Blurred Facial Expression Recognition System by Using Convolution Neural Network

Elaf J. Al Taee
Department of Computer Science, Faculty of Education, University of Kufa, Najaf, Iraq.
E-mail: elafj.altaee@uokufa.edu.iq

Qasim Mohammed Jasim
College of Dentistry, University of Alkafeel, Najaf, Iraq.
E-mail: qasim.alhassani@alkafeel.edu.iq

Received August 02, 2020; Accepted October 03, 2020
ISSN: 1735-188X
DOI: 10.14704/WEB/V17I2/WEB17068

Abstract

A facial expression is a visual impression of a person's situations, emotions, cognitive activity, personality, intention and psychopathology, it has an active and vital role in the exchange of information and communication between people. In machines and robots which dedicated to communication with humans, the facial expressions recognition play an important and vital role in communication and reading of what is the person implies, especially in the field of health. For that the research in this field leads to development in communication with the robot. This topic has been discussed extensively, and with the progress of deep learning and use Convolution Neural Network CNN in image processing which widely proved efficiency, led to use CNN in the recognition of facial expressions. Automatic system for Facial Expression Recognition FER require to perform detection and location of faces in a cluttered scene, feature extraction, and classification. In this research, the CNN used for perform the process of FER. The target is to label each image of facial into one of the seven facial emotion categories considered in the JAFFE database. JAFFE facial expression database with seven facial expression labels as sad, happy, fear, surprise, anger, disgust, and natural are used in this research. We trained CNN with different depths using gray-scale images from the JAFFE database.The accuracy of proposed system was 100%.

Keywords

Facial Expression Recognition, Deep Learning, Convolutional Neural Network, JAFFE Database, Face Detection, Rectified Linear Units.

Introduction

Recently, there is an urgent need for computer vision, which enters in many applications in human life. One of the branches of this field in computer science is the recognition of facial expressions [1], some emotional states show their effects on the face so it is possible to understand and know what the person feels from facial expressions [4]. Human facial expressions can be easily classified into 7 common emotions: happy, surprise, sad, fear, anger, disgust, and natural [3]. Research in Automatic Facial Expression Recognition (FER) is very imperative nowadays due to its wide collection of applications such as robotics, digital signs, mobile applications, psychiatry, criminal interrogations, human-computer interaction and medicine [1-4]. For people, the facial expressions are incredibly easy to understand and what they mean for each other, this is not the same for the computer or the machine, it is still a challenge to understanding the human emotional with high accuracy.

To complete the recognition process of facial expression, the image of the face must be identified first, and this acts as preprocessing to input image [6], then we can choose techniques of two approaches, either based on features or Model Template-Based approaches [7]. For the features approach, there are two types; [8], geometric features which is often sensitive to noise so that it needs accurate and precise detection and tracking methods. And appearance features, which is difficult to generalize across different persons [3].

After the selection of suitable features we need to decide which any emotion state is represent from the seven emotional states, and this do by classify this features into one emotion state [12]. There are more than one technique for classification, (SVM) which mean Support Vector Machine used in [10] to classification phase, while Mahesh et al in [11] use feed forward neural networks for classification.

Caifeng et al in [26] for feature extraction used Local Binary Patterns and to classify these features they applied template matching and Support Vector Machine. The development in deep learning lead to the appearance of a new method for classification and in the same time the feature extraction phase is dispensed. Yan Lecun in 1989 proposed Convolution Neural Networks (CNN) in which the learning happen in multiple levels (deep learning) as in conventional neural networks. CNN consist of two phases, in the first one, the image is convoluted with filters. And full connected neural networks phase which produce the classes at end, and this was used in a variety of applications like as image segmentation, face recognition etc. [13][14].

Based on convolutional neural networks, an Automatic Facial Expression Recognition system is develop in this work. An image is inserted into our system; then using the Viola-Jones algorithm for located the faces in the image after that using set of CNN to predict the facial expression label which should be one of the 7 labels, the JAFFE database is used to train and test the classifier.

Related Work

Nhan et al in [6] for extract data from the face they use local binary pattern and before perform LBP they dividing face images into non overlap square regions, then they used SVM for LBP features classification.

The most important facial regions that effects the facial expression recognition are eyebrows, eyes, lips, and nose, Jinglian et al in [5] find the contours of this regions and they used Active Appearance Models (AAMs) [25] in order to localized a sets of landmark points. Find features called Facial Animation Parameters (FAPs) by analysis the information of this regions by shapes, and they use Principal Component Analysis (PCA) in order to analyze the discriminative capabilities of these FAP groups (eyes, eyebrows, mouth and nose). Then they used Hidden Markov Model (HMM) for classification.

Augustine and Kamil in [12] performed FER based on Local Binary Patterns for feature extraction after preprocessing input image, for classifying this features they applied the distance classifier between the features of input image and images database.
Dinh Viet Sang et al in [9] used Convolutional Neural Networks to perform FER with input image 42x42x1, the output layer consists of 7 neurons according to 7 emotional labels. Different loss functions associated with supervised learning were applied in order to learn CNN in addition to several training tricks and they stated that the multiclass SVM loss is better than cross-entropy loss in the process of facial expression recognition.

Materials and Methods

The proposed work to distinguish the emotions of human based on expressions of the facial depend on CNN. The database which is used for training the proposed CNN is (JAFFE) database. We need to identify the face inside the image before the application of CNN, so in the next will explain the technique that used to face detection in the image and explain CNN and its parts, figure 1 show the proposed system.

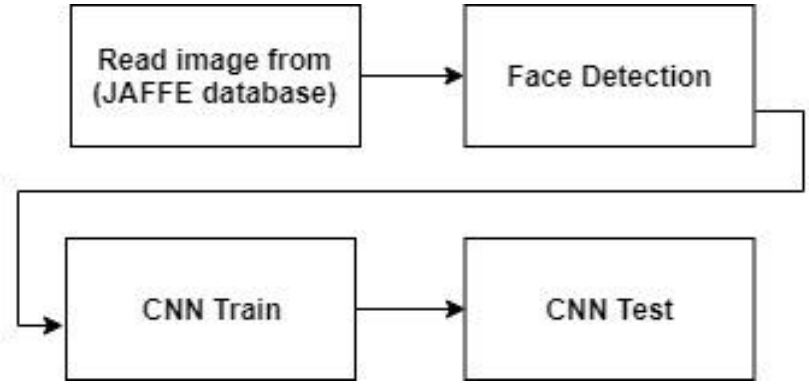

Figure 1 The proposed system

1) Read Image

The images used in this proposed work have 256x256 resolution from The JAFFE database which contains about 213 images, see figure 2. All images are grayscale in this database with (tiff) file format, and each image has inscription demonstrate the emotional expression.

Figure 2 The emotional expression images in JAFFE database

2) Face Detection

Detect the face inside the image is done by using image processing techniques that tell that the image contains a human face or not and,if there is one, located it exactly [15] see figure 3. In our work, we will use Viola-Jones algorithm [17] for Face Detection (FD). This algorithm based on three basic ideas Image Integral, classifier (AdaBoost) and Cascade Structure, and will explain these ideas in the next sections.

(a) Facial Expression (b) Face Detection

Figure 3 Face detection

- **Image Integral**

At beginning we need to extract the feature from input image and produce the integral image based on Haar functions. Three types are used two-rectangle, three-rectangle and four-rectangle features [17] as shown in figure 4. These three types of Haar features can be done by finding the sum of the all pixels inside it and then find difference between these two summation, this for the first one. For the second one, computes the summation within center rectangle and also for the two outside rectangles of the center, will applied the summation, then subtracted from the center outside and find the difference between diagonal pairs of rectangles for the third one.

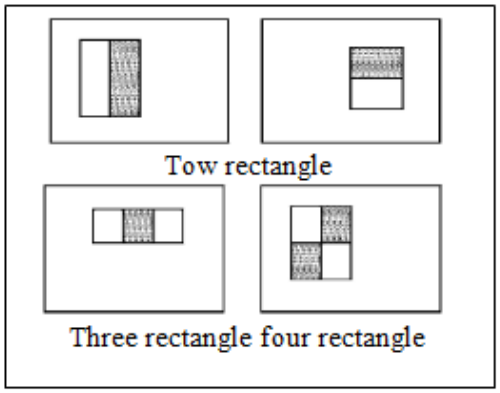

Figure 4 Three haar features

In rectangle feature, there is another type of image feature representation which is called Integral Image, for any pixel (x, y) of the original image can compute the equivalent integral image by the sum of all pixels which be above the pixel of (x, y) and to the left of it, and the formula of the image integral is:

$$ii(\mathrm{x}, \mathrm{y}) = \sum_{x' \leq x, y' \leq y} i(x', y') \ldots\ldots (1)$$

Where $i(x, y)$ equal to The pixel value in the source image, and $ii(x, y)$ is the value of image integral.

1	1	1
1	1	1
1	1	1

Input image

1	2	3
2	4	6
3	6	9

Integral image

Figure 5 Example for Integral pixels

- **Adaboost Algorithm**

In the Viola Jones algorithm, Adaboost algorithm was used for trained and classify Haar-like rectangular feature. Adaboost algorithm tried to increase the accuracy of the weak classifiers (any learning algorithm) by combining the classifications of many weak classifiers to produce accurate classification. In the weak classifier, error rate is slightly better than random guessing [24]. Adaboost takes $(x1, y1)\ldots(xn, yn)$ training set as input where each xi is one of (X) instance space, and each label yi is belong to label set Y.

In this algorithm, the weak learning algorithm is repeatedly evoked in a sequence of rounds t=1.....T. AdaBoosting aims to maintain a distribution weights over the training set and this act as the main ideas of this algorithm. For training example i on round t, the weight of this distribution is denoted by $Dt(i)$. At first, weights are set equally, while the weights of incorrectly classified examples are increased on each round. Because of that, the weak learner will be forced to focus on the hard examples in the training set [27].

- **Cascade Structure**

The goal of cascade is to increase the performance of detection process, this goal can be achieved by constructing boosted classifiers. So, alot of those negative sub-windows will be dismissed, while the all positive cases will be at almost detected (rounded the false negative rate to zero by adjusting the threshold of a boosted classifier)[28]. Most of sub windows with simpler classifiers are rejected before calling the much complex classifiers to achieve low false positive rates see figure 6.

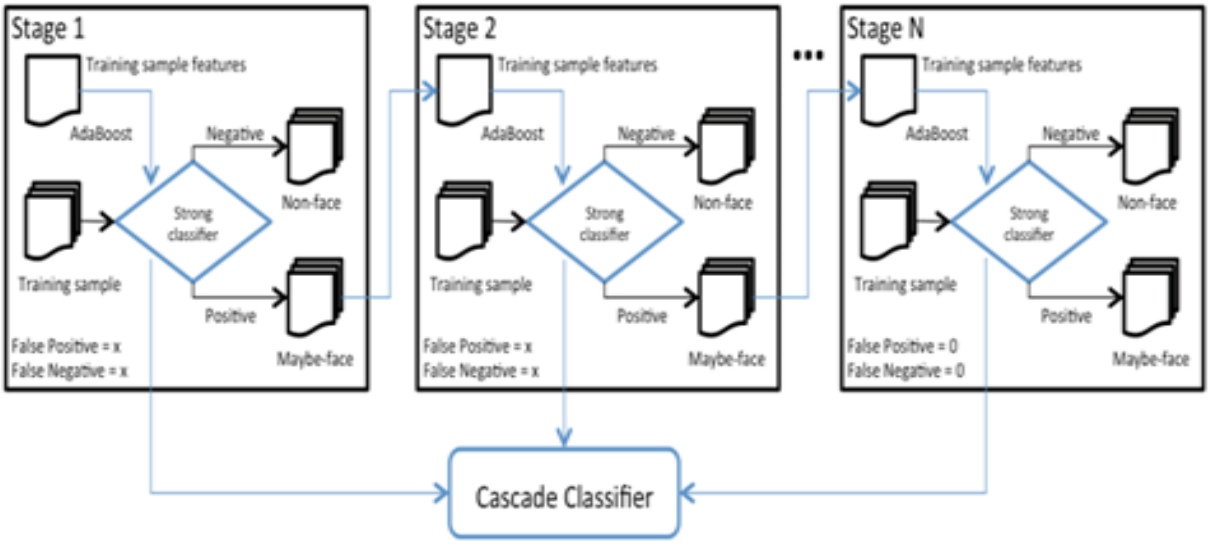

Figure 6 Cascade work flow

3) CNN Training

Convolution Neural Network is one of the best important methods in deep learning that mimics the human system of vision. CNN is employed in many applications, some of these important applications are image segmentation and image recognition. Generally, CNN made of two major parts, the first one consists of a convolution layer, Rectified Linear Units layer and pooling layer, the second part is a fully connected layers [18]. In the following will introduce these parts.

A) Convolution Layer

The key factor behind successful classification system is the adequate features with which the correct class can be produced. The convolution layer in CNN is the appropriate choice for extract the salient features from input image by convolute the pixels of input image with several trainable filters. Here, the same filter will pass on all pixels in the image and this leads to a reduction in the required parameters compared to the traditional neural network [21].

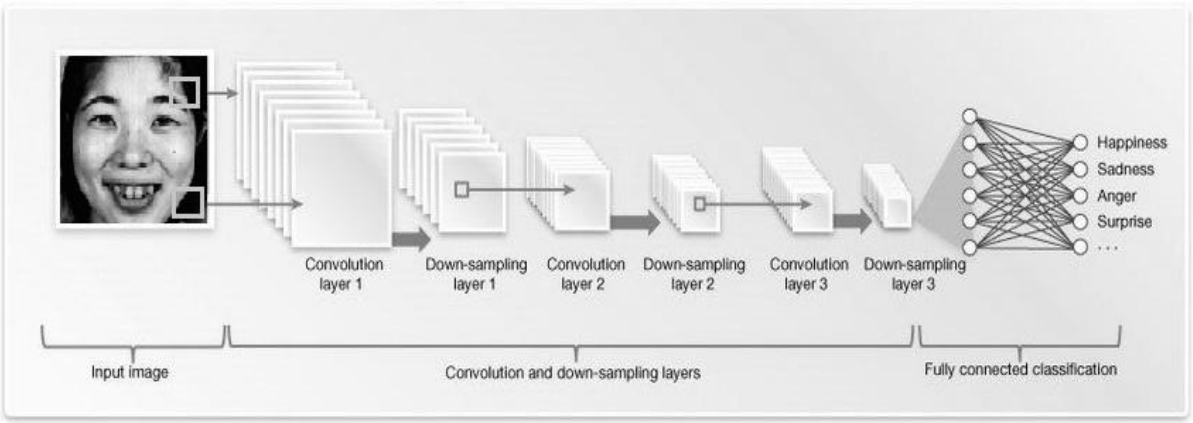

Figure 7 Architecture of proposed CNN

To explain, if we had image of 255 * 255, we would need a matrix of weights consisting of 65025 parameters. In the case of convolution, the number will decrease, if we had the same size as the previous image and we had a five filters each one consisting of 9 * 9 parameters, there will be 405 parameters for whole pixels of input image. This filter will be passed on the image as a window, and this window will be transferred at every turn by one step or more called a Stride (S). Sometimes, needs to preserve the size of the output after convolution of the input with the kernel of filter or to adjust the size of output as we need. This is done by adding zeros around the original image, and these zeros are called zero padding (P).

The output from the convolution layer, called Feature Map, for this input image if there 2 to stride and 0 to padding, it will 123*123*5 according to

(wi-wf+2*p)/s+1 (2)

(hi-hf+2*p)/s+1 (3)

And if we use 11 filters in the second convolution layer each one with 9*9 that means 891 parameters for whole 123*123*5, after apply equations 2&3 the output of this convolution layer will be 57*57*11.

Each element of features map can be obtained by convolute the elements of, input/feature map of the prior layer, with the elements of the kernel by using the equation 4.

$$Z_{i,j}^{(l,k)} = \theta\left(\sum_{e=0}^{K_h}\sum_{r=0}^{K_w} W_{(e,r)}^{k} * X_{(i+e,j+r)}^{l-1} + b^{(l,k)}\right) (4)$$

Where $Z_{i,j}^{(l,k)}$ the ith,jth feature map in the *l* layer after apply *k* kernel which between *l-1* and *l* layers which represent current layer, θ ( ) the activation function, K_h the height of the k_{th} kernel and k_{w} the width of this kernel, $X_{(i+e,j+r)}^{l-1}$ the *(i+e, j+c)* element of the X_{th} features map in the prior layer and $b^{(l,k)}$ the base from the current layer.

B) Rectified Linear Units

Rectified Linear Units ReLU in the CNN are represented as an activation function [21], which exists in every layer in the neural network and for each cell. There are several functions that are used for this purpose, including linear and nonlinear functions. The simplest and most widespread one used in the CNN is (ReLU), given by:

$$\theta(x) = \max(x) \quad (5)$$

Some researchers state that ReLU is a layer [22] and others consider it as part of convolution layer, also some use it after last convolution layer, while others use it after each convolution layer.

C) Pooling Layer

Since the convolution layers are continued and the number of features mapped in the current layer is the same as the number of filters in the prior layer, Although the parameter number is not equal to the number of filter parameters, it depend on the number of stride, whatever the element of features map are too much and therefor will be there a lot of mathematical operations so this is why we need to reduce the parameters and this is what the pooling layer does. The most popular methods for down sampling are Max Pooling, Average Pooling and Sum Pooling.

D) Full Connected

After Preparing the features from the inputs using the convolution layers which consider as high level of features, it became ready to classify it. Full connected multi-layer perceptron neural network used for this purpose in which each neuron from previous layer connected to all neurons in the next layer. It takes features from the last layer of the convolution as a one dimension matrix and the output for each one of neurons given by:

$$O(n_i) = \theta(\mathrm{Wi} * \mathrm{X} + \mathrm{b}) \quad (6)$$

Where the $O(n_i)$ the output of neuron i, Wi the vector of weight for neuron i, X the vector of input for this neuron, b the base and $\theta(x)$ the activation function.

In the proposed system we design a CNN architecture with 11 filters and three convolution layers as illustrated in Fig (7). The first convolution layer with kernel size 20 * 8, stride 1 and pad 1 is applied. Next, Rectified Linear Units (ReLU) layer is applied. Next a max pooling layer with kernel size 2 * 2, stride 2 and pad 1 is applied. This process is repeated 3 times with different strides and pads and kernels size. Finally a fully connected and Softmax layers are added to the network. This model is evaluated on images from JAFFE database. The evaluation seven facial expressions of various images from the JAFFE database are used for training.

4) CNN Test

The CNN training process aim to train the filter's characteristics to extract the most important features from input images which decide the class, this training process apply by use the JAFFE database. These characteristics are used as filters in the a test mechanism for classify a given input image to one of the available classes. The training process is carried out from still images of subjects displaying emotions as shows in Fig. 7

Results and Discussions

1. Convolutional Neural Network Architecture of proposed system has been trained and tested on JAFFE database. This database includes images for emotional facial expression of 10 female Japanese (natural face, in addition to six emotion expression), as shows in Fig. 2.
2. The 256 x 256 pixel grayscale images are compressed into 150 x 150 pixels, this procedure we need to reduce the parameters that will be included in the network training process. A total of 182 images were used to train our proposed CNN and for testing use 31 images, these images were categorized into seven emotional labels. See Table 1 for the results.

Table 1 Samples of the proposed system results

No.	CNN Train			CNN Test			Recognition rate (%)
	Read image from (JAFFE database)	Face Detection	Actual Facial Expression	Read image (De-Blur / Blur)	Face Detection (De-Blur / Blur)	Ideal Facial expression	
1			Anger	 	 	Anger	100
2			Disgust	 	 	Disgust	100
3			Fear	 	 	Fear	100
4			Happy	 	 	Happy	100
5			Neutral	 	 	Neutral	100
6			Sad	 	 	Sad	100
7			Surprise	 		Surprise	100

3. Recognition rate is the basic measure used in evaluating search strategies and was computed as follows:

$$\text{Recognition Rate} = \frac{Number\ of\ image\ correctly\ rocognized}{Total\ image} * 100\%$$

After applying the proposed system upon our selected database, the Recognition rate for facial expression images is 100% as shown in Table 1.

4. The proposed system shows better results than others when compared to other similar systems as illustrated in table 2.

Table 2 Comparing the performance of proposed system with other systems

Papers	Recognition methods	Recognition rate (%)
[17]	local features and machine learning	89.48
[18]	No-Reference Blur Metric	79.90
[19]	Adaptive Feature Extraction	68.94
[20]	BIEFR	82
The proposed system	Convolutional Neural Network	100

Conclusion

In this paper we proposed framework for automatic facial expression recognition based on the convolution neural network architecture, i.e. neutral, surprise, sad, happy, rage, fear and disgust. The proposed system has been tested on JAFFE database and give excellent results. The Recognition rate was 100% for facial expression. Also the system performance was compared with other works and gives better results than the other algorithms as shown in the table 2.

References

Chen, X., Yang, X., Wang, M., & Zou, J. (2017). Convolution neural network for automatic facial expression recognition. *In IEEE International conference on applied system innovation (ICASI)*, 814-817.

Bargshady, G., Soar, J., Zhou, X., Deo, R.C., Whittaker, F., & Wang, H. (2019). A joint deep neural network model for pain recognition from face. *In IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*, 52-56.

Majumder, A., Behera, L., & Subramanian, V.K. (2016). Automatic facial expression recognition system using deep network-based data fusion. *IEEE transactions on cybernetics*, *48*(1), 103-114.

Ousmane, A.M., Djara, T., & Vianou, A. (2019). Automatic recognition system of emotions expressed through the face using machine learning: Application to police interrogation

simulation. *In IEEE 3rd International Conference on Bio-engineering for Smart Technologies (BioSMART)*, 1-4.

Liang, J., Xu, C., Feng, Z., & Ma, X. (2015). Hidden Markov model decision forest for dynamic facial expression recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, *29*(7), 1556010.

Cao, N.T., Ton-That, A.H., & Choi, H.I. (2014). Facial expression recognition based on local binary pattern features and support vector machine. *International Journal of Pattern Recognition and Artificial Intelligence*, *28*(6), 1456012.

Kotsia, I., & Pitas, I. (2006). Facial expression recognition in image sequences using geometric deformation features and support vector machines. *IEEE transactions on image processing*, *16*(1), 172-187.

Shan, C., Gong, S., & Mc Owan, P.W. (2009). Facial expression recognition based on local binary patterns: A comprehensive study. *Image and vision Computing*, *27*(6), 803-816.

Sang, D.V., & Van Dat, N. (2017). Facial expression recognition using deep convolutional neural networks. *In IEEE 9th International Conference on Knowledge and Systems Engineering (KSE)*, 130-135.

Xie, L., Wei, H., Yang, W., & Zhang, K. (2014). Video-based facial expression recognition using histogram sequence of local gabor binary patterns from three orthogonal planes. *In IEEE Proceedings of the 33rd Chinese Control Conference*, 4772-4776.

Kumbhar, M., Jadhav, A., & Patil, M. (2012). Facial expression recognition based on image feature. *International Journal of Computer and Communication Engineering*, *1*(2), 117-119.

Ekweariri, A.N., & Yurtkan, K. (2017). Facial expression recognition using enhanced local binary patterns. *In IEEE 9th international conference on Computational Intelligence and Communication Networks (CICN)*, 43-47.

Guo, Y., Liu, Y., Georgiou, T., Lew, M.S. (2018). A review of semantic segmentation using deep neural networks. *International Journal of Multimedia Information Retrieval*, *7*(2), 87-93.

Anuse, A., & Vyas, V. (2001). A novel training algorithm for convolutional neural network. *Complex & Intelligent Systems, Springer*.

Wang, Y.Q. (2014). An analysis of the Viola-Jones face detection algorithm. *Image Processing on Line*, *4*, 128-148.

Dabhi, M.K., & Pancholi, B.K. (2016). Face detection system based on viola-jones algorithm. *International Journal of Science and Research (IJSR), 5*(4), 62-64.

Viola, P., & Jones, M.J. (2004). Robust real-time face detection. *International Journal of Computer Vision*, *57*(2), 137-154.

Liu, T. (2015). Implementation of Training Convolutional Neural Networks. *Computer Vision and Pattern Recognition*, 1-10.

El-Sawy, A., Hazem, E.B., & Loey, M. (2016). CNN for handwritten arabic digits recognition based on LeNet-5. *In International conference on advanced intelligent systems and informatics*, *Springer*, *Cham*, 566-575.

Yi, H., Shiyu, S., Xiusheng, D., Zhigang, C. (2016). A study on deep neural networks framework. *In IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, 1519-1522.

Xu, L., Fei, M., Zhou, W., & Yang, A. (2018). Face expression recognition based on convolutional neural network. *In IEEE Australian & New Zealand Control Conference (ANZCC)*, 115-118.

Peng, X., Xia, Z., Li, L., & Feng, X. (2016). Towards facial expression recognition in the wild: A new database and deep recognition system. *In Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 93-99.

Schapire, R.E. (1999). A brief introduction to boosting. *In International Joint Conference on Artificial Intelligence*, *99*, 1401-1406.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction. *Springer Science & Business Media*.

Al Taee, E.J. (2018). The Proposed Iraqi Vehicle License Plate Recognition System by Using Prewitt Edge Detection Algorithm. *Journal of Theoretical & Applied Information Technology*, *96*(10), 2754- 2764.

Coots, T.F. Active Appearance Models. *Proceeding European Conference on Computer Vision, (H.Burkhardt and B. Neumann Ed.s), Springer* 1998.

Shan, C., Gong, S., & Mc Owan, P.W. (2005). Robust facial expression recognition using local binary patterns. *In IEEE International Conference on Image Processing*, *2*, 2-370.

Viola, P., Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *In IEEE Proceedings of the computer society conference on computer vision and pattern recognition, CVPR*, *1*, 1-11.