Real-time Facial Emotion Recognition with Deep Neural Networks

Samad Azimi Abriz [a], Majid Meghdadi[b]

[a]Faculty of Engineering, Department of Computer Engineering, University of Zanjan, Iran, Samad.azimi.abriz@znu.ac.ir
[b]Associate Professor, Faculty of Engineering, Department of Computer Engineering, University of Zanjan, Iran, meghdadi@znu.ac.ir.

Article History: Received: 14 July 2020; Accepted: 2 January 2021; Published online: 5 February 2021

Abstract: Face emotions are an important part of human communications that helps to perceive others' intentions and behaviours. This crucial element creates a connection and interpersonal communications can be apprehended using this factor. So, there is no wonder that numerous researches have been dedicated to this matter during recent decades and facial emotion recognition is a critical matter in computer vision and artificial intelligence. One of the challenges of AI is to achieve high accuracy and great performance in the model. Fortunately, this has been dealt with by emerging deep learning networks and setting aside traditional machine learning methods. In this study, we established a model called ResEZAP (Residual Extended Zero Average Pooling) of deep learning networks for real-time facial emotion recognition and achieving an acceptable accuracy by reducing computational complexity. In this paper, the FER2013 dataset is used for training and the model accuracy with the test dataset is 69.74.

Keywords: Deep learning network, Convolution layers, Resnet network, Real time detection system, Fer, Depthwise separable convolution

1. Introduction

Studies have shown that most communications are non-verbal. Facial emotion is one of the critical factors using which a person with any language can communicate easily with others without knowing their identity. Darwin [1] believed that humans show similar facial emotions when reacting to different types of feelings. Another researcher named Paul Ekman [2] argued that humans show instinctively coded signals during different emotional states; a blind person shows a similar facial emotion to a normal person in emotional events and basic emotions [3] occur in all humans. Facial emotion recognition was done statically on fixed images [4] and by an improvement of deep learning, development of new techniques in this field, and with new powerful hardware, real-time recognition of dynamic images has thrived and we expect significant progress in real-time recognition of all fields in the future. Nowadays, facial emotion recognition has important applications in fields such as the study of society mental health, safety, autism patients, animating, and the relationship between human and machine, promoting further studies in this field.

Facial emotion recognition in artificial intelligence is performed in two common machine learning and deep learning methods. In common machine learning methods, feature extraction is conducted manually and some of the widely used types are SVM and Decision tree. However, in deep learning methods, features are obtained automatically and one of the commonly used types is CNN with three major layers of Full Connected, Pooling, and Convolution. This method extracts features until it connects to a classifier and facial emotion is recognized. Similar to other machine learning fields, especially machine and computer vision, FER consists of three main phases, as shown in Fig. 1:

Preprocessing: Before inputting data to the model for training, operations such as normalization, brightness and size changes, and one of the most applied technique, adding to the training data (Data augmentation), are performed to better train the model.

Model learning: Designing a model is one of the main sections of deep learning. The model must perform properly on test data while learning training data well and extracting features.

Classification: Using desired functions such as softmax, the class of the input is determined.

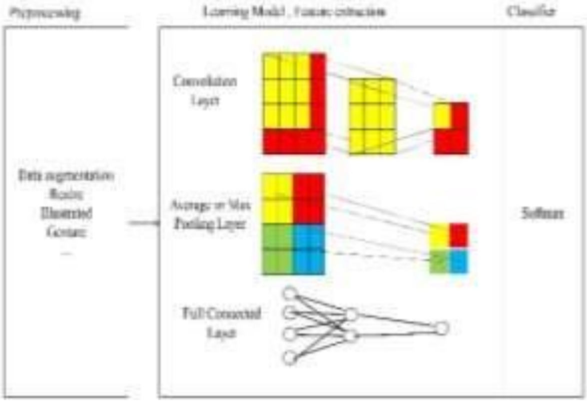

Fig. 1. Neural network model that receives input and performs preprocessing feature extraction, and classification, respectively.

2. Related works

Arriaga et al. [5] proposed a system that runs real-time gender and facial expression classification using convolutional neural networks. In this study, two models are proposed that both are designed based on the highest precision per number of parameters. The first model relies on the complete removal of fully connected layers and in the second model, in addition to the elimination of fully connected layers, residual [6] and Xception [7] modules are combined with wide-depth separatable convolutions [8]. This model has 60000 parameters with 888 KB size and 66% accuracy in the test dataset. Shabab et al. [9] searched for a deep network model for smartphones and showed that classifying accuracy is significantly low when the model is trained with one dataset and tested with a different dataset. They aimed to measure the error when a network learns one data set and is tested on different datasets. They tested different models with three layers on numerous data sets and concluded that a huge data set is needed to achieve a powerful deep network. Hence, they combined the datasets and attained better results. To classify human emotions, Donne et al. [10] used real-time facial expression recognition. They transport learning on VGG fully-connected layers; the training and testing accuracies are 90.7% and 57.1%, respectively. In the end, real-time images after facial recognition are transferred into the neural network. This network then classifies an arbitrary number of faces in each image with corresponding emoji on individuals' faces. Mira et al. [12] proposed a FER algorithm to observe driver feelings that can be executed in recognition equipments with poor hardware systems and is installable in vehicles. Hierarchial weighted random forest (WRF) that is trained based on similarity of samples is used for accuracy improvement. To reduce load of feature extraction in a real-time system, they proposed a geometric feature descriptor based on locational relations between face parts using the ratio between distance and angle. Furthermore, for classifying facial expressions, they proposed hierarchical classifier WRF.

3. Proposed method

Proposed model and network is transformed forms of Resnet network blocks. Global Max Pooling was applied in ending layers and there are an Average Pooling layer and a Zero padding layer after each block. The entire model is consisted of three blocks that several calculations, including 6 activation functions, are removed and number of parameters are reduced to improve processing speed and power. Comparison of the designed block with RESnet standard block can be observed in Fig. 2. The model consists of three blocks and each includes three convolution layers with the same filter dimensions in each block. All filters are 3×3 and the depth of filters in the first, second, and third blocks are, respectively, 32, 64, and 96. The output of each layer is sent to the next layer and the output of the first layer is added to the output of the third layer and sent to the next block. After the last convolution layer in each block, we have Relu [13] activation function and class normalization [14]. After the last block, Global Max Pooling layer and then only a full connected layer with 96 features are present. Ultimately, the probability of classes is computed using Softmax. In this model, Adam [15] is utilized for optimization and quickening rate of convergence. Fig. 3 represents the entire architecture of the proposed network.

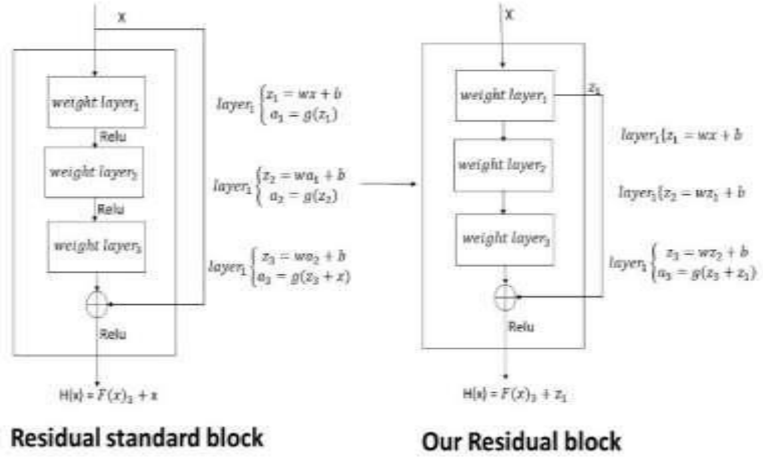
x
weight layer$_1$
Relu
weight layer$_2$
Relu
weight layer$_3$
Relu
$\text{H(x)} = F(x)_3 + x$
$layer_1 \begin{cases} z_1 = wx + b \\ a_1 = g(z_1) \end{cases}$
$layer_2 \begin{cases} z_2 = wa_1 + b \\ a_2 = g(z_2) \end{cases}$
$layer_3 \begin{cases} z_3 = wa_2 + b \\ a_3 = g(z_3 + x) \end{cases}$
Residual standard block
x
weight layer$_1$
z_1
weight layer$_2$
weight layer$_3$
Relu
$\text{H(x)} = F(x)_3 + z_1$
$layer_1 \{ z_1 = wx + b$
$layer_2 \{ z_2 = wz_1 + b$
$layer_3 \begin{cases} z_3 = wz_2 + b \\ a_3 = g(z_3 + z_1) \end{cases}$
Our Residual block

Fig. 2. Comparison of the proposed block with standard type.

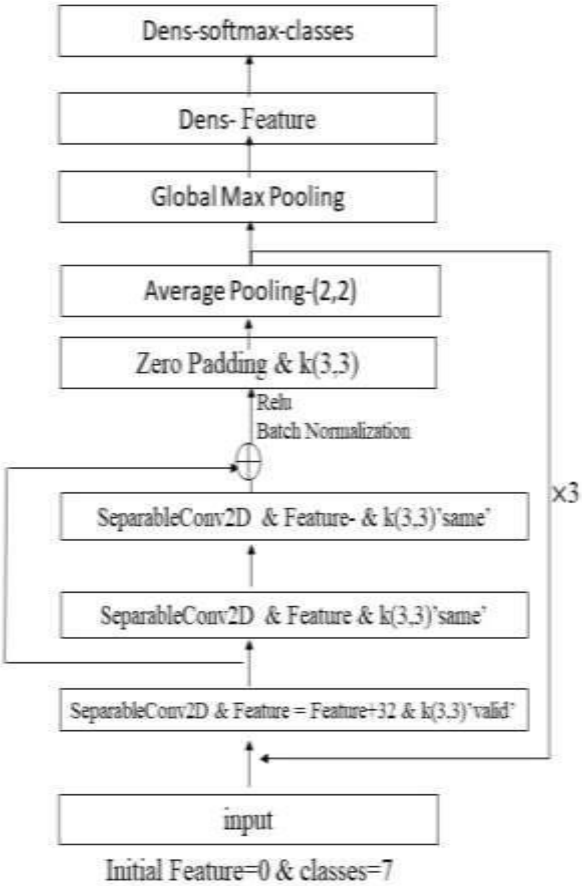
Dens-softmax-classes
Dens- Feature
Global Max Pooling
Average Pooling-(2,2)
Zero Padding & k(3,3)
Relu
Batch Normalization
SeparableConv2D & Feature- & k(3,3)'same'
SeparableConv2D & Feature & k(3,3)'same'
SeparableConv2D & Feature = Feature+32 & k(3,3)'valid'
×3
input
Initial Feature=0 & classes=7

Fig. 3. Architecture of the proposed network.

In the proposed model, Depthwise separable convolution layers were used instead of normal convolution that reduces the number of parameters and computation complexity and the procedure in these layers as compared with normal convolution is described.

Separable convolution firstly deals with the spatial dimensions of an image and filter. However, Depthwise Separable Convolution considers spatial dimensions and depth, number of channels. An input image may have three channels: RGB. After some convolutions, an image may have several channels. You can regard each channel as a specific interpretation of the image. For example, “red”, “blue”, and “green” channels interpret each red, blue, and green pixel. An image with 64 channels suggests 64 different interpretations. Similar to separable convolution, a Depthwise Separable Convolution divides a filter into two separate kernels that perform two convolutions: Depthwise and pointwise convolutions. Depthwise Separable Convolution is an excellent idea in real-time systems since it requires a low computational load with a low number of parameters compared with normal convolution. First, the multiplications procedure of Depthwise Separable Convolution is explained. Then the number of calculations in these layers is compared with a normal Convolution layer and their significance is realized. Depthwise Separable Convolution layer separates the kernel into two parts. Firstly, it isolates depth and corresponding channels are multiplied. Secondly, by creating 1×1 kernels in depth, multiplications are performed and the ultimate form is obtained after convolution. For instance, the filter separates a 3×3×32 channel (32 is depth) into two 3×3×1 and 1×1×32 kernels. Fig. 4 shows a convolution process in Depthwise Separable Convolution.

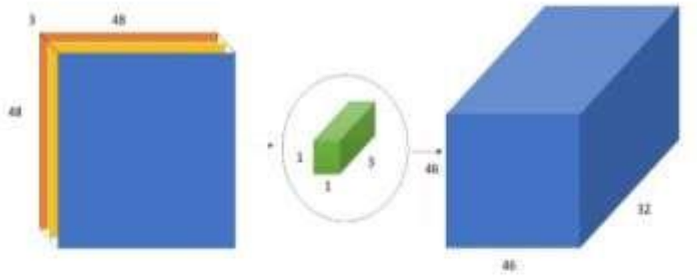

Fig. 4. Execution procedure of depthwise separable convolution layer.

Model is implemented once with normal Convolution and another time with Depthwise Separable Convolution layers and the number of multiplications for each is computed to illuminate how using Depthwise Separable Convolution layers helps model and network by reducing the number of multiplications and computation complexity. In the following computation of the number of multiplication operations for the first layer of the first block is shown and in Table 1, the difference between the number of multiplications and system calculations with two implementations can be observed.

The number of calculations for the first layer by the implementation of normal convolution layer:

$$46 \times 46 \times 1 \times 3 \times 3 \times 32=609408$$

Number of computations for the first layer by implementing Depthwise Separable Convolution layer:

$$46 \times 46 \times 1 \times 3 \times 3=19044$$

$$46 \times 46 \times 32 \times 1 \times 1 \times 1=67712$$

$$19044 + 67712=87756$$

The difference between the number of multiplications in two layers:

$$86756-609408=521652$$

As can be seen, Depthwise Separable Convolution has a lower number of multiplications in comparison to normal convolution layer with a larger difference in this number in deeper layers. The number of multiplications for all 9 layers is computed for normal convolution and Depthwise Separable Convolution layers. As can be noted in Table 1, Depthwise Separable Convolution layers have far lower multiplications as compared to that in normal Convolutions that aids in deep learning, especially in heavy computational load.

Table 1. Number of multiplications for convolution layers.

Layer	Number of multiplications	Number of layers
Normal Convolution	130075776	9
Depthwise Separable Convolution	16909540	9

According to Table 1, Depthwise Separable Convolution performed multiplication 113166236 times less than normal Convolution with reduced computation and processing operation. In the proposed model, all block filters are 3×3 and their form after model training in each block can be seen in Fig. 5:

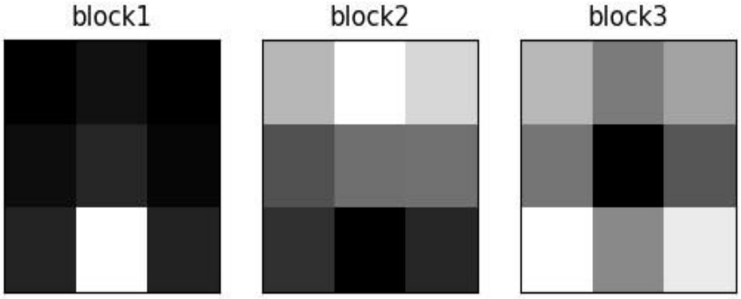

Fig. 5. Form of filters in each block of our proposed model.

Global pooling layer reduces dimensions from three or two dimensions to one with no need for flatting the convolution layers. Thus, global pooling layer represents one response for each feature plot. This can be maximizing, averaging or any other operation. Global pooling layers are an essential part of neural networks (CNN). Global average pooling or global max pooling are used to transform convolutional features from size-varying images to fixed-size. The implementing process can be seen in Fig. 6. Global pooling layer transforms the entire feature map into one value. Global pooling layers prevent overfitting to some extent by global averaging and global max pooling the feature maps.

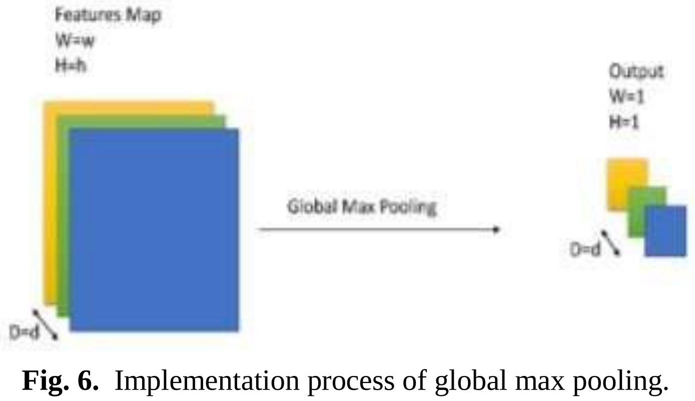

Fig. 6. Implementation process of global max pooling.

The overall size of the model is 331 KB with almost 54000 parameters. Its accuracy in the test set is 69.49, which is acceptable for real-time systems in this size, number of parameters, number of calculations, and processing load. The proposed method is compared in Table 2 to other related works with the same datasets. It can be noted that the proposed method performed better than others in terms of the number of parameters, weight, and accuracy, indicating that deep learning network can be used in real-time systems.

Table 2. Performance of some real-time face emotion recognition methods with FER2013 dataset

	Total	Size	test
Method param	weights	accuracy	
Our proposal	54480	331 KB	69.74
Jeon [16]	4285863	16.41 MB	69.04
Arriaga[5]	60000	855 KB	66
Talegaonkar [17]	2375879	9.15 MB	60.12
Samsani [18]	250000	94 KB	60.10

To achieve a real-time deep learning facial emotion model, the following six operations were performed:

1. Data augmentation(Keras library is used): With data augmentation from original data using operations such as orientation, brightness changing, cropping, and others, the number of training samples for the network increases.
2. Since one of the reasons for overfitting is the presence of a large number of fully connected layers in the ending section of the model, instead of using multiple fully connected layers, one global max pooling layer with only one fully connected layer is used.
3. Using average pooling layer with zero padding after each block.
4. Using a Depthwise separable convolution layer instead of normal Convolution reduced the number of parameters from 345319 to 54480 by eliminating 290839 parameters. This increased the speed of the model and prevented overfitting. Also, the size was decreased from 1.43 megabyte to 331 KB.
5. Only one activation function was used in each block instead of three and consequently, a total of 3 activation functions were used in the model that reduced the computing process by six times.
6. Designing a new block and changing the normal Resnet standard block decreased the number of parameters from 66488 to 54480 by 12008.

4. Experiments

In this article, the standard FER2013 dataset, available on the Kaggle website [19] is used. Data include black & white 48×48 images from faces. Faces were extracted automatically and cover the entire image. We aim to classify each face based on emotions into one of the seven classes (angry=0, disgust=1, fear=2, happy=3, sad=4, surprise=5, neutral=6). The number of training samples for each emotion can be seen in Fig. 7.

Train.csv includes two columns, “emotions” and “pixel”. Column “emotions” contain a number from 0-6 for emotion expressed in the image. The column for ”pixels” involve a sequential string of numbers, derived from each image. Test.csv include only column “pixels” and our duty is to predict emotions column. This dataset is provided by Pierre-Luc Carrier and Aaron Courville as part of a research project. The model is trained with this dataset and real-time facial emotion recognition is performed in stages, as shown in Fig. 9. The number of samples in each class and confused matrix is shown in Fig. 7 and Fig. 8.

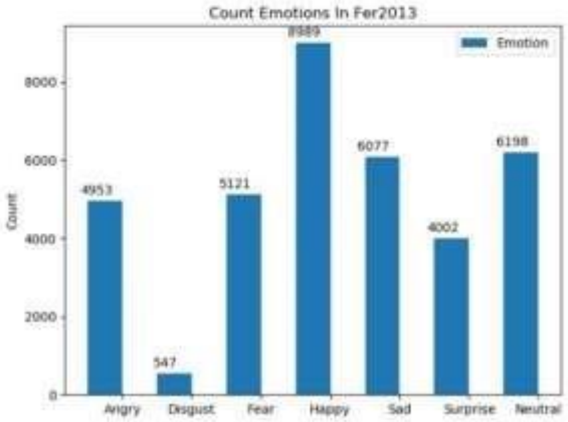

Fig. 7. Number of training samples for each emotion in the FER2013 database.

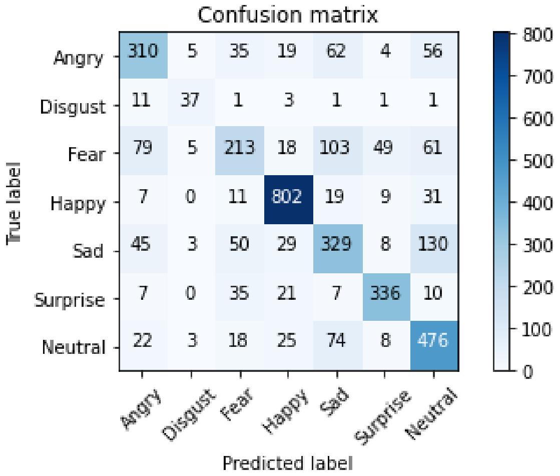

Fig. 8. Confused matrix on FER2013 test set

After training, real-time facial expression recognition was tested on a sample in DELL i5 8200 laptop and a prediction time of 0.007 seconds was achieved, indicating that the proposed model is fast. Realtime facial expression recognition is done in five phases, as shown Fig. 9:

1. Input
2. Extraction and identification of faces
3. Pre-processing

4. Classification and prediction of emotion
5. Output

The steps for recognizing the facial emotion of a test sample from entry to end are shown in Fig. 9.

Input: Data is inputted to the system through different ways, such as a webcam, connected camera, or any device that can transfer images to the model.

Facial extraction and detection: Inputted images to the system are turned to black-white. Afterwards, the faces are detected using an opencv library and quick Haarcascade tool [21] and desired regions are cropped

Preprocessing: In this step, after isolating the region filled by face, the image is resized into 48×48 and normalized. Then, it is turned into an array and prepared for inputting after adding one dimension.

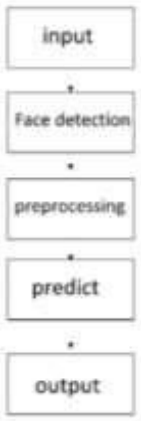

Fig. 9. steps of real-time facial expression recognition.

Prediction: Received image is given to the model after preprocessing so that to define the class for facial expression. This process is performed based on weights and extracted features.

Output: After model prediction, the class with the highest probability is obtained and printed as the final result in the image.

Face expression recognition steps of a test sample from inputting to final step are shown in Fig. 10.

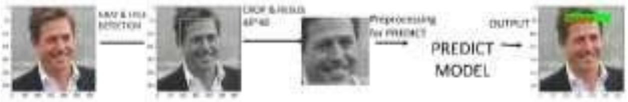

Fig. 10. Steps to recognize the facial emotion of an input sample, an image taken from the IMDB-WIKI database [20]

The output of each block during feature extraction and the definition of related class for an input sample can be observed in Fig. 11. The deeper a network is, the more complex the features are. By noting the output of blocks, we note that the first block identifies edges and extracts simple features. In the following blocks that network becomes deeper, the image blurs and features become more complex. Other features are extracted and finally classified emotion.

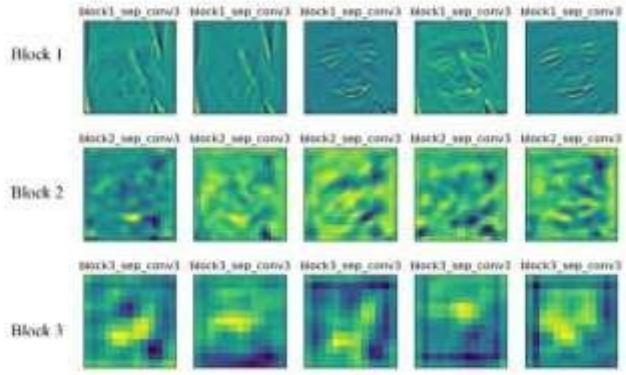

Fig. 11 . The output of each block when an input sample is given to the model in Fig. 9.

In Fig. 12, the model output on some samples selected randomly from the FER2013 dataset can be seen.

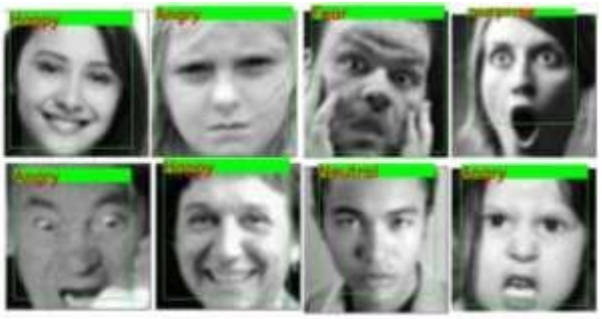

Fig. 12. Model output to examples from the FER2013 data set.

5. Conclusion

According to the experiment results, the proposed model was capable of solving some of the most fundamental issues of deep learning networks, including large size and the number of parameters, computation complexity, and hardware requirements, paving the way for further researches and model design in these fields for smartphones that can achieve acceptable results with its capacities. By comparing the proposed model with other methods, it was observed that although the model is light and has a low number of parameters, it gives acceptable results. The strength of deep learning network as compared with common machine learning methods is their high accuracy and their weakness is high computational complexity and a large number of parameters that lead to an increased processing load. However, in this paper, despite the proper performance of the model, processing load has decreased and we can conclude that deep neural networks are not dependent on a large number of parameters for proper learning, and other low-parameter techniques can be used to achieve better results.Therefore, these networks can be applied in devices with low storage and low computational power. With a low computational complexity in deep learning networks, the best idea is to use these networks in real-time systems and it is expected that these networks be used more frequently in real-time systems. The proposed model is a combination of other networks, as a result, a combination of successful models such as VGG and Squeeze net with other networks can be used in the format of a united network and normal Convolution layers can be replaced with Depthwise Separable Convolution to reduce computational complexity. Since full connected layers cause overfitting, they can be removed from the end of the network and global pooling layers can be placed. A large dataset is needed for better training of the network and model. Datasets plays an essential role in deep learning networks and consequently,some methodes such as GAN networks can be used to solve small datasets and increase the training dataset. Ultimately, it is expected that more attention is dedicated to light deep learning networks and we see deep learning networks that are very light and applied in various fields so that systems perform more accurately and respond more quickly.

References

1. Charles D, Phillip P, "The Expression of the Emotions in Man and Animals," *USA,: Oxford University Press,* 1998.
2. Ekman P, Friesen W, "The repertoire of nonverbal behavior: Categories, origins, usage, and coding," *Mouton Classics: From Syntax to Cognition. From Phonology to Text,,* pp. 19-868, 2013.
3. Izard C, "Basic emotions, relations among emotions, and emotion-cognition relations," *APA PsycArticles,* p. 561–565, 1992.
4. Ng H, Nguyen V, Vonikakis V et al, "Deep learning for emotion recognition on small datasets using transfer learning," *CMI 2015 - Proceedings of the 2015 ACM International Conference on Multimodal,* pp. 443-449, 2015.
5. Arriaga OValdenegro-Toro MPlöger P, "Real-time Convolutional Neural Networks for Emotion and Gender Classification," Cornell University Library. Retrieved from https://arxiv.org/abs/1710.07557v1, 2017.
6. He K, Zhang X, Ren S et al, "Deep Residual Learning for Image Recognition," In Proceedings of the IEEE conference on computer vision and pattern recognition, p. 770–778, 2016.
7. Chollet F, "Xception: Deep learning with depthwise separable," *https://arxiv.org/abs/1610.02357,* 2016.
8. Andrew G. Howard et al, "Mobilenets: Efficient convolutional neural," *https://arxiv.org/abs/1704.04861,* 2017.
9. Bazrafkan S, Nedelcu T, Filipczuk P, Corcoran P, "Deep learning for facial expression recognition: A step closer to a smartphone that knows your moods," *2017 IEEE International Conference on Consumer Electronics,* pp. 217-220, 2017.
10. Duncan D, Shine G, English C, "Facial Emotion recognition in Real Time," *Stanford University,* pp. 1-7, 2016.
11. Chatfield KSimonyan KVedaldi A et al, "Return of the devil in the details: Delving deep into convolutional," *British Machine Vision Conference,* 2014.
12. Jeong M, Ko B, "Driver's facial expression recognition in real-time for safe driving," *Sensors,* 2018.
13. Glorot X, Bordes A, Bengio Y, "Deep sparse," *Proceedings of the Fourteenth International,* p. 315–323, 2011.
14. Ioffe S, Szegedy C, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *https://arxiv.org/abs/1502.03167 ,32nd International Conference on Machine Learning,* pp. 48-456, 2015.
15. Kingma D, Ba J, "Adam: A method for stochastic optimization," *3rd International Conference on Learning Representations, https://arxiv.org/abs/1412.6980,* 2015.
16. Jeon J, Park J,Jo Y et al, "A real-time facial expression recognizer using deep neural network," in *ACM IMCOM 2016: Proceedings of the 10th International Conference on Ubiquitous Information Management and Communication,* 2016.
17. Talegaonkar I, Joshi K,Valunj S et al, "Real Time Facial Expression Recognition using Deep Learning," *SSRN Electronic Journal,* 2019.
18. Samsani S, Gottala V, "A Real-Time Automatic Human Facial Expression Recognition System Using Deep Neural Networks," *Springer Verlag,* pp. 431-441, 2020.

19. "KaggleChallengesDataset,"*https://www.kaggle.com/c/challenges-in-representationlearningfacial-*.
20. "https://github.com/opencv/opencv/tree/master/data/haarcascades," opencv.
21. Rothe R, Timofte R, Van Gool L, "IMDB-WIKI – 500k+ face images with age and gender labels," *https://data.vision.ee.ethz.ch/cvl/rrothe/imdb-wiki/*.