Realtime Face-Detection and Emotion Recognition Using MTCNN

Muhammad Azhar Shah

June 29, 2020

Realtime Face-Detection and Emotion Recognition Using MTCNN

M.Azhar
Ripha International University
Lahore

Abstract— In this paper, the problem of facial expression is addressed, which contains two different stages: 1. Face detection, 2. Emotion Recognition. For the first stage, an MTCNN (Multi-Task Convolutional Neural Network) has been employed to accurately detect the boundaries of the face, with minimum residual margins. The second stage, leverages a ShuffleNet V2 architecture which can tradeoff between the accuracy and the speed of model running, based on the users' conditions. The experimental results clearly Shows that our proposed model outperforms the state-of-the-art on FER 2013 dataset which has been provided by Kaggle.

Keywords—Image Precessing, Computer Vision, Deep Convolution Neural Network, Face Detection, Facial Emotion Recognition

I. Introduction

Considering the recent advancements in the area of Artificial Intelligence and image processing and understanding technologies, media producers and broadcasters are gaining so much interest in evaluating their customers with respect to the contents or their reactions against programs. This will help them improve their programs and provide the contents as per their customers' will. This task, however, needs some knowledge of cognitive evaluation of people. One way to tackle such a difficult problem, is to evaluate their facial expressions against what they encounter, from media contents up to interviews, and so on. For example, in a live TV program, the TV show host can adaptively change the questions based upon the emotional reaction of the guest, or he can change the questions so that he is sure the guest will not react aggressively or uncertainly to the impending questions.

What we mean by facial expression, is to extract the emotional status of a person, as a reaction to some events through image processing tasks. Fortunately, for most people in the world, the indications being demonstrated as a result of emotional reactions to the events, are identical irrespective of their culture and geography. That is why Ekman et al. [1] classified the emotional faces around the world through six different statuses, including frightened, happiness, sadness, aggression, disgust and amazed. All these emotional states are differentiable from the neutral status. They could also figure out, that a person may carry a combination of these emotions in the face, at the same time.

Surveillance strategies are apt to be capable of recognizing feelings, including facial expressions. In this newsletter, we integrate face detection and emotion recognition tasks to create a unified system that can carry out facial popularity and emotion popularity duties in real time. To do that, we use M, MTCN (Multi Task Cascaded Convolutional Neural Network) to take benefit of the capability of this community [2,3] for joint detection and alignment of faces. , And integrate it with a Shuffle NetV2 structure [4,5] to make the most the capability for inherent tradeoff between output accuracy and speed. The reason of this mixture is to produce facial expressions in a sensible fashion..

The outline of this paper appears, as the following. Next section, brings the theoretical explanation over the MTCNN and ShuffleNet architectures, as well as the intuition behind the chosen architectures to tackle the problem. In section III, the experimental setup and scenario clarifies different practical aspects of the work and detailed information about how to implement the system. The paper, then is terminated with a conclusion and the references being cited through the course of the aforementioned subjects.

II. Neural Network Architectures

A. MTCNN Network Architecture and Applications

The cascade Haar-like face detector (aka Viola-Jones) [6], while being prevalent for a decade in face detection tasks, may degrade significantly in applications with larger visual variations of faces which happens in real-world applications. Inspired by the success achieved by using deep convolutional neural networks (CNNs) in computer vision tasks, several studies were motivated to use this architecture for face detection. In this regard, Zhang et al. [2] proposed a new architecture, in which face detection and face alignment tasks are both addressed, in a unified structure. They also concerned about the online implementation of this architecture. thus, The produced three -stages network structure was proposed which perform in a coarse-to-fine fashion, as: (1) The first shallow CNN architecture aims at producing the candidate windows for the face locations in an input image, (2) the second CNN, refines the windows by rejecting a large number of non-faces windows, and (3) a more powerful CNN to refine the results again and output five facial landmarks positions [2]. These three CNNs are named as P -Net (Proposal Network), R-Net (Refinement Network), and O-Net (Output Network), respectively.

B. ShuffleNet Architecture and Applications

Usual CNN architectures contain several convolutional layers and hundreds of channels to achieve a reasonable result. ShuffleNet is an extremely efficient CNN architecture in computational sense [5]. It allows more feature map channels, therefore it helps to encode more information, which is critical to the performance of very small networks. This architecture involves two new operations, namely pointwise group convolution and channel shuffle. The group convolution aims at distributing the convolutions over different GPUs for the sake of parallel separable convolution operations, as it was used in other architectures such as ResNeXt [7,8,11], and DeepRoots [10], and Xception [9]. These modern CNN architectures introduce group convolutions into the building blocks to make a tradeoff between representation capability and computational cost. In these designs, the 1×1 convolutions (aka pointwise convolutions) burden considerable complexity. The pointwise group convolution reduces the computation complexity of the 1×1 convolutions. In tiny networks, expensive pointwise convolutions result in limited number of channels to meet the complexity constraint, which might significantly damage the accuracy [15]. A solution to this problem would be to apply the channel sparse connections, for example group convolutions even for 1×1 layers.

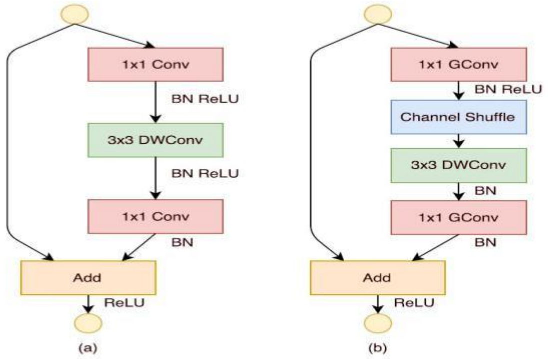


Figure 1. ShuffleNet units: (a) bottleneck unit with depth wise convolution, (b) ShuffleNet unit with pointwise group convolution and channel shuffle. [5]

As in Fig 1(a), it depicts a bottleneck unit, which is a residual block with a branch. For the 3×3 layer, a 3×3 depth-wise convolution has been applied on the bottleneck feature map. Then, as in Fig 1(b), the first 1×1 layer has been replaced with pointwise group convolution followed by a channel shuffle operation to form a ShuffleNet unit. The second pointwise group convolution is used to match the channel dimensions with the shortcut path. The computational cost of ShuffleNet is much less than ResNeXt or Xception. In other words, given the same computational budget, ShuffleNet can use wider feature maps, hence provides a better accuracy by manipulating more information out of the input data [5].

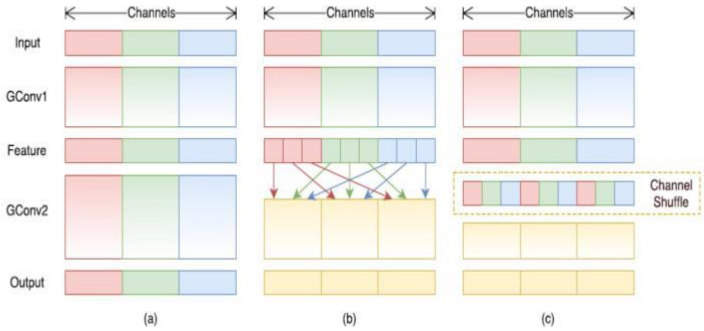


Figure 2. 1×1 grouped convolution with channel shuffling [5]

III. Experimental Setup and Results

The dataset being used in this work is FER2013 [12], which has been taken from Kaggle. The input images are in grayscale with 48×48 Pixels resolution. From this dataset, 28,709 images are used for training the networks, and 3,589 images for testing the results. These images belong to one of the seven emotional classes of {“angry”, “disgust”, “fear”, “happy”, “sad”, “surprise”, “neutral”}. About 20% of the total images are considered for the validation phase. A sample from this dataset is presented in Fig 3, below.

The hardware used for the experiments, was a laptop with CPU 4700MQ, Core i7- 2.4GHz, with 8 GB RAM.

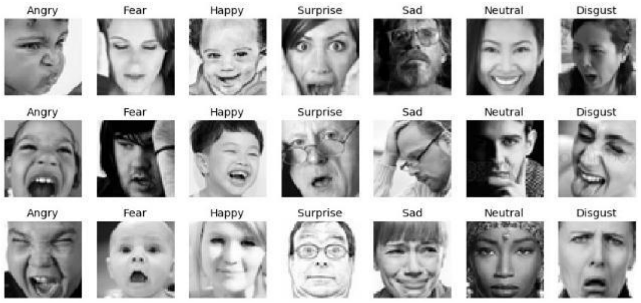


Figure 3. Sample images from the FER2013 dataset

Due to the asymmetric distribution of data for different classes of emotions, we performed five different augmentation types on each sample of data, to create a normalized version of data, as in Fig 4.

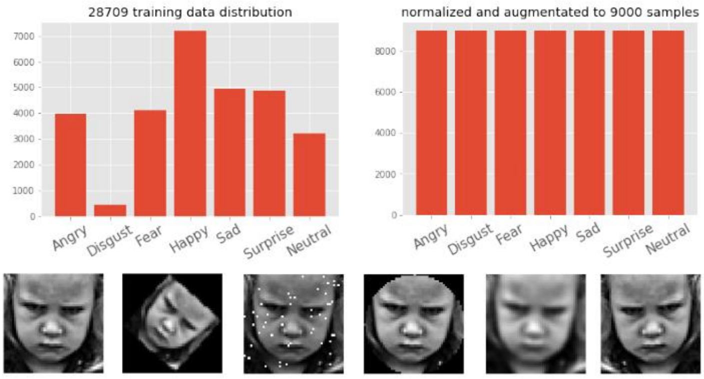


Figure 4. (left-up) Distribution of the samples per class; (right-up). Normalized data samples using augmentations. (Bottom). Various augmentation types containing: Main sample, Rotated, Noisy, Grid, Blurred, Flipped, respectively from left-to-right.

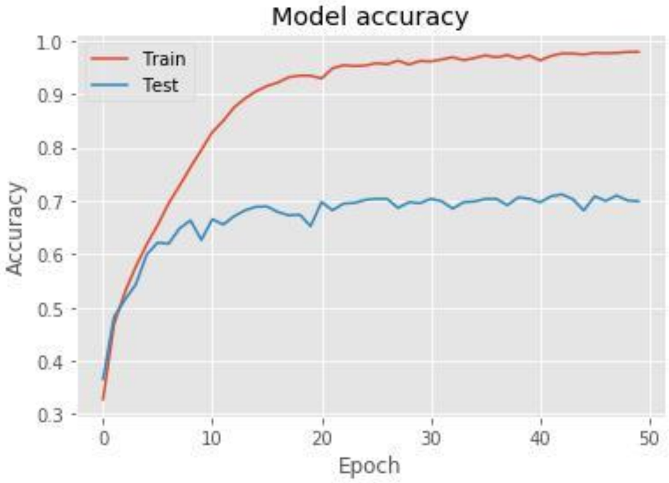


Figure 5. The achieved accuracy of our proposed model (miniShuffleNet V2), using 50 epochs. The final accuracy we could achieve for the test data was 71.19%, to our knowledge is the best achieved, so far. The model parameters could be stored in 4.6Mbytes, which could be easily exploited in a realtime fashion.

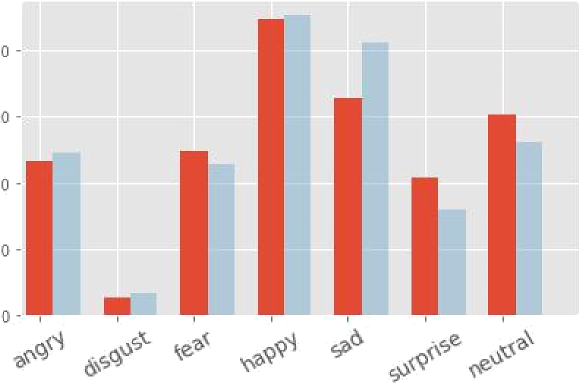
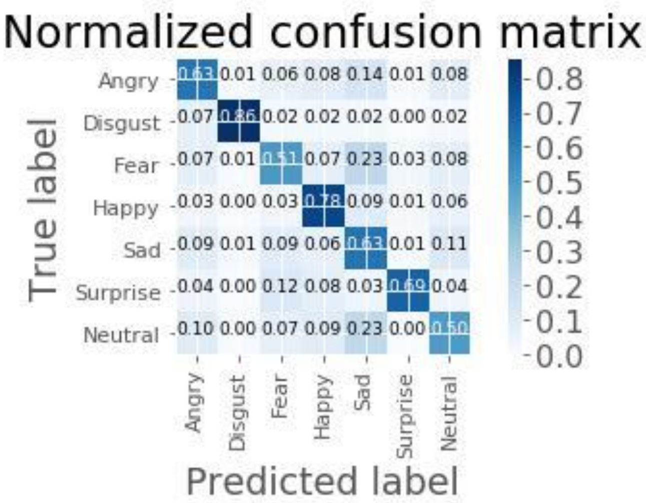


Figure 6. Red-bars depict the real output labels; Blue-bars, are the predicted labels. The predictions are very close to the real values.



	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry	0.63	0.01	0.06	0.08	0.14	0.01	0.08
Disgust	0.07	0.86	0.02	0.02	0.02	0.00	0.02
Fear	0.07	0.01	0.51	0.07	0.23	0.03	0.08
Happy	0.03	0.00	0.03	0.78	0.09	0.01	0.06
Sad	0.09	0.01	0.09	0.06	0.63	0.01	0.11
Surprise	0.04	0.00	0.12	0.08	0.03	0.69	0.04
Neutral	0.10	0.00	0.07	0.09	0.23	0.00	0.50

Figure 7. The normalized confusion matrix. The output clearly states that our model can recognize the emotions effectively, specifically for the disgust case the result was remarkable compared to the state-of-the-art.

The accuracy of our model during training and test, are depicted in Fig 5. The achieved result was 71.19%, which is the best achieved so far, to our knowledge. Fig 6, further emphasizes on our model prediction capability. In Fig 7, the normalized confusion matrix clearly shows that our model outperforms the state-of-the-art, specifically for disgust emotion, which could be expected, in advance, due to the augmentation of data and normalizing the distribution of samples over different emotional states.

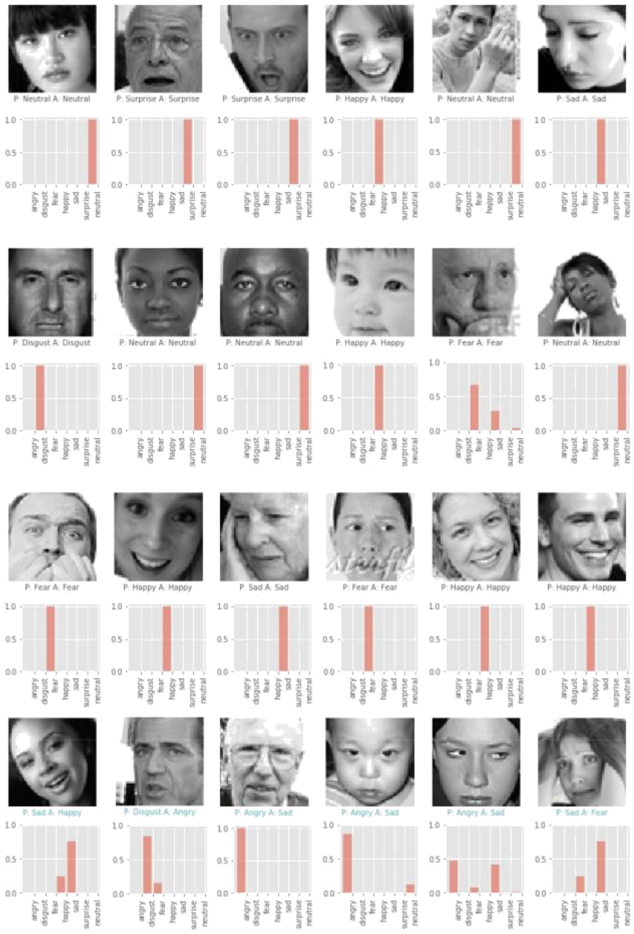


Figure 8. Upper three rows depict the correct recognition samples, and their distributions over different classes. The bottom row, depicts the misclassification case.

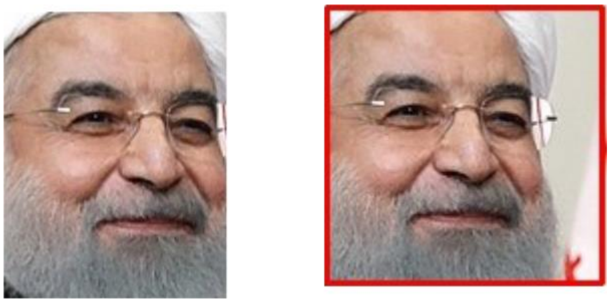


Figure 9. (left) MTCNN face boundary versus (right) Haar-cascade boundary detection. Boundary detections could directly impact on the recognition phase, as it is shown in the next figure, on our examples.

Fig 8, depicts some correctly classified examples, along with their distribution predicted over classes. The bottom row of Fig 8, is the misclassified case. Fig 9, and 10, together explain that how much the recognition stage is dependent on the correct boundaries, being detected. A misclassification instance has been happened for Mr. Rouhani, due to a wrong boundary detection of Haar-cascade algorithm. However, this problem has not happened when using our MTCNN-based model. The complete model and settings is included in Fig 11.

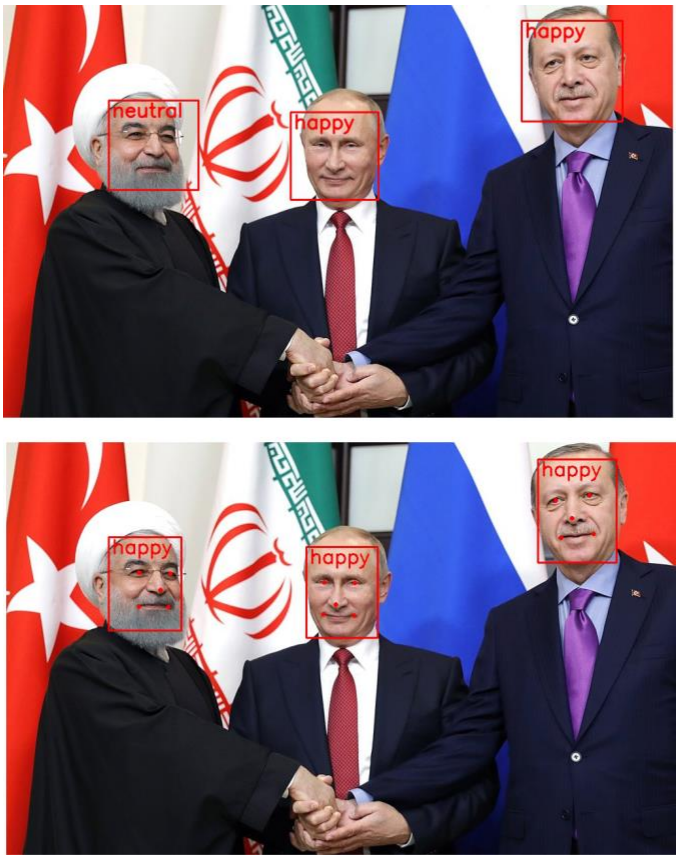


Figure 10. The result of facial expression shown for three presidents. Due to the wrong boundary detections in the upper figure, Mr. Rouhani has been wrongly classified to be neutral. However, as depicted in the bottom figure, our MTCNN-ShuffleNet model correctly classifies all facial expressions, including Mr. Rouhani.

Conclusion

In this paper, a unified architecture is proposed which integrates two different CNN (Convolutional Neural Network) based modules. This combined model tries to benefit from the advantages of each single module individually. An MTCNN, is used for correctly cropping the face boundary, a ShuffleNet V2 is exploited to recognize the emotions using the optimum depth which could be used in realtime. This combination has been proved to be efficient in real-world evaluation.

Acknowledgement

The authors of this paper, acknowledge Satyajit Sarangi for creating libraries which we could employ for extracting our figures. Moreover, we express our thanks to Octavio Arriaga for patiently answering our questions during our implementations.

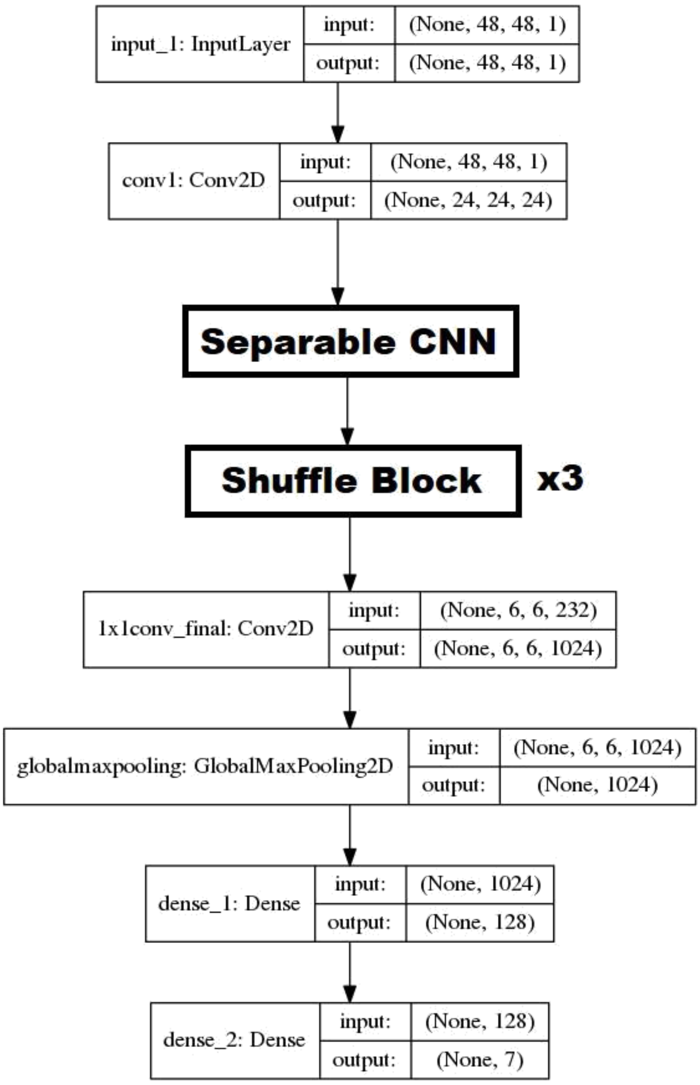


Figure 11. The complete model for the facial expression task [14]

REFERENCES

[1] Ekman, Paul, and Wallace V. Friesen. "Constants across cultures in the face and emotion." Journal of personality and social psychology, 17.2 (1971): 124.

[2] Zhang K, Zhang Z, Li Z, Qiao Y. “Joint face detection and alignment using multitask cascaded convolutional networks”. IEEE Signal Proc-essing Letters. 2016 Oct;23(10):1499-503.

[3] Abdulnabi AH, Wang G, Lu J, Jia K. “Multi-task CNN model for attribute prediction”. IEEE Transactions on Multimedia. 2015, Nov; 17(11):1949-59.

[4] Ma, Ningning, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design." In Proceedings of the European Conference on Computer Vision (ECCV), pp. 116-131. 2018.

[5] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, Jian Sun, “ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices” , Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6848-6856, June, 2018.

[6] Viola, Paul, and Michael Jones. "Rapid object detection using a boosted cascade of simple features." In Computer Vision and Pattern Recognit-ion, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, vol. 1, pp. I-I. IEEE, 2001.

[7] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778. 2016.

[8] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Identity mappings in deep residual networks." In European conference on comp-uter vision, pp. 630-645. Springer, Cham, 2016.

[9] Chollet, François. "Xception: Deep learning with depthwise separable convolutions." arXiv preprint (2017): 1610-02357.

[10] Ioannou, Yani, Duncan Robertson, Roberto Cipolla, and Antonio Criminisi. "Deep roots: Improving CNN efficiency with hierarchical filter groups." (2017).

[11] Xie, Saining, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. "Aggregated residual transformations for deep neural networks." In Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on, pp. 5987-5995. IEEE, 2017.

[12] Goodfellow, Ian J., Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski et al. "Challenges in repres-entation learning: A report on three machine learning contests.", In Int-ernational Conference on Neural Information Processing, pp. 117-124. Springer, Berlin, Heidelberg, 2013.

[13] Arriaga, Octavio, Matias Valdenegro-Toro, and Paul Plöger. "Real-time Convolutional Neural Networks for Emotion and Gender Classification. " arXiv preprint arXiv:1710.07557(2017)

[14] Köpüklü, Okan, Maryam Babaee, and Gerhard Rigoll. "Convolutional Neural Networks with Layer Reuse." arXiv preprint arXiv:1901.09615 (2019).

[15] Lyu, Jiancheng, et al. "AutoShuffleNet: Learning Permutation Matrices via an Exact Lipschitz Continuous Penalty in Deep Convolutional Neural Networks." arXiv preprint arXiv:1901.08624 (2019).