A Comparative Study on Convolutional Neural Network Based Face Recognition

Tanvir Ahmed*, Prangon Das*, Md. Firoj Ali*, Md- Firoz Mahmud†
*Department of Mechatronics Engineering, Rajshahi University of Engineering & Technology, Bangladesh.
†Department of Electrical and Electronic Engineering, Rajshahi University of Engineering & Technology, Bangladesh.

Email: tanvir14ruet@gmail.com; prangon.ruet,mte@gmail.com; eeemfa07@gmail.com; firoz.eee13@gmail.com

***Abstract*—This paper presents a comparative study to recognize faces from a customized dataset of 10 identities of different celebrities using Convolutional Neural Network based models such as AlexNet, VGG16, VGG19 and MobileNet. These pre-trained models previously trained on ImageNet dataset are used with the application of Transfer Learning and Fine Tuning. For our experiment we used Keras API with TensorFlow backend written in Python. The performance analysis includes training, validation, and testing on different images created from original dataset. The validation accuracy of VGG19 model is found better than the other three but MobileNet model showed better test accuracy.**

***Index Terms*—Deep Learning, Neural Network, Convolutional Neural Network, Transfer Learning, Face Recognition**

I. Introduction

Recognition of faces has been a trending topic in the area of computer vision. The fundamental aspects of face recognition are it's broad interdisciplinary like machine vision; biometrics and security; multimedia processing; psychology and neuroscience etc. [1]. For this it has a wide fields of research. Human has been trying effortlessly to achieve more and more accurate results in this field over the era.

In certain circumstances, face recognition has quite some vital points for recommendation over other biometric modalities [2]. It is well accepted, very familiar and easily understandable by people. As a result, it has a large area of applications like identification of criminals, unlocking smartphones and laptops, home access and security, finding missing person, helping blind people, identifying people on social media, disease diagnosis, real time monitoring and management systems etc.

II. Related Works

But recognition of faces has been a challenging task since the very beginning. Convolutional Neural Network (CNN) is a very recent established competent image recognition method which uses local receptive field as neurons in brain, weights sharing and linking information and greatly reduces the training constraints in comparison with other neural networks [3]. CNN became more popular by Alexnet in computer vision by winning the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [4]. In order to achieve higher accuracy in CNN image classification, the development and usage of deeper and complex CNN has become a trend in the research area [5] [6] [7].

In the recent times, face recognition and clustering has been made using facenet in [8]. In [9] video based emotion has been recognised using CNN-RNN and c3D hybrid networks. In [10] two efficient approximations to standard convolutional neural networks: Binary-Weight-Networks and XNOR-Networks have been proposed. In [11] face recognition has been significantly advanced by the emergence of deep learning with VGGNet and GoogLeNet. In [12] a class of efficient models called MobileNets for mobile and embedded vision applications has been presented.

Face recognition is been a very trendy topic of research in the field of deep neural network. The size of the MobileNet model is just 17 megabytes approximately which can easily be applied in low end embedded systems. We wanted to find out how well this small neural network performs in recognizing faces which may be applied in embedded security system. That is what motivates us to research with MobileNet and comparing it with other larger networks for this specific implementation of face recognition. In this work we have contributed by building a new dataset which can be applied for further research. We have also shown how the deep neural network models perform well with a small dataset with data augmentation.

In this paper we have compared between four convolution neural network based models which are AlexNet, VGG16, VGG19 and MobileNet for face recognition of a customized dataset of faces with their training and validation results. Section II describes some related works in this field and section III tells about the dataset collection. After that section IV gives an overview of the models; how we fine-tuned the models for our purpose. Then section V refers to the experiment and section VI shows the result on our dataset. Finally Section VII concludes with summery.

III. Dataset

A. Image Collection

Collection of good dataset has always been a hard task in the field of computer vision. Thus for this paper a customized

dataset has been made of face iamages of 10 celebrities from google image search which consists of 130 images for each identity. For each class the images have been sorted like as 100 images to train, 20 images for validation and 10 images to test. Fig. 1 shows some example of images from our dataset. Our dataset is available publicly at this link: FaceDataset for further research in the future.

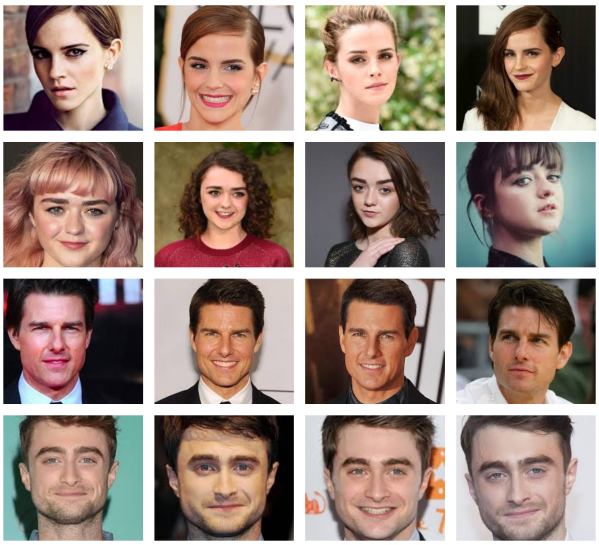

Fig. 1: Example images from our dataset for four identities

B. Data Augmentation

The training and validation images were augmented before feeding to a deep neural network. As deep neural networks need a large amount of data but our dataset is not large enough, hence we performed data augmentation to avoid over-fitting. Augmented data from original images are obtained by applying simple geometric transformations such as translation, rotation, change in scale, horizontal flip etc. The augmented data replace the original training and validation data before going through a neural network model in each epoch and that is how the neural network model is getting different types of images of the same class in each epoch.

IV. Models Overview

A. AlexNet

The architecture of the AlexNet model contains eight learned layers—five convolutional and three fully-connected. The output of the final fully-connected layer is nourished to a 1000-way softmax which in our case is reduced to 10. The network maximizes the multinomial logistic regression objective, which is identical to maximizing the normal over training cases of the log-probability of the right label under the prediction conveyance. The kernels of the second, fourth, and fifth convolutional layers are connected only to those kernel maps in the previous layer which reside on the same GPU. The kernels of the third convolutional layer are connected to all kernel maps in the second layer. The neurons in the fully connected layers are connected to all neurons in the previous layer. Response-normalization layers follow the first and second convolutional layers. Max-pooling layers follow both response-normalization layers as well as the fifth convolutional layer. The ReLU non-linearity is applied to the output of every convolutional and fully-connected layer.

B. VGG

The VGG16 architecture consists of 12 convolutional layers, some of which are followed by maximum pooling layers and then 4 fully-connected layers and finally a 1000-way softmax classifier. In this paper we have eliminated the 2 fully connected layer and the classification layer at the output end and added a single fully connected layer with ReLU activation and finally a classification layer of 10 neurons with softmax classification. We have not updated the weights learned from the ImageNet dataset for the first 14 layers of VGG16 model. So, the model does not train its first 14 layers and that is how we fine tuned the VGG16 model for our purpose and applied Transfer Learning. Classification for 10 classes reduces the total number of trainable parameters from 138,357,544 of original model to 13,504,778 for the fine-tuned model. Adam optimizer is also used for VGG16 model with the same learning rate as MobileNet which is 0.0001. Similarly VGG-19 is applied with 14 layers pre-trained and reduced weights.

C. MobileNet

MobileNets are based on a streamlined architecture which builds light weight deep neural networks by using depth wise seperable convolutions. Two simple global hyper parameters that efficiently trade off between latency and accuracy are introduced [13]. The MobileNet model is based on depthwise separable convolutions which may be a frame of factorized convolutions which factorize a standard convolution into a depthwise convolution and a 1×1 convolution called a point-wise convolution. For MobileNets the depthwise convolution applies a single filter to each input channel. The pointwise convolution at that point applies a 1×1 convolution to combine the outputs of the depthwise convolution. The first MobileNet model has 28 layers. In this paper we eliminated the last fully connected layer; the classification layer which was built to classify 1000 classes of ImageNet dataset and again added a fully connected layer of 10 classes to classify our data. That reduces the number of trainable parameters from 4,231,976 of original model to 3,217,226 and the non-trainable parameters still remain the same; 21,888. We have trained all the layers from scratch. Adam [14] optimization which is an adaptive learning rate optimization algorithm designed specifically for training deep neural networks is applied with a learning rate of 0.0001.

V. Experiment

For our experiment we used Keras [15] with TensorFlow backend. Keras is a high-level neural networks API, written in Python which provides a huge amount of functions and models regarding neural networks and image processing. It was developed with a focus on enabling fast experimentation by providing easy functions to build a customized neural network model as well as enabling user to apply and customize pre-trained models. The models we used were trained and tested in Google Colaboratory which a research tool for machine learning education to get GPU support in order to reduce training time. In our experiment MobileNet model worked excellently with a 100% training accuracy achieved on 5th epoch where VGG16 got its full accuracy on 46th epoch. Alexnet was a bit short to achieve 100% training accuracy. There were 100 epochs in total to train and validate the dataset. Fig. 9 shows both the training loss and validation loss for MobileNet were reducing in a smooth manner which means the model was not over-fitting. On the other hand, Fig. 7 shows VGG16 had fluctuation in both training loss and validation loss. Alexnet performed very poorly in validation shown in Fig. 2 and Fig. 3. The simplicity in Alexnet architecture might be the reason behind it. VGG19 shows the best validation accuracy, the performance of which is shown in Fig. 6 and Fig. 7. We have tested the models with 100 images of 10 classes having 10 images for each class. All the images are different from the training and validation images. Fig. 13 shows MobileNet predicted 84 images correctly where which is the highest among others.

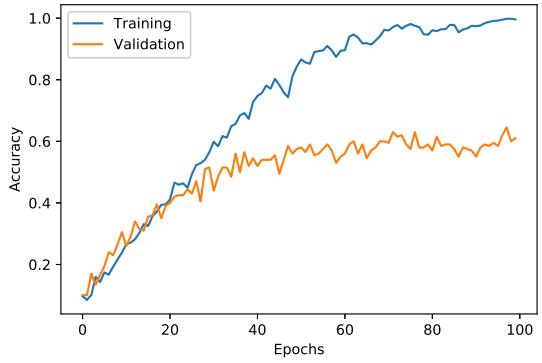

Fig. 2: AlexNet Training and Validation Accuracy

We have tested the models with 100 images of 10 classes having 10 images for each class. All the images are different from the training and validation images. Fig. 13 shows MobileNet predicted 84 images correctly where VGG16 predicted 71 images to be correct shown in Fig. 12. Both the models struggled most to predict the images of Emilia Clarke.

VI. Results

The final result of our fine-tuned models is summarized in Table 1 shows that both the models perform with a

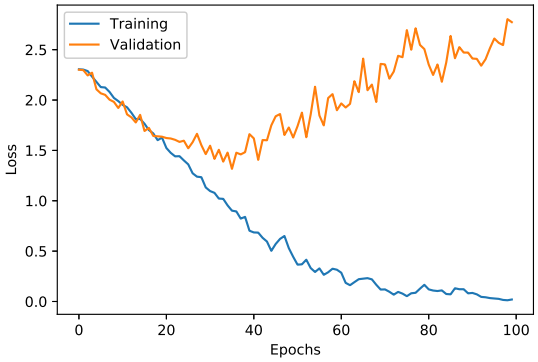

Fig. 3: AlexNet Training and Validation Loss

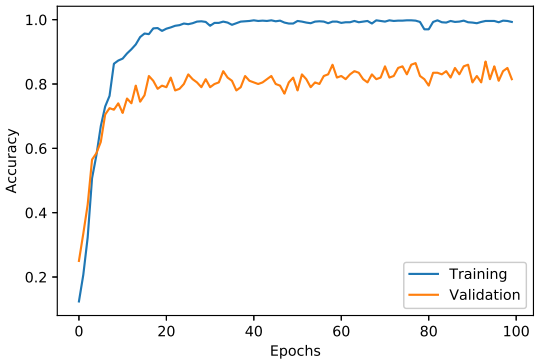

Fig. 4: VGG19 Training and Validation Accuracy

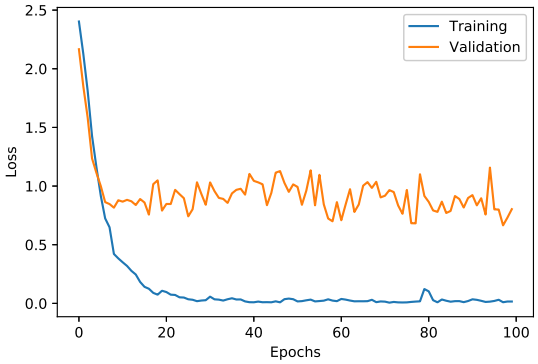

Fig. 5: VGG19 Training and Validation Loss

TABLE I: Comparison Between the Models

Model Name	Maximum Training Accuracy	Maximum Validation Accuracy	Test Accuracy
AlexNet	99.8%	64.5%	57%
VGG16	100%	86%	71%
VGG19	99.8%	87%	56%
MobileNet	100%	85%	84%

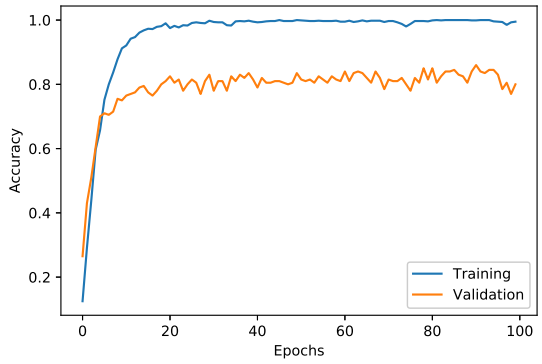

Fig. 6: VGG16 Training and Validation Accuracy

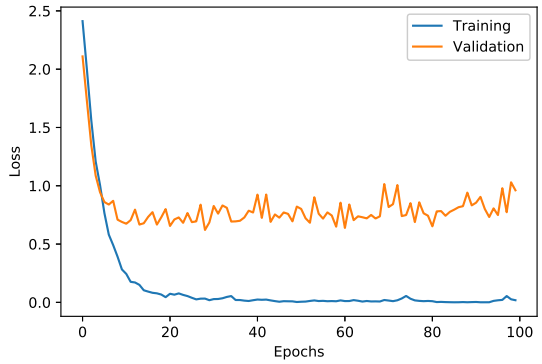

Fig. 7: VGG16 Training and Validation Loss

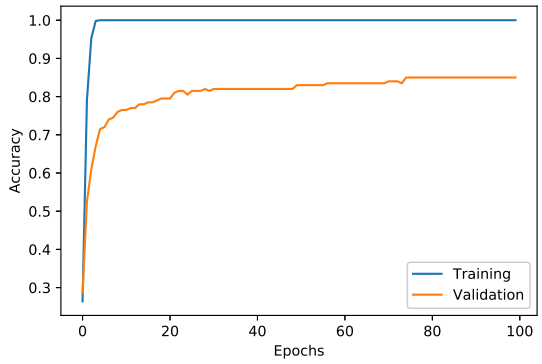

Fig. 8: MobileNet Training and Validation Accuracy

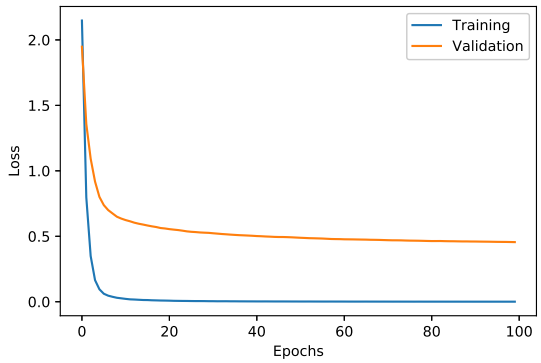

Fig. 9: MobileNet Training and Validation Loss

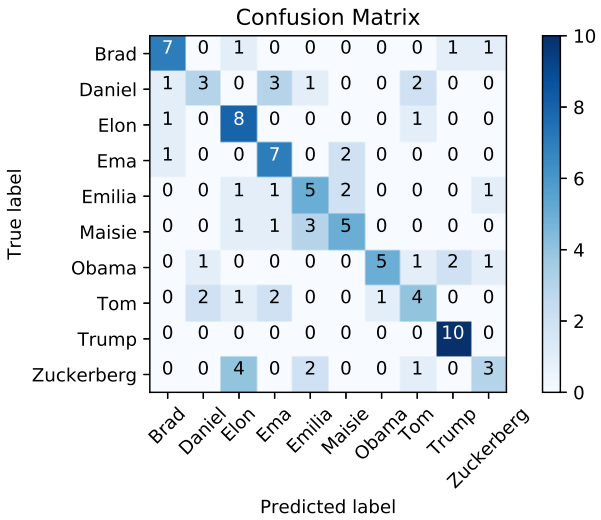

Fig. 10: Confusion Matrix for AlexNet on Test Dataset

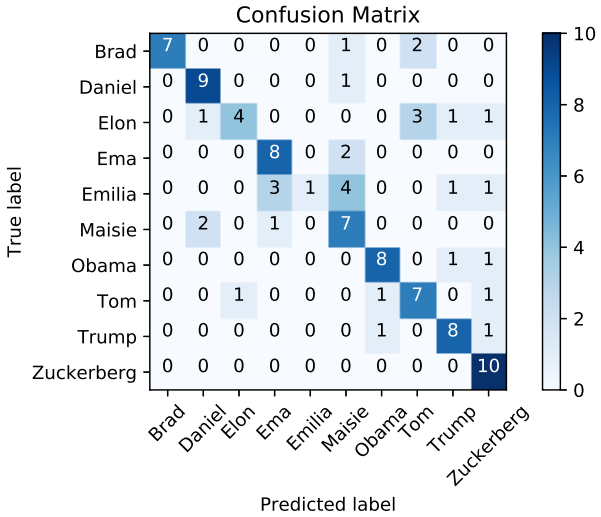

Fig. 11: Confusion Matrix for VGG16 on Test Dataset

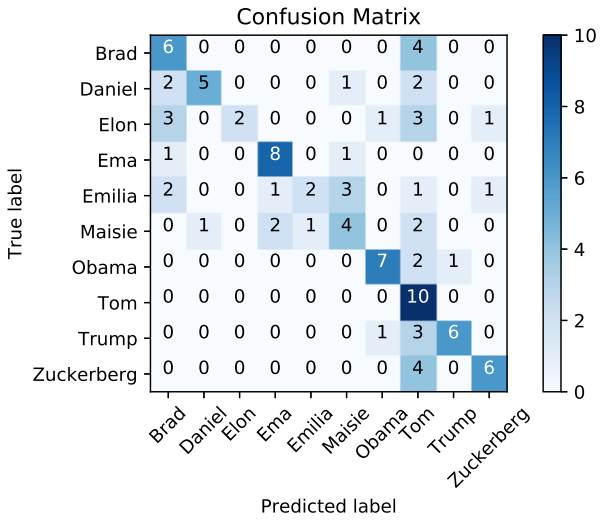

Fig. 12: Confusion Matrix for VGG19 on Test Dataset

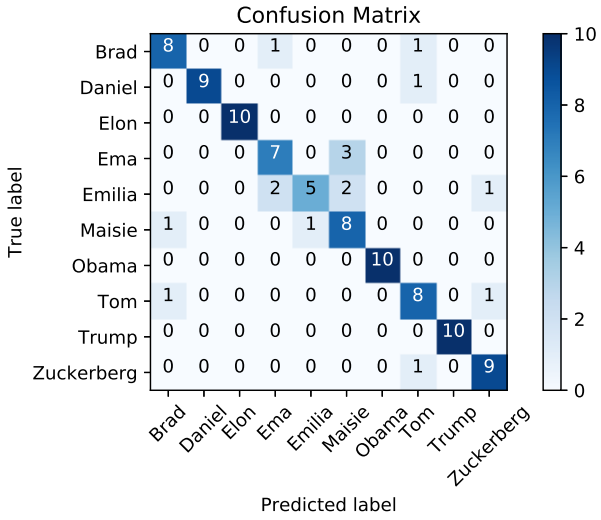

Fig. 13: Confusion Matrix for MobileNet on Test Dataset

good validation accuracy but in case of testing, the result of VGG16 model was not satisfactory. Although we have trained AlexNet and MobileNet model from scratch with no pre-trained weights, MobileNet shows better accuracy than other three models in testing where pre-trained weights achieved from ImageNet dataset were not updated for first 14 layers. But in comparison among four VGG19 performed better with best validation accuracy about 87%.

VII. Conclusion

Collecting of dataset was a challenging part and a task of patience in our work. We collected the images that have a better view of subjects' faces. Both the MobileNet model and VGG16 model were fine-tuned in order to train our dataset. The MobileNet model is very smaller in size comparing the other three. Undoubtedly MobileNet shows a better test accurate performance but comparing with the other three VGG19 showed best performance in validation accuracy. Our dataset is made available publicly so that anyone can do further research with it. Other versions of these models such as MobileNet v2 or other convolutional models can also be applied. Also other deep convolutional neural network based models such as ResNet, Inception can be applied with different versions in further research.

Appendix

Abbreviations and Acronyms

CNN	Convolutional Neural Network
$ReLU$	Rectified Linear Unit
GPU	Graphical Processing Unit
RNN	Recurrent Neural Network
API	Application Programming Interface

References

[1] A. W. Senior and R. M. Bolle, "Face recognition and its application," in *Biometric solutions*. Springer, 2002, pp. 83–97.

[2] R. Chellappa, C. L. Wilson, S. Sirohey *et al.*, "Human and machine recognition of faces: A survey," *Proceedings of the IEEE*, vol. 83, no. 5, pp. 705–740, 1995.

[3] N. R. Gavai, Y. A. Jakhade, S. A. Tribhuvan, and R. Bhattad, "Mobilenets for flower classification using tensorflow," in *2017 International Conference on Big Data, IoT and Data Science (BID)*. IEEE, 2017, pp. 154–158.

[4] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[6] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[7] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[8] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[9] Y. Fan, X. Lu, D. Li, and Y. Liu, "Video-based emotion recognition using cnn-rnn and c3d hybrid networks," in *Proceedings of the 18th ACM International Conference on Multimodal Interaction*. ACM, 2016, pp. 445–450.

[10] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "Xnor-net: Imagenet classification using binary convolutional neural networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 525–542.

[11] Y. Sun, D. Liang, X. Wang, and X. Tang, "Deepid3: Face recognition with very deep neural networks," *arXiv preprint arXiv:1502.00873*, 2015.

[12] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[13] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, B. C. Van Esesn, A. A. S. Awwal, and V. K. Asari, "The history began from alexnet: A comprehensive survey on deep learning approaches," *arXiv preprint arXiv:1803.01164*, 2018.

[14] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[15] F. Chollet *et al.*, "Keras," https://keras.io, 2015.