Research Article

Real-Time Gender Recognition with a Deep Neural Network

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: Nowadays the existence of artificial intelligence (AI) and convolutional networks had led to significant progress in machine vision. Machine vision can automatically perform many tasks that are difficult and arduous and have a high error for humans. One of these difficult tasks is the determination of gender that nowadays has many applications. Using AI and machine vision to determine gender can speed up this process. Deep neural networks have had significant progress in comparison to other common machine learning methods but the number of parameters and calculations is one of the major issues of these networks. In this paper, we have presented a real-time deep neural network model that performs gender recognition faster and with fewer calculations by reducing the model parameters and calculations. The proposed model is a rather light model and a mixture of multifold filters that have been trained and tested on three datasets Wikipedia, Audience, Celeba.

Keywords: Deep learning network, Convolution layers, Gender classification, Real time detection system, Multifold filters, Depthwise separable convolution

1. Introduction

When people meet each other, visual information plays an important role. When we look at somebody's face, not only do we understand who they are, but we collect information about their gender, race, age, and current mental status based on their moods. Gender classification and recognition of a person are based on their face. This is a simple task for humans, but a complex one for machines. Gender classification can be important in the interaction of man and machine. Also, this is a useful preprocessing step in face recognition. A computer system equipped with a gender classification functionality has extensive applications in basic and applied research such as human and machine connection, security, law enforcement, demographic studies, psychology, education, telecommunication, etc. Face recognition has been studied by many researchers. But only a few works have paid any attention to gender classification. Faces are used for gender classification, therefore the process of gender classification can double the face recognition process speed by reducing the search time for identifying the individual.

Gender classification is a subject that was first presented in psychological studies. These studies have focused on understanding human visual processing and finding the key features used for classifying women and men. Studies show that you can use the differences between male and female faces to improve the performance of face recognition software in biometrics, human and machine interaction, monitoring, and machine vision. However, the challenge in real-world is how to deal with factors such as lighting, gesture, facial expressions, obstruction, and background information, for this, we can train a model based on rich and real world-like datasets, therefore this paper has used Wikipedia, Audience and Celeba datasets which are close to reality. This is also a challenge in developing face-based real-time gender classification systems that have a high classification accuracy and proper performance in real-time.

In this paper, we use a new real-time gender classification method based on CNN. The main portion of this paper is as follows: A three-block deep CNN, in which each block has three layers and Separable Convulsion Depthwise layers have been used instead of Normal Convulsion layers to reduce calculation for real-time face-based gender classification. The network structure in comparison with other methods is less complex with fewer layers and neurons for learning. We have created a new algorithm in the convulsion layer mixture with multifold filters improving CNN performance in classification rate and processing speed. We have trained and tested our CNN method for gender recognition in three datasets and the results show that our method performed better in the number of calculations, complexity, accuracy, processing, and response time in comparison to the other models. The second section of this paper reviews related papers, the third section defines the proposed method, the fourth section presents the experiments, the datasets, and their results and finally, the conclusions and suggestions are presented.

2. Related work

[4] Proposed a neural method called Pretrained Inception that recognized the face using a model called Facenet. Their proposed method can be divided into three stages. First, the face in each picture is recognized and cropped. Second, the face pictures are given to the neural network. Third, the gender is classified using adjusted weights. This network used 1*1, 3*3, and 5*5 filters. The UTK Face dataset was used for training and testing this model. This paper analyzes all machine learning methods. [5] Proposed real-time deep neural network model with a unique pre-processing that uses four convolutions, three pooling, and two fully connected layers. The CAS-PEAL and FEI datasets were used for training this model and they achieved 98% accuracy in the mixture of these two sets. [6] Proposed a K-means clustering machine learning method using multi-feature, that first increases picture quality and contrast then recognized and crops the face using a deep neural network, then uses the SIFT descriptor to extract the features and finally perform gender classification. [7] Proposed a neural network method using the ICA algorithm for adjusting the weights. They extracted the face using the Viola algorithm. Then the features were selected using the NSGA-II algorithm and finally an artificial neural network (ANN) with ICA (ANN-ICA) performed the gender classification using facial components. The data used in this paper were extracted randomly from the Bao, Indian Face, MIT-CBCL, F EI Face and FRI CVL Face databases. [13] Presents a deep neural network in which facial components are selected and categorized using the dlib library. The proposed VGG-Face network is created from 13 convulsion layers that use the three Adience, Wikipedia, and Labelled Faces in the Wild (LFW) datasets with an 87.08% accuracy for Adience and an 98.45% accuracy for LFW. [17] Creates an age and gender classification system that contains two networks for classifying age and gender based on the GoogLeNet deep neural network with the help of the Caffe deep learning framework that classifies the gender and age of pictures recorded by cameras.They used the Adience and CAS-PEAL datasets [18] to train GoogLeNet. [19] Proposes a multifold deep neural network, which is created from a series of sub-networks and each network extracts different features. Each network is separately trained on the AGFW dataset, and then a voting system is used to combine these predictions and reaching a conclusion. The whole system of this paper consists of the pre-processing, deep neural networks, and the voting system stages. The main purpose of using multiple sub-networks is to increase the prediction accuracy and decrease overfitting. Three subnetworks were proposed in this method that nearly have 10 million, 5 million, and 5 million parameters for learning. [20] Proposes a framework that first splits the face into the different facial components and then automatically performs gender classification. A Conditional Random Field (CRF) based classifier model was trained using facial components with manual labeling and then a Random Decision Forest (RDF) classifier was trained for classifying faces into female or male groups. The performance of this proposed method was measured in the Adience, LFW, FERET, and FEI datasets with an accuracy of 91.4%, 94.4%, 100%, and 93.7% accordingly.

3. Proposed method

The proposed method is in real-time that uses neural networks because of their high capabilities based on six ideas; Fig. 1 shows the architecture of the proposed deep learning network. The first idea was using Depthwise Separable Convolution layers instead of normal Convolution layers which will considerably decrease the number of calculations and learning parameters. The second idea is using Global Average Pooling layers instead of flatting after the last block and using Full Connected layers. The third idea removes Full Connected layers from the network which caused an increase in the computational load and overfitting. The fourth idea was using filters with different kernels which allows the network to extract different features and present a more accurate classification. The fifth idea was concatenating the layer outputs with different filters in each block and forwarding them for the input of the next block layers independently which are opposite to the last method of trying to connect layers with similar filter kernels. The sixth idea was using the least possible parameters and gaining the most possible accuracy. In other, for the model and network to be suitable for real-time purposes, the proposed model requires to have limited calculations and learning parameters.

The proposed model uses Depthwise Separable Convolution layers instead of normal Convolution to solve this important problem because they use around 1/n fewer parameters and calculations in comparison with normal Convolution layers which as the network becomes deeper, the calculations and speed difference of these two layers become more and more prominent; therefore our proposed method fully used Depthwise Separable Convolution layers alongside the Inception module in the network structure. Depthwise Separable Convolution is a great idea in real-time systems because of its need for fewer calculations and learning parameters in comparison with normal Convolution layers; Fig. 2 shows how these layers work. Depthwise Separable Convolution separates filters into two parts, first the deep is separated and peer-to-peer channels are multiplied, and the second step creates 1*1 kernels in the multiplication depths, and the final form is created after the Convolution. For example, if our filter is like a 3*3*32 in which 32 is channel depth, this layer seperates that into two 3*3*1 and 1*1*32 cores. The proposed method consists from three blocks and each block uses three layers with 5*5, 3*3 and 1*1

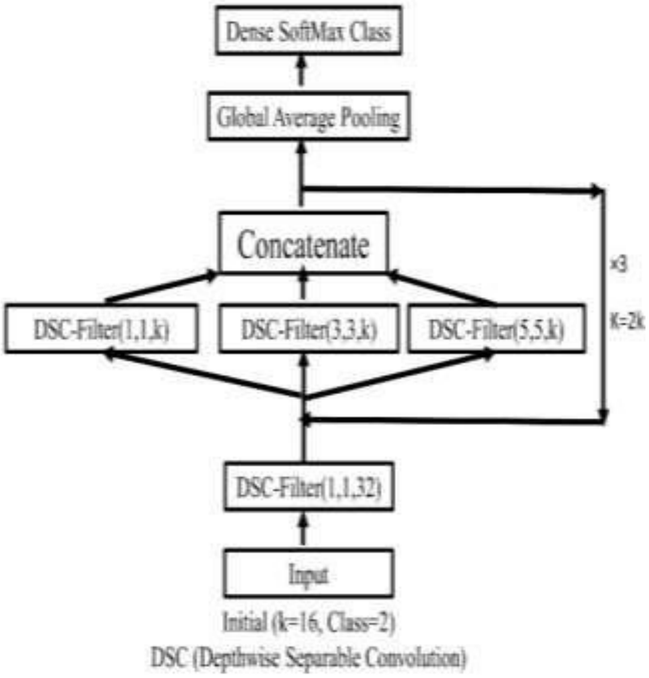


Fig. 1. The Architecture of the proposed network.

filters which similar filter depths in each block; Filter depth in the first, second and third block is 16, 32 and 64, accordingly. In each block, the input layers are taken independetly from the output of the privious block and the output of each block is concatenated and forwarded to the next block.

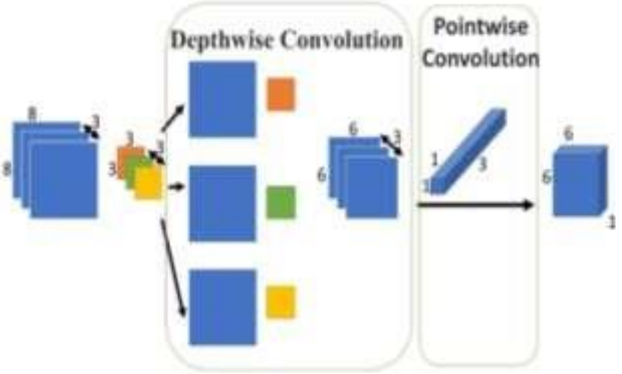


Fig. 2. Depthwise Separable Convolution layer execution steps.

In the proposed method, a Global Average Pooling has been placed after the last block to prevent overfitting and no Full Connected layers were used. Global Average Pooling layers, as you can see how they work in Fig. 3, reduce the dimensions into one and remove the need for flatting the Convolution layer. The Global Pooling layers can be of the maximizing or averaging types. Global Pooling layers are an important part of Convolutional Neural Networks (CNN). They are used for gathering activations from spatial locations for creating a fixed vector in multiple points of the CNN. The Global Average Pooling or Global Max Pooling layers are used for changing Convolution features from varied values into a fixed value. Instead of removing sampling parts from the feature input map, the Global Pooling layer converts the whole map into a single value. Afterward, the layers will be directly connected to the classifier. There are not Full Connected layers in the network which reduces the high number of parameters and calculations and also the simultaneous usage of Global Pooling layers has prevented overfitting.

The proposed method first receives the input using the Depthwise Separable Convolution and 1*1*32 filters, then passes it on to the first block. Each layer receives its input indepently and there is no connection between these layers in each blocks, meaning that the output of one layer is not sent to another layer and the outputs are only concatenated with each other. The purpose of using filters with different kernels in each block is to extract different features so that the network could perform better in classification.

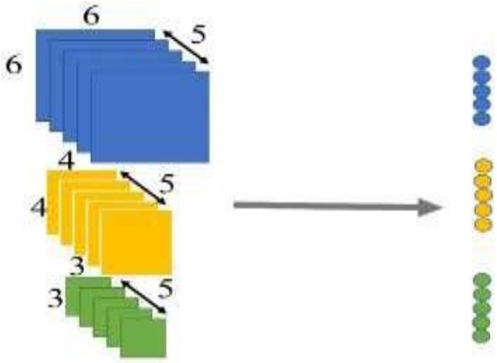


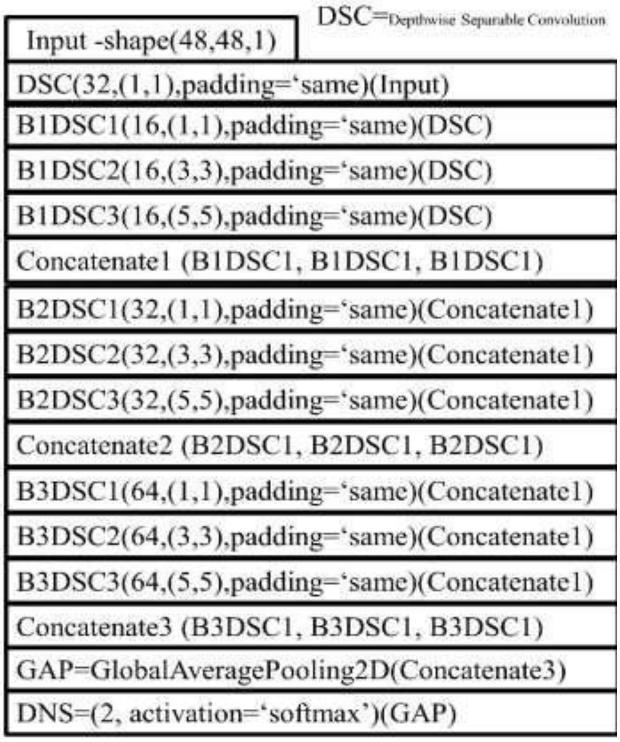
Fig. 3. Global Pooling Layers execution

The proposed network is rather light, and it would be safe to say that this network is one of the lightest yet strongest deep-learning networks that are appropriate for a real-time system that has around 32,000 learning parameters and overall size of 260 kilobytes. After each layer of convolution in each block, the Relu activation function [21], the Batch normalization [22] and Max Pooling are used; Adam was used in this model to accelerate and optimize convergence. This model is rather light and can have a good response time in real-time systems. The model is a mixture of Depthwise Separable Convolution layers and the Inception module. We have tried to keep this model light while extracting better features to reduce error and increase accuracy. This lightness allows the system to be better in real-time and have a better performance. Fig. 4 shows the implementation of the proposed network in Keras.

Table 1. Results and comparing related works and the proposed method with different datasets

#	Capacity	Param	Acc(%) -Wild	Acc(%) Audience	Acc(%)- –Celeba	Acc(%) -FEI+CAS
Our Method	260 KB	32000	95	91.56	60.12	-
Arriaga et al. [25]	833 KB	60000	95	-	-	-
Ari et al. [3]	30 MB	7907714	-	85.9	-	-
Khurram et al.[5]	17 MB	4520450	-	-	-	95
Brian et al. [13]	VGG-Face	42243953	92.69	87.08	-	-

As was mentioned before, each block uses different filters. Fig. 5 shows the filters after network training while Fig. 6, 7, and 8 show how to extract the features of an example for each filter of each block. We can conclude, based on the Shape of each filter and how they extract the features, that each filter extract different features, and this diversity will make the network stronger, allows it to learn better, and perform classification with high accuracy and low error. As you can see, the first layers and the first block only extract simple features and the edges of the picture and as the layers get deeper, the filter shape blurs while extracting the features and more complex features are extracted.



DSC=Depthwise Separable Convolution

Input -shape(48,48,1)

DSC(32,(1,1),padding='same)(Input)

B1DSC1(16,(1,1),padding='same)(DSC)

B1DSC2(16,(3,3),padding='same)(DSC)

B1DSC3(16,(5,5),padding='same)(DSC)

Concatenate1 (B1DSC1, B1DSC1, B1DSC1)

B2DSC1(32,(1,1),padding='same)(Concatenate1)

B2DSC2(32,(3,3),padding='same)(Concatenate1)

B2DSC3(32,(5,5),padding='same)(Concatenate1)

Concatenate2 (B2DSC1, B2DSC1, B2DSC1)

B3DSC1(64,(1,1),padding='same)(Concatenate1)

B3DSC2(64,(3,3),padding='same)(Concatenate1)

B3DSC3(64,(5,5),padding='same)(Concatenate1)

Concatenate3 (B3DSC1, B3DSC1, B3DSC1)

GAP=GlobalAveragePooling2D(Concatenate3)

DNS=(2, activation='softmax')(GAP)

Fig. 4. Proposed network implementation with Keras.

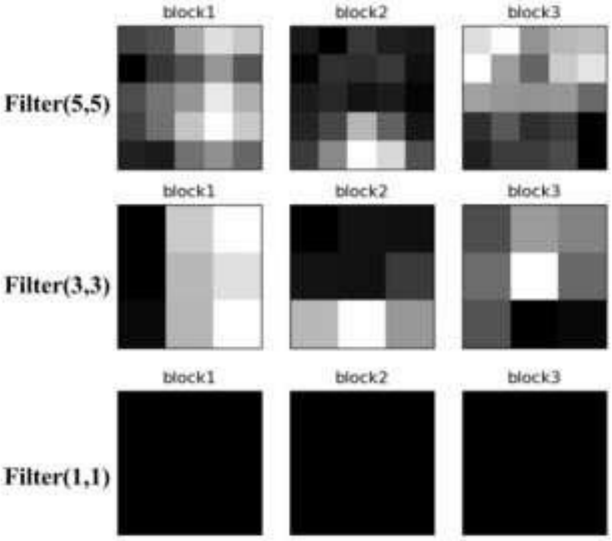


Fig. 5. Filter shapes in blocks, after training network

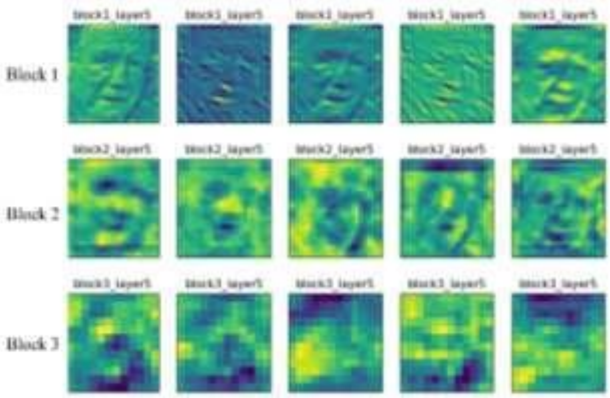


Fig. 6. Extracting the features with 5*5 filters in different blocks

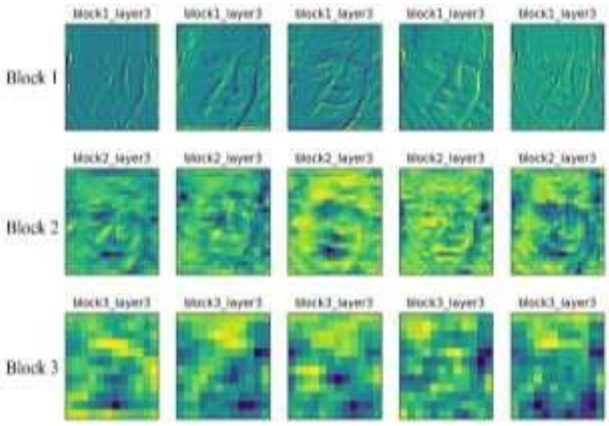


Fig. 7. Extracting the features with 3*3 filters in different blocks

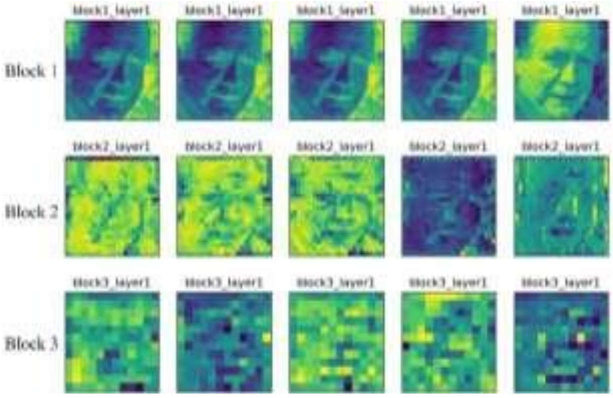


Fig. 8. Extracting the features with 1*1 filters in different blocks

4. Experiments

This paper uses the three Wikipedia, Audience, and Celeba datasets. The Wikipedia dataset includes 62328 pictures that are graded based on their quality, the headshots or the components are not separated and some pictures even include unclear faces. The Audience includes 18591 pictures, some of which were pictures of unrelated subjects that were removed before the model learning. After the data preprocessing, they were sent to the model for learning. The accuracy of our proposed model was 95.20% on the Wikipedia set, 91.5% on the Audience set without pre-processing and 60.12% on the Celeba set without pre-processing. Our model size is equal to 260 KB; Our proposed model in comparison with other works showed a high speed and low numbers of calculations and parameters alongside a good accuracy which is depicted in Table 1.

Table 2 compares the results of our proposed method and other related works after learning and testing the Audience dataset. The results show that proposed model, even with fewer parameters, was able to perform well, therefore more parameters and a deeper network don't necessarily correlate with better and more accurate results.

Table 2. Compare related work on the Audience dataset[13]

Dataset		Accuracy(%)	Method
Audience	Our Method	91.56	
	Khan et al. [20]	91.4	
	Levi et al. [26]	86.8	
	Lapuschkin et al. [27]	85.9	
	CNNs-EML [28]	77.8	
	Hassner et al. [29]	79.3	
	Brian et al. [13]	87.8	

The real-time system operates in three steps: Input and pre-processing, prediction, printing. When the input is given to the system, the face is recognized, cropped, resized, predicted, and readied for the model to select the proper class. Fig. 9 shows the face-recognizing and size change of some samples.

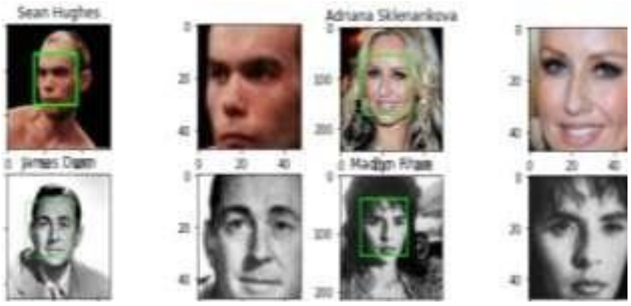


Fig. 9. Recognizing the face in pre-processing.

In the prediction step, the model predicts the gender based on its training data and Adjusting weights, and then puts the gender type with a label on top of the person's face. Fig. 10 shows the complete recognition process for the real-time gender system.

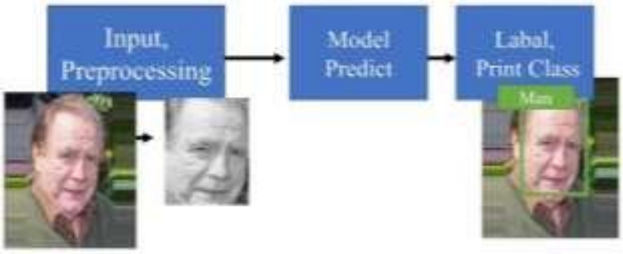


Fig. 10. Real-time gender recognition steps for a test sample from the Wiki dataset.

Fig. 11 shows the results of real-time gender recognition for some randomly selected samples from the Wiki dataset and Figure 12 shows the same for some samples from the Audience dataset.

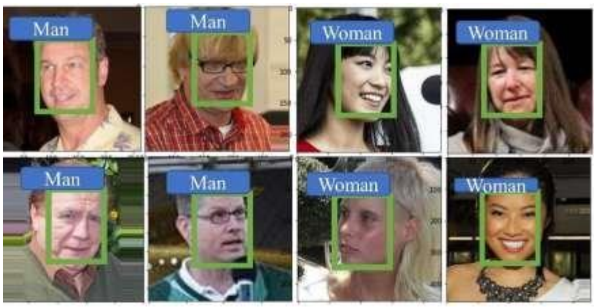


Fig. 11. Real-time gender recognition for some random samples from the Wiki dataset.

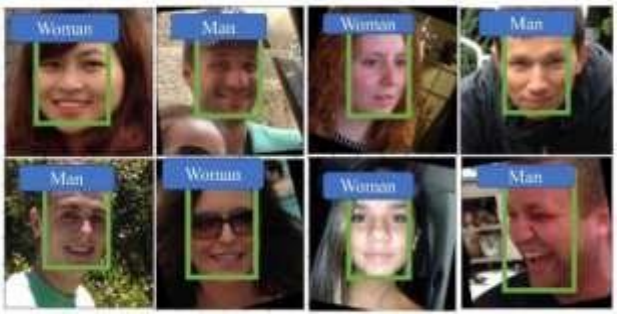


Fig. 12. Real-time gender recognition for some random samples from the Audience dataset.

5. Conclusion

Deep neural networks have undergone considerable progress during the last few decades and have drawn the attention of many researchers in most fields. These systems could be used in real-time if we reduce the number of parameters and calculations. Using Depthwise Separable Convolutions instead of Normal Convolutions is one way of reducing the parameters and calculations of known networks such as VGG, RESnet, Squeeze net, etc. Global Pooling can be used for preventing overfitting and the Full Connected layers can be removed from the end of networks. Filters have a key rule in extracting the features, and using different filters empowers the network to have better learning. Pre-processing the data is effective on the results, in this paper pre-processed datasets performed better than datasets that are not pre-processed. We predict that in the future new deep learning networks will be designed that can work on any hardware which will make these networks even lighter and faster than before, and this will make a considerable amount of researchers interested in real-time systems in deep learning networks.

References

1. Rai P, Khanna P, "Gender classification techniques: A review," in Advances in Intelligent and Soft Computing, 2012.
2. Tivive FHC, Bouzerdoum A, "A gender recognition system using shunting inhibitory convolutional neural networks," in International Joint Conference on Neural Networks, Vancouver, Canada. New York, NY, USA, 2006.
3. Shan S, Mohamed Kh, Feeza R, Rabia B, "Gender Classification: A Convolutional Neural Network Approach," Turkish Journal of Electrical Engineering and Computer Sciences , pp. 248-1264, 2016.

   A. e, "Convolutional Neural Networks," Stanford University, 2016.
4. Haider K, Malik K, Khalid S et al., "Deepgender: real-time gender classification using deep learning for smartphones," in Journal of Real-Time Image Processing, 2019.
5. Kumar S, Singh S, Kumar J, "Gender classification using machine learning with multi-feature method," in 2019 IEEE 9th Annual Computing and Communication Workshop and Conference, 2019.
6. Nejatian A, Sarbishei G, "Implementation real-time gender recognition based on facial features using a hybrid neural network Imperialist Competitive Algorithm," in 2017 25th Iranian Conference on Electrical Engineering, 2017.
7. F. R,"Baoface databaseat the face detection homepage," in Available: http://www.facedetection.com.
8. V.Jain,"Human_Face_Classificationusing_Neural_Networks","http://cbcl.mit.eduisoftwaredatasetslFaceData2.html.
9. Golomb B.A, Lawerence B.T , "SEXNET: A neural Network," in Perception, 1997.
10. D. C. E, "Image Processing Laboratory," in Department of Electrical Engineering, Centro Universitario da FEI, Sao Bernardo do Campo, Sao Paulo, Brazil Phone: +55 (0) II 4353-2910 e-mail: cet@fei .edu.br http://fei.edu.br/- cetlfacedatabase.html.
11. Solina, Franc, et al, "Color-based face detection in the" 15 seconds of," 2003.
12. Lee B, Gilani S, Hassan G et al., "Facial Gender Classification - Analysis using Convolutional Neural Networks," in 2019 Digital Image Computing: Techniques and Applications.
13. "Available:http://www.openu.ac.il/home/hassner/Adience/data.html," 20. [Online].
14. "Available:https://data.vision.ee.ethz.ch/cvl/rrothe/imdb-wiki/," [Online].
15. "Available:http://vis-www.cs.umass.edu/lfw/," [Online].
16. Liu X, Li J, Hu C et al, "Deep convolutional neural networks-based age and gender classification with facial images," in 1st International Conference on Electronics Instrumentation and Information Systems, 2019.
17. "Available: http://www.jdl.ac.cn/peal/index.html," [Online].
18. Hassan K, Ali I, "Age and Gender Classification using Multiple Convolutional Neural Network," in IOP Conference Series: Materials Science and Engineering , 2020.
19. Khan K, Attique M, Syed I et al, "Automatic gender classification through face segmentation," Symmetry, 2019.
20. Glorot X, Bordes A, Bengio Y, "Deep sparse," Proceedings of the Fourteenth International," p. 315–323, 2011.
21. Ioffe S, Szegedy C, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in International Conference on Machine Learning, 2015.
22. Kingma D, Ba J, "Adam: A method for stochastic optimization," in 3rd International Conference on Learning Representations, 2015.

23. "Available:http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html," [Online].
24. Arriaga O, Valdenegro-Toro M, Plöger P, "Real-time Convolutional Neural Networks for Emotion and Gender Classification," in Cornell University Library, 2017.
25. Levi G, Hassner T, "Age and gender classification using convolutional," in Proceedings of the IEEE Conference on Computer, 2015.
26. Lapuschkin S, Binder A, Muller k, Samek W, "Understanding and comparing deep neural networks for age and gender classification," in Proceedings of the IEEE International Conference on Computer V, 2017.
27. Duan M, Li K, Yang C, Li, K., "A hybrid deep learning CNN–ELM for age and gender classification," Neurocomputing, p. 448–461, 2018.
28. Hassner T, Harel S, Paz E, Enbar R, "Effective face frontalization in unconstrained images," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,, Boston,, 2015.