05040, pp. 1–7, 2019.

[37] F. Shen, C. Shen, X. Zhou, Y. Yang, and H. T. Shen, "Face image classification by pooling raw features," *Pattern Recognit.*, vol. 54, pp. 94–103, 2016, doi:

10.1016/j.patcog.2016.01.010.

[38] "torch.nn — PyTorch master documentation," 2020. https://pytorch.org/docs/stable/nn.html (accessed Jul. 06, 2020).

[39] K. Du, Y. Deng, R. Wang, T. Zhao, and N. Li, "SAR ATR based on displacement- and rotation-insensitive CNN," *Remote Sens. Lett.*, vol. 7, no. 9, pp. 895–904, 2016, doi: 10.1080/2150704X.2016.1196837.

[40] J. Shao and Y. Qian, "Three convolutional neural network models for facial expression recognition in the wild," *Neurocomputing*, vol. 355, pp. 82–92, 2019, doi: 10.1016/j.neucom.2019.05.005.

[41] O.-P. Heinisuo, "opencv-python," 2020. https://pypi.org/project/opencv-python/ (accessed Jul. 06, 2020).

[42] T. R. Patil and M. S. S. Sherekar, "Performance Analysis of Naive Bayes and J48 Classification Algorithm for Data Classification," *Int. J. Comput. Sci. Appl.*, vol. 6, no. 2, pp. 256–261, 2013.

[43] T. C. W. Landgrebe, P. Paclik, R. P. W. Duin, and A. P. Bradley, "Precision-Recall Operating Characteristic (P-ROC) curves in imprecise environments," in *Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06)*, 2006, pp. 1–5, doi: 10.1109/ICPR.2006.941.

[44] I. Düntsch and G. Gediga, "Confusion Matrices and Rough Set Data Analysis," *J. Phys. Conf. Ser.*, vol. 1229, pp. 1–6, 2019, doi: 10.1088/1742-6596/1229/1/012055.

[45] W. Woszczyk, J. Cooperstock, J. Roston, and W. Martens, "Shake, Rattle, And Roll: Gettiing Immersed In Multisensory, Interactive Music Via Broadband Networks," *AES J. Audio Eng. Soc.*, vol. 53, no. 4, pp. 336–344, 2005.

[46] R. Rowe, "Real time and unreal time: Expression in distributed performance," *J. New Music Res.*, vol. 34, no. 1, pp. 87–95, 2005, doi: 10.1080/1080/09298210500124034.

[47] D. Huang, C. Shan, M. Ardabilian, Y. Wang, and L. Chen, "Local binary patterns and its application to facial image analysis: A survey," *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 41, no. 6, pp. 765–781, 2011, doi: 10.1109/TSMCC.2011.2118750.

[48] P. Kumar, S. L. Happy, and A. Routray, "A Real-time Robust Facial Expression Recognition System using HOG Features," in *Proceedings of 2016 International Conference on Computing, Analytics and Security Trends (CAST)*, 2017, pp. 289–293, doi: 10.1109/CAST.2016.7914982.

[49] A. Poursaberi, H. A. Noubari, M. Gavrilova, and S. N. Yanushkevich, "Gauss-Laguerre wavelet textural feature fusion with geometrical information for facial expression identification," *Eurasip J. Image Video Process.*, vol. 17, pp. 1–13, 2012, doi: 10.1186/1687-5281-2012-17.

[50] D. Ghimire, S. Jeong, J. Lee, and S. H. Park, "Facial Expression Recognition Based on Region Specific Appearance and Geometric Features," in *Proceedings of the 2015 Tenth International Conference on Digital Information Management (ICDIM 2015)*, 2017, pp. 142–147, doi: 10.1007/s11042-016-3418-y.

[51] M. Liu, S. Li, S. Shan, and X. Chen, "AU-aware Deep Networks for facial expression recognition," in *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG 2013)*, 2013, pp. 1–6, doi: 10.1109/FG.2013.6553734.

[52] M. Liu, S. Li, S. Shan, R. Wang, and X. Chen, "Deeply Learning Deformable Facial Action Parts Model for Dynamic Expression Analysis," in *Asian Conference on Computer Vision (ACCV 2014)*, 2015, pp. 143–157, doi: 10.1007/978-3-319-16817-3_10.

[53] C. Zhang, P. Wang, K. Chen, and J. K. Kämäräinen, "Identity-Aware Convolutional Neural Network for Facial Expression Recognition," *J. Syst. Eng. Electron.*, vol. 28, no. 4, pp. 784–792, 2017, doi: 10.21629/JSEE.2017.04.18.

ABDULLAH TALHA KABAKUS received the bachelor's degree in computer engineering from Cankaya University in 2010, the master's degree in computer engineering from Gazi University in 2014, and the philosophy of doctorate degree in Electrical-Electronics & Computer Engineering from Duzce University in 2017, respectively. He is currently working as an Assistant Professor at the Department of Computer Engineering, Faculty of Engineering, Duzce University. His research areas include mobile security, deep learning, and social network analysis. He has been serving as a reviewer for many highly-respected journals.