SINERGI Vol. 23, No. 3, October 2019: 239-244
Accredited **SINTA 2** by KEMENRISTEKDIKTI, Decree No: 10/E/KPT/2019
http://publikasi.mercubuana.ac.id/index.php/sinergi
http://doi.org/10.22441/sinergi.2019.3.008 .

REAL-TIME CLASSIFICATION OF FACIAL EXPRESSIONS USING A PRINCIPAL COMPONENT ANALYSIS AND CONVOLUTIONAL NEURAL NETWORK

Dwi Lydia Zuharah Astuti[1], Samsuryadi[2], Dian Palupi Rini[2]
[1]Graduate Program in Informatics Engineering, Faculty of Computer Science, Universitas Sriwijaya
[2]Department of Informatics Engineering, Faculty of Computer Science, Universitas Sriwijaya
Jl. Srijaya Negara, Bukit Besar, Palembang 30128, Indonesia
Email: dwizohra@ilkom.unsri.ac.id syamsuryadi@unsri.ac.id dian.palupi.rini@gmail.com

Abstract -- *Classification of facial expressions has become an essential part of computer systems and human-computer fast interaction. It is employed in various applications such as digital entertainment, customer service, driver monitoring, and emotional robots. Moreover, it has been studied through several aspects related to the face itself when facial expressions change based on the point of view or perspective. Facial curves such as eyebrows, nose, lips, and mouth will automatically change. Most of the proposed methods have limited frontal Face Expressions Recognition (FER), and their performance decrease when handling non-frontal and multi-view FER cases. This study combined both methods in the classification of facial expressions, namely the Principal Component Analysis (PCA) and Convolutional Neural Network (CNN) methods. The results of this study proved to be more accurate than that of previous studies. The combination of PCA and CNN methods in the Static Facial Expressions in The Wild (SFEW) 2.0 dataset obtained an accuracy amounting to 70.4%; the CNN method alone only obtained an accuracy amounting to 60.9%.*

Keywords: *Classification; Facial Expressions; Image Recognition; Convolutional Neural Network; Principal Component Analysis*

Received: July 1, 2019 *Revised: August 25, 2019* *Accepted: August 28, 2019*

INTRODUCTION

Introduction of facial expressions to the Face Expressions Recognition (FER) has been the topic of recent studies in human-computer interaction. Human facial expressions provide valuable clues about human emotions and behavior. The introduction of facial expressions plays a crucial role in applications such as digital entertainment, customer service, monitoring of drivers, and emotional robots. There have been extensive studies and methods developed (Nariswari, Wirayuda and Dayawati, 2011; Ilyas et al., 2018; Byeon & Kwak, 2014).

PCA (Principal Component Analysis), often referred to as Eigenfaces, is a multifunctional method since Eigenfaces has many functions. Mainly, the function of this field is face recognition, such as predictions, deletions redundancy, data compression, dimensional reduction, and feature extraction (Putra, Dwidasmara and Astawa, 2014). Some previous studies included employing a Fisherface method with the Backpropagation Neural Network approach, where the test data employed the JAFFE Dataset (Abidin, 2011). The face recognition system using the Convolutional Neural Network was implemented against Data Testing The Extended Yale Face Database B (Abhirawan, Jondri and Arifianto, 2017). In another CNN study, applying the Extended Local Binary Pattern as a texture classification was able to overcome the effect of light intensity on the image. So, the image affected by light intensity could produce feature pattern extraction that was almost the same as the image with low illumination. Therefore, the configuration weighting initialization parameters using standard spreads that could speed up convergence and stability rather than randomly initialized (Navas z et al., 2018; Chen et al., 2017; Zufar, 2016).

Most of the performance of the proposed methods decreased when handling non-frontal and multi-view FER cases (Liu et al., 2018). The paper proposes a Principal Component Analysis (PCA) method and a Convolutional Neural Network (CNN) method to classify facial expressions in multi-view and unrestricted environments. Based on the matters as mentioned earlier, real-time facial expression classifiers were examined through utilizing the

PCA method as feature extraction and facial expression classifier using (CNN). CNN was a deep learning method that proved very efficient in the image classification used to carry out the learning process on a computer to find the best representation. Therefore, it is expected that it could obtain accurate facial expression classifiers. If the lighting was less or more adequate using specific recognition of facial expressions or facial positions, this study combined PCA with CNN for classification of facial expressions in real-time using the SFEW 2.0 dataset with an accuracy amounting to 70,4%. The results are compared with previous work based on (Liu et al., 2018) study.

METHOD

The process in the classification of human facial expressions consists of three stages, namely face image detection, feature extraction, and classification of facial expressions. In this study, the face detection process used a Viola-Jones method. Viola-Jones method is a face detection method that provides face detection results with high accuracy (Syafira & Ariyanto, 2017). Fig. 1 shows the flow of the Viola-Jones method in detecting faces.

Figure 1. Process Flow of the Viola-Jones Method (Fitriyah, 2015)

Input data was in the form of images that had facial objects and frontal face positions, using haar features as object detectors and feature capture. Then, by using an integral image, we determined the presence or absence of hundreds of haars in an image through the Adaboost algorithm, used to select essential features and to practice classification. Features that had the most significant restrictions between the objects and non-objects were deemed to be the best features. The next step was the cascade classifier. A method employed to combine complex classifiers in a multilevel structure that could increase the speed of object detection by focusing on the area of the image had a chance. Fig. 2 showed the structure of the cascade classifier.

After the image was pre-processed, the face image was extracted to get the essential features of the image. Feature extraction was a process of taking characteristics found in the object in the image. In this study, feature extraction employed the Principal Component Analysis method and a new approach in the feature extraction process, namely Feature Learning and Convolutional Neural Network.

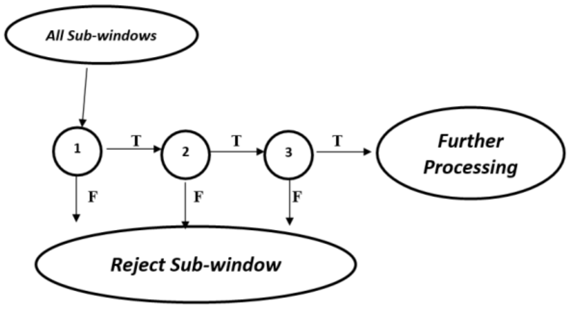

Figure 2. Structure of the Cascade Classifier (Fitriyah, 2015)

PCA is a way of identifying patterns in the data, and then the data is extracted based on their similarities and differences. Since it is difficult to find patterns in the data that have large dimensions where large graphic images are not sufficient, PCA is a powerful method to analyze that data (Zhou et al., 2013).

We looked for an Eigenface value (using the PCA method) that was a significant feature and a principal component in the face collection in the database. Eigenface was obtained from the Eigenvector covariance matrix from the set of images in the database. This Eigenvector was a feature describing variations among the facial images. The stages of taking the features with this method were calculating the average value of the image, calculating the image covariance matrix, calculating the eigenvalue and eigenvector PCA, sorting the eigenvalue from the largest to small and eliminating the small eigenvalue, and determining the eigenface value to be taken.

Calculate the average value of the image:

$$\mu = \frac{1}{N}\sum_{k=1}^{N} x_k \quad (1)$$

Calculating the covariance matrix image:

$$C = \sum_{k=1}^{n} (x_k - \mu)\,(x_k - \mu)^T \quad (2)$$

Calculate the Eigenvalue and the Eigenvector:

$$Cu_n = \lambda_n u_n \quad (3)$$

where:

u = *eigenvector*

λ = *eigenvalue*

Sort the eigenvalues from the largest to the smallest and eliminate the small eigenvalues, specifies the eigenface value to be retrieved:

$$\frac{\sum_{i=1}^{M}\mu_i}{\sum_{i=1}^{M}\mu_k} = A \quad (4)$$

The classification process was the process of grouping objects into the Convolutional Neural Network method, precisely at the last layer of the Convolutional Neural Network, namely the fully connected layer in the appropriate class.

In the CNN method, the data was transmitted to a network, so it became two-dimensional data that could produce linear operations and the weighting parameters on CNN that were different. In the CNN linear operation method using convolution operations, although the weight was not one dimensional in size, it would transform into four dimensions that were a set of convolution kernels as Fig. 3 showed us. The dimensions of the weight on CNN were:

Neuron input x *Neuron output* x *Height* x *Width* (5)

Due to the nature of convolution, CNN could only be used in data sets that had two-dimensional structures such as image and sound.

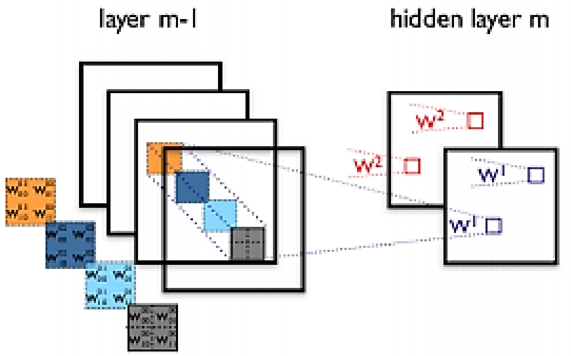

Figure 3. Convolution Process on CNN (Ian, Yoshua, & Courville, 2014)

If we used a two-dimensional image as our input, we might also want to use a two-dimensional *K* kernel:

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n) \quad (6)$$

Convolution was commutative, meaning that we could write equally:

$$S(i,j) = (K * I)(i,j) = \sum_m \sum_n I(i-m, j-n)K(m,n) \quad (7)$$

Fig. 4 showed a 2-D convolution without flipping the kernel. We limited the output only to the position where the kernel was located entirely in the image, called "valid" convolution in some contexts. We drew a box with an arrow to show how the upper left element of the output tensor was formed through applying the kernel to the top left area corresponding to the input tensor.

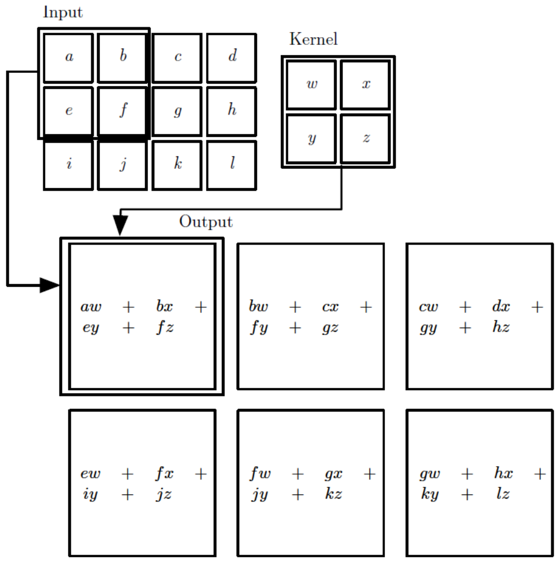

Figure 4. A 2-D convolution without flipping the kernel (Ian, Yoshua, & Courville, 2014)

In the training process, the model would be trained using one piece of data that had been shared with the k-fold cross-validation technique. Then, the model was tested using one-part test data, and each test measured its accuracy value. If the dataset that was used were unbalanced, each test would also measure the value of precision, recall, and f-measure. Fig. 5 shows the test flow process.

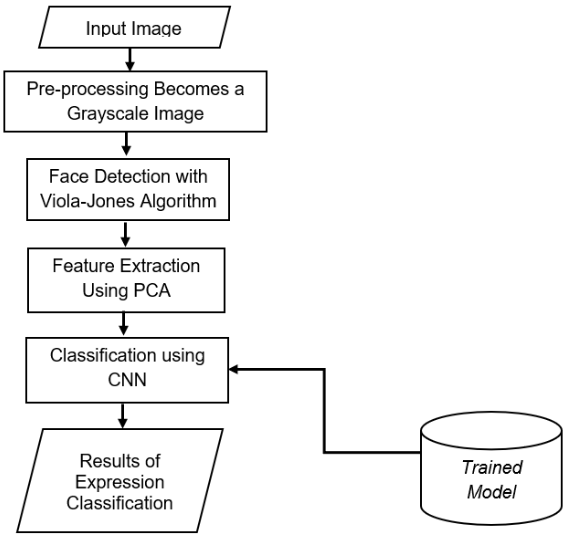

Figure 5. Research Testing Stages

RESULTS AND DISCUSSION

The method that was used had been implemented in the dataset of images taken through Static Facial Expressions in The Wild (SFEW) 2.0 containing 1073 images with six different expressions; anger, disgust, fear, happiness, sadness, and surprise (Dhall et al., 2012).

In this study, testing and calculation of accuracy used a confusion matrix, as Table 1 showed us.

Table 1. Confusion matrix facial expression from the SFEW 2.0 dataset

Confusion Matrix:

Predicted / Actual	Angry	Disgust	Fear	Happy	Sadness	Surprise
Angry	**0.72**	0	0	0.04	0.16	0
Disgust	0.13	**0**	0	0.174	0.565	0
Fear	0.143	0	**0.571**	0	0.286	0
Happy	0	0	0	**0.878**	0.041	0
Sadness	0	0	0	0.033	**0.9**	0
Surprise	0	0	0	0	0.143	**0.571**

Table 1 showed the calculation of the accuracy of the confusion matrix in the six facial expressions of the SFEW 2.0 dataset, where the expression of disgust was obtained by 0% accuracy, due to the expression generated and variations in expression. The highest accuracy was obtained at happy expressions amounting to 43% in Fig. 6 with the blackest colored box. The following was a confusion matrix graph from Table 1 where the predicted scale was limited to only 45%.

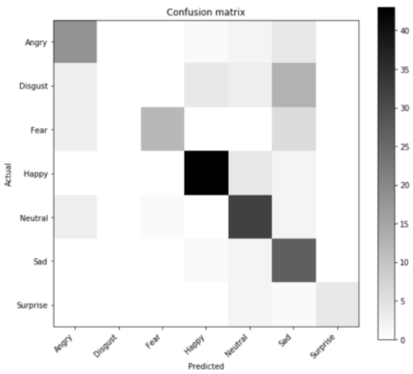

Figure 6. Graph of confusion matrix facial expressions

Table 2 showed the results of previous studies, with an average of 60.9-percent. Accuracy. This study used the SFEW 2.0 dataset, a feature classifier method with extracts robust in-depth salient features of saliency-guided facial patches on CNN, as listed in Table 2.

Table 2. Confusion matrix facial expressions from the LFW dataset

Confusion Matrix:

Predicted / Actual	Angry	Disgust	Fear	Happy	Sadness	Surprise
Angry	**0.559**	0.202	0.166	0	0.009	0.006
Disgust	0.137	**0.505**	0.087	0.02	0.111	0.139
Fear	0.055	0.106	**0.595**	0.016	0.184	0.004
Happy	0.002	0.16	0.01	**0.852**	0.004	0.012
Sadness	0.146	0.142	0.115	0	**0.572**	0.025
Surprise	0.019	0.121	0.156	0.013	0.036	**0.655**

Source: (Liu et al., 2018)

Fig. 7 showed a graph of increasing accuracy in the training process and SFEW 2.0 validation/test dataset with an accuracy amounting to up to 75% at epoch 160. Fig. 8 showed the process of a decreasing loss in the SFEW 2.0 training process and validation/test dataset, where at each epoch, the loss had decreased until epoch 160, which had the lowest loss (error) value amounting to 0,8 shown in the decreasing graph movement. This fact proved that the method used in this study could reduce the level of loss in the training data.

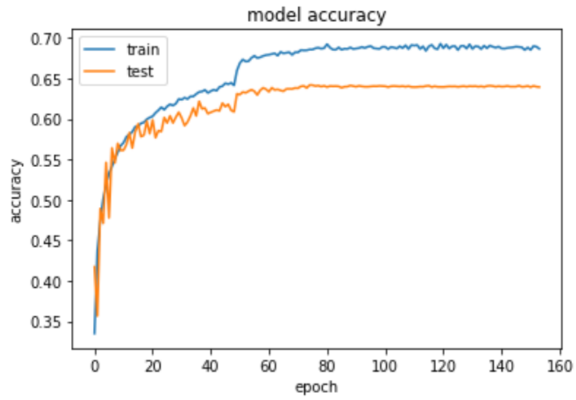

Figure 7. SFEW 2.0 accuracy dataset model

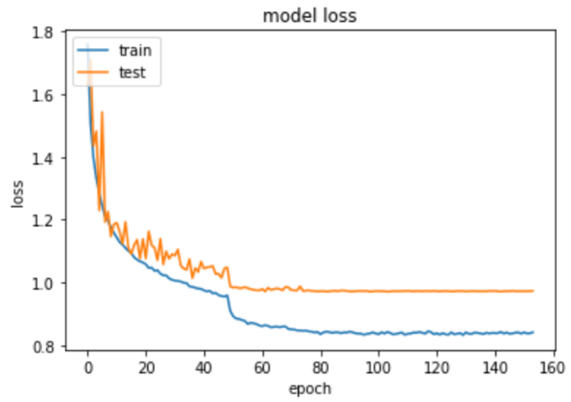

Figure 8. SFEW 2.0 loss dataset model

Fig. 9 was a display of interface application classification of facial expressions in real-time by PCA and CNN methods, where this interface would detect any facial objects in real-time and would classify them.

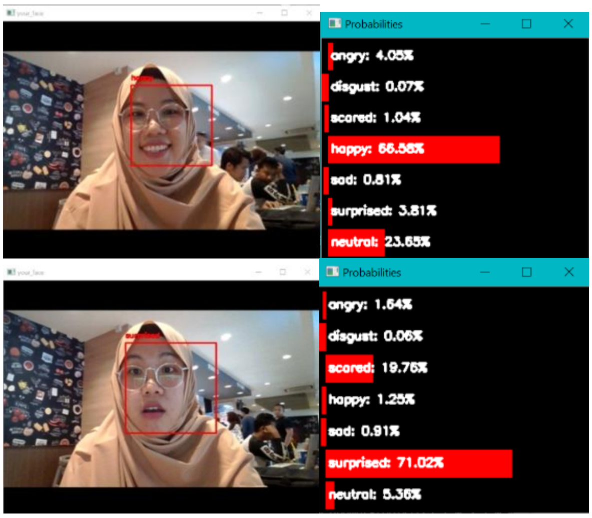

Figure 9. Main Interface Application of Classifying Facial Expressions in Real Time that Shows Happy and Surprised Expressions

CONCLUSION

The combined performance of Principal Component Analysis (PCA) and Convolutional Neural Network (CNN) resulted in a 9,5-% higher accuracy than that of the previous studies. PCA served as feature extraction and feature selection that could improve the performance of CNN in classifying facial expressions that resulted in better performance than that of merely using CNN alone.

We suggest that we add some image preprocessing methods and focus more on the face detection algorithms that will take an important part in the image. In this study, the face detection methods employed haar; the cascade was good for the real-time applications, yet it could not focus on the face image detection. It was difficult for us to distinguish between the classification of anger expressions from that of disgust expressions since the datasets that were used were almost the same. This study still lacked the performance of the haar cascade method as a face detection since this method could only detect anything other than facial images.

ACKNOWLEDGMENT

This study was supported/partially supported by Static Facial Expressions in the Wild (SFEW) 2.0. We would like to express our appreciation to our colleagues from the Faculty of Computer Science, Universitas Sriwijaya, who provided us with their insight and expertise that greatly assisted this study despite the fact that they might not agree with all of the interpretations /conclusions of this study.

REFERENCES

Abhirawan, H., Jondri, & Arifianto, A. (2017). Pengenalan Wajah Menggunakan Convolutional Neural Networks (CNN). *Universitas Telkom*, *4*(3), 4907-4916.

Abidin, Z. (2011). Pengembangan Sistem Pengenalan Ekspresi Wajah menggunakan Jaringan Syaraf Tiruan Backpropagation (Studi Kasus pada Database MUG). *Jurnal Matematika Murni Dan Terapan*, *5*(1), 21–30.

Byeon, Y.H. & Kwak, K. C. (2014). Facial Expression Recognition using 3D Convolutional Neural Networl. International *Journal of Advanced Computer Science and Application*, *5*(12), 107-112.

Chen, X., Yang, X., Wang, M. and Zou, J. (2017). Convolution neural network for automatic facial expression recognition, *2017 International Conference on Applied System Innovation (ICASI)*, Sapporo, Japan. (pp. 814-817) http://doi.org/10.1109/ICASI.2017.7988558

Dhall, A., Goecke, R., Lucey, S., & Gedeon, T. (2012). Collecting Large, Richly Annotated Facial-Expression Databases from Movies. *IEEE MultiMedia*, *19*(3), 34–41. http://doi.org/10.1109/MMUL.2012.26

Fitriyah, N. (2015). Analisis dan simulasi sistem pengenalan wajah dengan metode fisherface berbasis outdoorvideo. *Prosiding Seminar Nasional Universitas PGRI Yogyakarta 2015*. Yogyakarta, Indonesia. (pp. 433-438)

Ian, G., Yoshua, B., & Courville, C. (2014). *Deep Learning*. MIT Press.

Ilyas, C. M. A., Haque, M. A., Rehm, M., Nasrollahi, K., & Moeslund, T. B. (2018). Facial Expression Recognition for Traumatic Brain Injured Patients, Proceeding *in International Conference on Computer Vision Theory and Applications*. Madeira, Portugal. (pp. 522–530). https://doi.org/10.5220/0006721305220530

Liu, Y., Yuan, X., Gong, X., Xie, Z., Fang, F., & Luo, Z. (2018). Conditional convolution neural network enhanced random forest for facial expression recognition. *Pattern Recognition*, *84*, 251–261. https://doi.org/10.1016/j.patcog.2018.07.016

Nariswari, N., Wirayuda, T. A. B., & Dayawati, R. N. (2011). Pengenalan Ekspresi Wajah menggunakan Kombinasi Gabor Wavelet, Adaptive Boosting Feature Selection, dan Support Vector Machine. *Thesis*. Telkom University.

Navaz, A.S.S., Sri, T.D. and Mazumder, P. (2018). Face Recognition using Principal Components Analysis and Neural Network. *International Journal of Computer Networking. Wireless and Mobile Communication, 3*(1), 245-256.

Putra, I. N. T. A., Dwidasmara, I. B. G., & Astawa, I. G. S. (2014). Perancangan Dan Pengembangan Sistem Absensi Realtime Melalui Metode Pengenalan Wajah. *Jurnal Sains Dan Teknologi*, *3*(2), 450-467.

Syafira, A. R., & Ariyanto, G. (2017). Sistem Deteksi Wajah Dengan Modifikasi Metode Viola Jones. *Jurnal Emitor*, *17*(01), 26-33.

Zhou, C., Wang, L., Zhang, Q., & Wei, X. (2013). Face recognition based on PCA image reconstruction and LDA. *Optik*, *124*(22), 5599–5603. https://doi.org/10.1016/j.ijleo.2013.04.108

Zufar, M. & B. S. (2016). Convolutional Neural Networks untuk Pengenalan Wajah Secara Real-Time. *Jurnal Sains Dan Seni*, *5*(3), 1–6.