International Journal of Engineering Research and Technology.
ISSN 0974-3154 Volume 11, Number 7 (2018), pp. 1037-1049

http://www.irphouse.com

Improved Joint Cascaded CNN for Face Detection

Uwinema Joseline

Department of Software Engineering,
Kumoh National Institute of Technology,
61 Daehak-ro, Gumi, Gyeongbuk, South Korea.

Hae-Yeoun Lee

Department of Computer Software Engineering,
Kumoh National Institute of Technology,
61 Daehak-ro, Gumi, Gyeongbuk, South Korea.

Abstract

Many face detection algorithms result in high quality classification at the expense of high computational complexity. Cascade network has been used where classifier with low computation cost can be firstly used to shrink background while keeping the faces. Then, the cascade in detection was used in subsequent works of other pipelines, such as DPM and CNN. This paper presents an improved joint cascade CNN method for face detection. The cascade CNN is designed in similar to that of Li et al, which includes three stages of detection in cascade and a calibration stage is brought in each of the three detection stages. The output of each calibration stage is used to adjust the position of detection window position for input to the subsequent stage. The key idea is improving the accuracy by analyzing the existing C-CNN structures and applying various optimization such as augmenting data, optimizing model and its parameters, and adjusting drop-out. Experiments are performed using Google TensorFlow to verify the advantages of the joint training and calibration by comparing with other face detection methods. Our proposed method has achieved accurate performance in terms of detection and false-positive rates.

Keywords: Face Detection, Neural Networks, Cascade CNN, GPU, TensorFlow.

I. INTRODUCTION

Face detection in images or video streams is a classical problem of computer vision.

Accurate and efficient detected face is related to the performance of various face-based applications such as face identification, face clustering, tagging and retrieval [1, 2, 3]. However, because of various unexpected conditions such as pose, presence or absence of structural components, facial expression, orientation or imaging conditions, face detection is very difficult due to false rejection. Therefore, many works in face detection focus on faces under more wild conditions, category level, variations in subject level because the simple features like Harr in the cascade training are insufficient to capture the more complex face variations.

Many algorithms have been developed for face detection such as Viola Jones (Haar features, integral image, Adaboost, Cascading), Neural Networks, and Support Vector Machine. These algorithms have been delivered to real applications such as automatic target recognition or generic object detection and recognition, video surveillance, human computer interface, video conference and biometric applications. But as mentioned above that faces can be in different various conditions, the single model in Viola-jones framework cannot handle those faces from different poses, neural network concept with efficient cascade structures was proposed.

The neural networks concept including different networks like convolutional neural networks (CNN), hope field networks, recurrent neural networks, long short term neural networks, and fully connected neural networks are mostly trained with an algorithm known as back-propagation as mentioned in [4]. In the late eighties, the neural networks for machine learning was fully connected neural networks and they have a large big number of weights parameters. By contrast, CNN is a neural network in which a single neuron is connected to only a few neurons in the previous layer and shares weights. Neural network, recently called deep learning has been a successful topic just because of the success of CNN.

CNN is one of the most attractive research area since it has various applications including face detection and recognition. Cascaded CNN is a well going model on image related fields where in terms of accuracy they beat other related models in the field. The main advantage of Cascaded CNN is that without any human interactions, important features can be detected. Researches thus have attempted to deeply study successful deep learning methods and techniques for the task of object detection [5]. In object detection, one of the most important and highly successful framework is the region-based CNN (RCNN) method [5, 6].

In this paper, we present an improved joint cascade CNN method for face detection. In similarly to the cascade model of Li et al, we design three stages of detection and a CNN calibration stage is brought in each of the three detection stages in the cascade. Especially, we improve the accuracy of cascade CNN by analyzing the existing C-CNN structures and applying various optimization such as augmenting data, optimizing model and its parameters, and adjusting drop-out.

The paper is composed of as follows. In Sec. II, related works are presented and the proposed method is explained in Sec. III. Experimental results are shown in Sec. IV and Sec. V concludes.

II. RELATED WORKS

II.I CNN and Cascades for Object Detection

Before the emergence of CNN, the deformable part model has been the state-of-the-art object detector for years [5]. With its image classification capabilities, CNN has been applied to object detection including face detection, and has achieved promising results.

For face detection, Vaillant et al showed that CNN is used in a sliding window manner to traverse different locations and scales and classify faces from the background [6]. Also, they proposed in their work to identify the presence or absence of a face in a given image window by training a CNN and scan that whole image at all possible locations with the network.

Rowley et al used CNN for frontal face detection and showed quite good performance [7]. Their method is the CNN based method but is quite similar to the modern CNN methods. Also, it can be used on easy datasets and get relatively good performance.

There are approaches to eliminate candidates by combining a series of simple features. Sun et al. combined networks focusing on different facial parts for facial point detection [8]. Li et al. used a shallow detection network with small scale input images to first reject easy non-face samples and then apply two deeper networks to eliminate more negatives while maintaining a high recall [4]. During the process, a calibration network is appended for bounding box calibration after each detection network. Recently, a combination of a tiny deep network and a modified AlexNet is followed to achieve real time pedestrian detection. That tiny deep network is applied to remove a large number of candidates and leaves in action a reliable size of candidates for evaluation in a large network.

II.II Cascaded Networks

For face detection, a number of algorithms using cascaded stages have been widely proposed because of the advantages of handling unbalanced distribution of negative and positive samples. There has also been significant progress in recent years and many people have used Adaboost, SVM or random forest classifiers. Component detectors uses local images features, but for cascaded networks, week classifiers can reject most false negatives in the early stages and stronger classifiers can save computation with less candidates in later stages.

Multi-stage CNNs has gained popularity because of the development of deep CNNs. These mechanisms are adopted in state-of-the-art object detection algorithms such as cascaded CNNs [4] and faster R-CNN [9], where the first stage is a network for candidate region generation and other following stages are networks for detection. These deep CNNs can be optimized jointly using back-propagation for optimization other than boosting methods. With this optimization, layers of different networks can be jointly optimized, which helps to share computation and information.

II.III Cascaded Convolutional Neural Network (CNN)

Initially, it starts with a CNN with an input layer, hidden layers, and an output layer with neurons, but by combining it or adding a calibration process, it becomes a cascaded CNN or, in other words, cascaded jointly CNN.

III. PROPOSED JOINT CASCADE CNN MODEL FOR FACE DETECTION

An effective face detection algorithm demands an advanced discriminative model to accurately differentiate faces from the background as well as face vs non-face binary classification [10]. A single CNN model expresses a single differentiable score function from the raw image pixels to class scores of both sides but has performance limitation for face detection [10], [11]. In this paper, cascaded CNN (C-CNN) is considered to overcome the disadvantages of convolutional methods [3]. The main point is to combine training and calibration to learn both tasks together in the same cascade framework to minimize the number of candidates at a later stage. It also improves the accuracy by analyzing the existing C-CNN structures and applying various simulations of the algorithm.

III.I Overall Framework

A neural networks takes input and transforms it into a series of hidden layers. Each hidden layer consists of a set of neurons in which all neurons are completely connected to neurons of the previous layer. The output layer is the last fully connected layer and it symbolizes the class scores in classification settings [12].

The cascaded CNN operates at multiple resolutions, in which the background regions are rejected in the low resolution stage and the small number of challenging candidates is precisely calculated at the last high resolution stage [12] [13]. After each of the detection stages in this structure, the CNN based calibration stage is applied at later phases to increase localization effectiveness and decrease the number of candidates [13].

III.II C-CNN Structure of Proposed Method

As shown in Fig. 1, the C-CNN structure of the proposed method is designed by referring the model of Li et al [14]. The C-CNN is composed by six CNNs whereby, three CNNs such as 12-net, 24-net, and 48-net are for face and non-face binary classification and three CNNs such as 12-calibration-net, 24-calibration-net, and 48-calibration-net are for bounding box calibration. In this way, AlexNet is used to apply ReLU after the pooling layer and fully-connected layer [15]. Unlike regular neural networks, the layers of CNN have neurons in 3 dimensions of width x height x depth.

Cascaded CNN has a variety of settings for accuracy computation tradeoffs [15]. Considering the test image, the 12-net scans images across various scales to rapidly discard to the tune close to 93% of the detection windows or more. The left detection

windows are processed by the 12-calibration-net as 12x12 images one after the other in order to regulate its size and location to target any potential face as close as possible.

12-net densely scans an image of size W x H at 4-pixel spacing for 12x12 detection windows, which is equivalent to apply the 12-net to the whole image to obtain a map of confidence scores. Detection windows for 12_net are obtained by applying (1).

$$\left(\left\lfloor\frac{W-12}{4}\right\rfloor+1\right)\text{x}\left(\left\lfloor\frac{H-12}{4}\right\rfloor+1\right) \tag{1}$$

In order to cover faces at different scales, the test image will first be built into an image pyramid. When the minimum size of the face is F, every stage in the image pyramid is resized by 12/F as the input image in the 12-net as shown in Fig. 2.

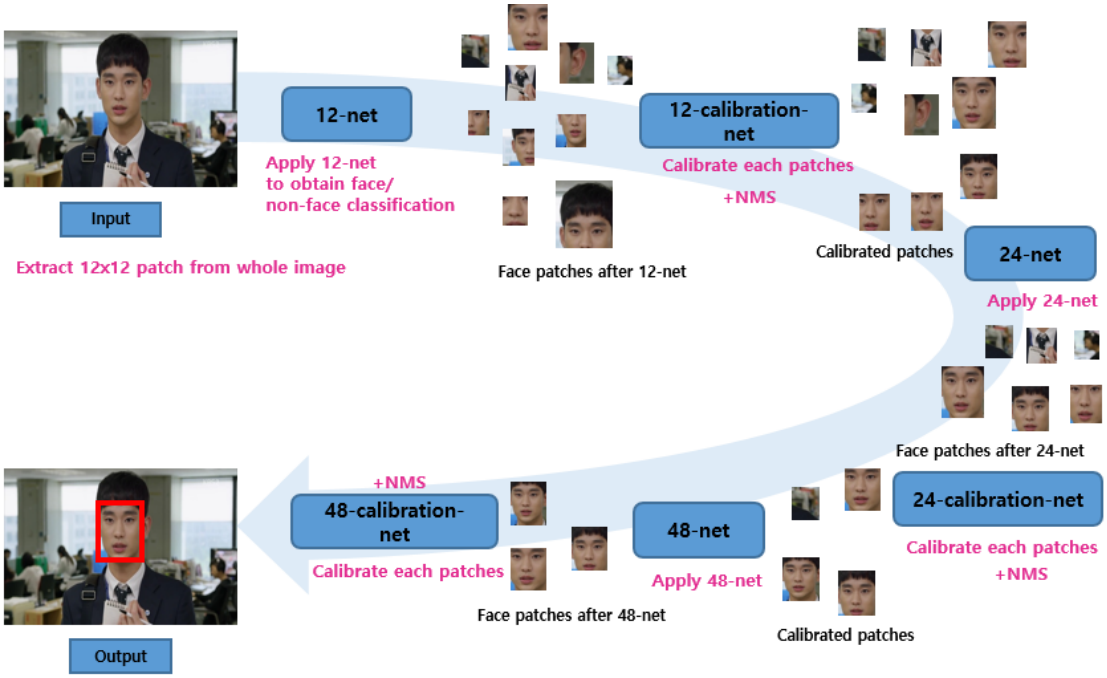


Fig. 1. Overall C-CNN structures of proposed method.

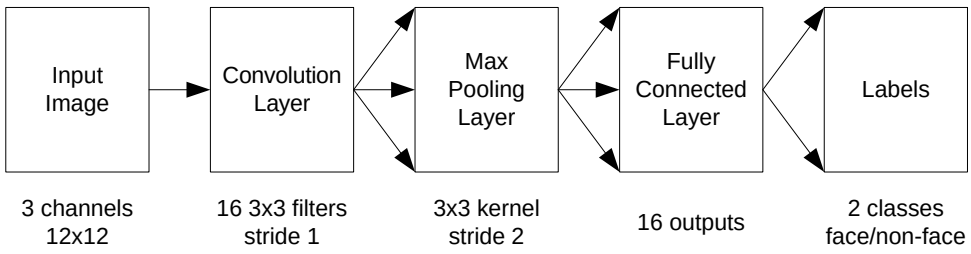


Fig. 2. 12_net C-CNN structure.

12-calibration-net refers to the CNN for bounding box calibration. 12-calibration-net is a shallow CNN as shown in Fig. 3. N calibration patterns are pre-defined as a set of 3-dimensional scale changes and offset vectors as shown in (2).

$$\{[S_n, x_n, y_n]\}_{n=1}^{N} \tag{2}$$

When the detection window is (x, y, w, h) with (x, y) as top-left corner of size (w, h), the calibration pattern adjusting the window is defined as (3).

$$\left(\mathrm{x} - \frac{x_n w}{S_n}, \mathrm{y} - \frac{y_n \mathrm{h}}{S_n}, \frac{\mathrm{w}}{S_n}, \frac{\mathrm{h}}{S_n}\right) \tag{3}$$

The calibration net outputs a vector [c1; c2; _; cN] where [S; x; y] is the average results of the calibration patterns

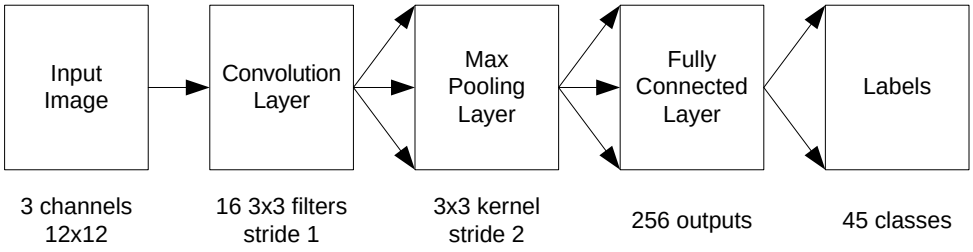


Fig. 3. 12_net calibration C-CNN structure.

24-net is an intermediate binary classification CNN that reduces the number of detection windows. The left detection windows are cropped and resized into 24x24 as input images for the 24-net to further discard approximately 93% of the left detection windows [16]. Similar to the previous process of 12-net, the remaining detection windows are adjusted by the 24-calibration-net. The 12x12 resolution input into the sub-structure is similar to the 12-net in 24-net. From the 12-net substructure, the fully-connected layer is assigned to the 128 output fully connected layer for classification. This multi-resolution structure enables the 24-net to be enhanced by the information from 12x12 resolution, which helps detect minor faces.

By comparing the detection rate with or without using multi-resolution structure in the 24-net at the same recall rate, the one with the multi-resolution structure can achieve the same recall level with less false detection windows. At high recall level, the gap is more vivid. Fig. 4 shows 24-net CNN structure.

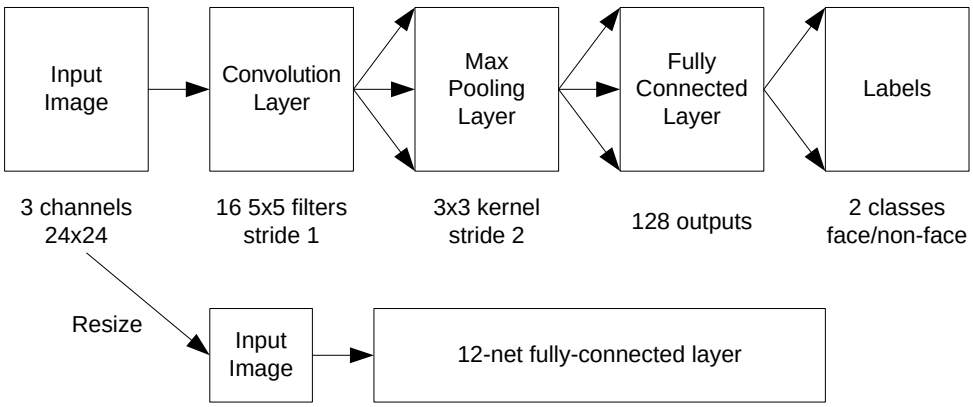


Fig. 4. 24-net C-CNN structure.

Similar to the previous calibration-net, 24-calibration-net is a form of calibration network having an N calibration design. The only difference is that the input size of 24-calibration-net is set to 24x24, but other processes are similar to the ones in the 12-calibration-net.

48-net is the last binary classification C-CNN structure of our C-CNN. The 48-net is quite complex. Considering the previous processes in other networks, the 48-net adopts multi-resolution structure with an additional input copy of 24x24.

For the calibration, 48-net calibration is the last stage in the cascade and the calibration designs are set to N=45. In this stage, only one pooling layer is used in this C-CNN to have more accurate calibration.

III.III Accuracy Improvement of Proposed C-CNN Model

Improving the accuracy of face detection in the referring model of Li et al [14] is the focus of this paper. Compared to previous features, C-CNN can automatically learn features to capture complicated visual disparities by taking advantage of huge amount of training data and easily parallelizing the testing stages on GPU core for acceleration [1, 17]. The C-CNN method for face detection has more benefits in efficiency by training cooperatively different stages to attain superior performance when compared to traditional cascade.

However, during training and testing images for face detection, some challenges are encountered that caused the bad results of the goal we are going to get. To overcome those challenges and improve performance, we had to consider some factors that helped to increase the accuracy for face detection.

Data Augmentation: The reference method was trained on AFLW and COCO datasets for positives and negatives images, as well as FDDB for test images. In the proposed model, we train with more datasets where we increased the number of positives and negatives images by including PASCAL VOC2012 and 3R datasets.

Parameters and Model Optimization: We chose the right parameters, such as increasing the learning rate, input channels, number and size of hidden layers etc. In doing so, we have to check the loss function for the over-fitting problem where the model memorizes the training examples and becomes kind of ineffective for the test set. Hence, the loss may reduce but no changes on the accuracy. This is why drop-out regularization is needed to make sure the weights do not grow too much and overcome the over-fitting.

Dropout: This involves setting the output of every hidden neuron to zero and assigning each with a probability of 0.45 for each iteration of stochastic gradient descent. We also arbitrarily chose a portion of neurons in every layer L and set their values to zero for dropping out of the optimization. At the test time, we do not drop out any neurons but have to scale the weights correctly. Concretely, when the neurons at layer L are dropped out with probability p where p = 0.5, the incoming weights to the layer L should be scaled by p at the test time as (4).

$$\text{test}(\text{L}-1) = p \cdot \theta(\text{L}-1) \tag{4}$$

It is a good test dataset because all of the inputs are equal to 0, so the output will eventually be set to 0.

Table 1 describes some CNN Parameters used to increase accuracy in our proposed model comparing to the referring model of Li et al [4, 14].

Table 1. CNN parameters for each net.

Parameters	12_net	24_net	48_net
Learning rate	0.05	0.05	0.05
Threshold	0.003	0.1	0.1
Input channels	3	5	5
Batch-size	3	5	5

III.IV Training Process

To train the cascaded CNN model, AFLW dataset for positives images and COCO dataset for negatives plus also 3R dataset are used. 3R dataset encompasses around 26,000 images with faces and 27,000 images without faces. The images in 3R dataset reflect everyday life of real world. In order to increase negative samples images, we extract images which have no persons as background images from PASCAL VOC2012 [18]. The datasets all together contain 47,211 images with 82,987 faces and around 32,000 background images. The multinomial logistic regression objective function for optimization is applied in training.

12-net, 24-net, and 48-net are trained following the cascade structure. To train the 12-net, we randomly sample 27,000 non-faces patches from background images and resize all training faces to 12 x 12 size. The 12-net and 12-calibration-net then form a 2 stage cascade that applies to a subset of AFLW images to choose a threshold T1 at a recall rate of 99.5%.

All background images are scanned in the 2-stage cascade. All detection windows with a confidence score greater than T1 become immediately the negative training data for the next network, 24-net. The 24-net is trained with mined negative training data and all training faces are 24 x 24 in size. The same process is repeated for the 4-stage cascade consisting of 12-net with its 12-calibration-net. In addition, 24-net with its 24-calibration-net and threshold T2 are set to maintain a 98% recall rate. The same procedure is applied by mining negative training data for the last network which is 48-net with the 4-stage cascade on all the background images. The 48-net is trained with positive and negative training samples 48 x 48 in size.

In comparison to designs with a simple CNN that scan the full image for different faces, the cascade makes it possible to have simpler CNNs and achieve the same or even better accuracy because at each stage of the cascade, the CNN is trained to address a sub-problem that are easier than addressing the face vs. non-face classification globally.

III.V C-CNN for calibration

When training data are collected for the calibration-nets, the face is annotated with N=45 calibration patterns. Specifically, for the nth pattern $[S_n, x_n, y_n]$, $[1/S_n, x_n, y_n]$ is applied to make accurate adjustment to the face annotation bounding box, crop them and resize into proper input sizes (12x12, 24x24 and 48x48).

One of the key advantages of the proposed model in C-CNN is time efficiency. The joint C-CNN method can be fast for face detection. Moreover, by only varying the threshold values T1 and T2, a task specific accuracy computation tradeoff can be found. During training, in the faster version, the thresholds are set to be aggressively high to reject a large portion of the detection windows in the early stages. The calibration-nets help in improving recall in the later stages by adjusting the bounding boxes. On average only 2% the detection windows passed 12-net and 12-calibration-net. 15.3% of the retained detection windows passed 24-net and 24-calibration-net to put into the last and most computationally expensive network, 48-net.

IV. RESULT AND ANALYSIS

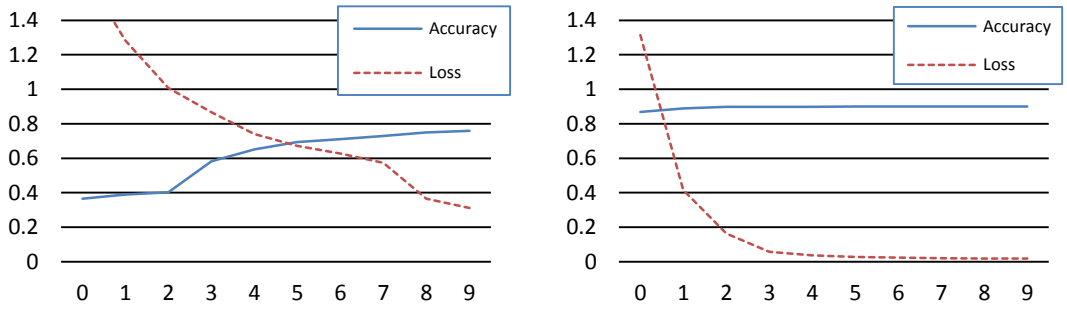
IV.I Accuracy Calculation

Using AFLW, COCO, and 3R dataset, experiments are performed to determine the accuracy of the improved joint cascaded CNN model. For joint loss, each network has face vs non-face detection classification loss and a bounding-box regression loss. Hence, the joint loss function is obtained by adding the two with loss weights as shown in (5)

$$L_{joint} = \lambda_1 Lx12 + \lambda_2 Lx24 + \lambda_3 Lx48 \quad (5)$$

Lx12, Lx24 and Lx48 represent losses of three networks and λ1, λ2, and λ3 represent loss weights of those networks mentioned above.

The accuracy is calculated in the training image process where from 12-net to 48-net. During training and testing, some results for 12-net and 48-net are depicted in Fig. 5 in aspect of accuracy and loss where X axis means the epoch. Training starts at epoch 0 to 9 yet we set epoch number of 10, means in training we take epoch = n-1. And the more loss function decreases, the more our accuracy increases.



(a) (b)

Fig. 5. (a) Accuracy and loss for 12_net and (b) accuracy and loss for 48_net.

As a results, the accuracy increases from 75% of 12-net to 89% of 48-net. The main factor that increases the accuracy is learning frequency, threshold and dropout regularization. The accuracy of cascaded CNN model is summarized in Table 2.

Table 2. Achieved high accuracy of proposed C-CNN.

Stages	Epoch	Iteration	High accuracy calculated
12_net	From 0-9	3000/10000 6000/10000 9000/10000	75%
24-net	From 0-9	3000/10000 6000/10000 9000/10000	85%
48-net	From 0-9	3000/10000 6000/10000 9000/10000	89%

IV.II Comparison with other methods

As mentioned before, there are already other methods in detecting or recognizing a face. Some CNN method with a single level or traditional method with detailed face features are used. These traditional methods with features: LBP, MB-LBP, NPD, LOMO have classifiers as SVM, DQT+AdaBoost but C-CNN has a cascade classifier.

The comparison results of the proposed C-CNN method with other methods are summarized in Table 3. In fact, C-CNN achieves the highest accuracy compared with other methods, it can achieve a higher identification rate even if it needs more training time because of its density but less processing and testing time with accurate results.

Table 3. Comparison of C-CNN with other methods.

Algorithm	# Features	# Selected features	Training time(h)	Testing time(h)	Platform	Accuracy
LOMO	7,252	283,323	5.35	1.52	X5650 CPU	82.1%
LBP	768	20,269	1.80	0.44	X5650 CPU	72.1%
NPD	165,600	6,877,535	6.42	1.10	X5650 CPU	58.0%
MB-LBP	5,120	228,606	3.91	0.34	X5650 CPU	82.0%
CIFAR-10CNN	64	64	2.20	0.50	K40 GPU	75.0%
C-CNN(Proposed)	560	560	5.85	0.6	TitanXP GPU	89.0%

V. CONCLUSION

In this paper, we have presented an improved joint cascaded CNN for face detection and addressed how to improve the accuracy for face detection. The cascade CNN is designed in similar to that of Li et al and improved the accuracy by analyzing the existing C-CNN structures and applying various optimization such as augmenting data, optimizing model and its parameters, and adjusting drop-out. By applying training and testing process methods, the accuracy of cascaded CNN is evaluated and compared with other methods.

Face detection results can be used in various applications such as PIN replacement, criminal identification, prison visitor systems, border control systems, voting systems, computer security, banking using ATM, physical access control of buildings areas etc.

Many applications require high accuracy of face detection. Also, although CNN algorithms are getting somehow faster on high-end GPUs, there is problem in most practical applications like mobile applications, where they are not fast enough.

Therefore, further researches are required to increase performance in speed and accuracy by designing the C-CNN method.

Acknowledgment

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2017 R1D1A1B03030432).

REFERENCES

[1] M. J. Er, W. Chen, and S. Wu, High speed face recognition based on discrete cosine transform and RBF neural network, IEEE Trans. On Neural Network, 16, 2005, 679-691.

[2] S. Yang, P. Luo, C. C. Loy, and X. Tang, From facial parts responses to face detection: A deep learning approach, Proceedings of IEEE Int. Conf. on Computer Vision, 2015, pp. 3676-3684.

[3] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, ImageNet Large Scale Visual Recognition Challenge, International Journal of Computer Vision, 115, 2015, 211-252.

[4] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, A convolutional neural Network cascade for face detection, Proceedings of IEEE Int. Conf. on Computer Vision and Pattern Recognition, 2015, 5325–5334.

[5] P. F. Felzenszwalb, R. B. Girshick, and D. A. McAllester, Cascade Object detection with deformable part models, Proceedings of IEEE Int. Conf. on Computer Vision and Pattern Recognition, 2010, 2241–2248.

[6] R. Vaillant, C. Monrocq, and Y. Le Cun, Original approach For the localization of objects in images, IEE Proceedings of Vision, Image and Signal Processing, 141, 1994, 245-250.

[7] H. Rowley, S. Baluja, and T. Kanade, Neural network-based Face detection, IEEE Trans. On Pattern Analysis and Machine Intelligence, 20, 1998, 23-38.

[8] Y. Sun, X. Wang, and X. Tang, Deep convolutional network cascade for facial point detection, Proceedings of IEEE Int. Conf. on Computer Vision and Pattern Recognition, 2013, 3476–3483.

[9] S. Ren, K. He, R. Girshick, and J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, IEEE Trans. on Pattern Analysis and Machine Intelligence, 39, 2015, 1-14.

[10] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks, Proceedings of

European Conference on Computer Vision, 2016, 525-542.

[11] K. Kavukcuoglu, P. Sermanet, Y.-L. Boureau, K. Gregor, M. Mathieu, and Y. LeCun, Learning convolutional feature hierarchies for visual recognition, Proceedings of Int. Conf. on Neural Information Processing Systems, 2010, 1090-1098.

[12] H. Qin, J. Yan, X. Li, and X. Hu, Joint training of cascaded cnn for face detection, Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition, 2016, 3456-3465.

[13] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, L. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, Speed/accuracy trade-offs for modern convolutional object detectors, Proceedings of IEEE Int. Conf. on Computer Vision and Pattern Recognition, 2017, 7310-7319.

[14] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, A convolutional neural network cascade for face detection, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2015, 5325-5334.

[15] X. Zhu, and D. Ramanan, Face detection, pose estimation, and landmark localization in the wild, Proceedings of IEEE Int. Conf. on Computer Vision and Pattern Recognition, 2012, 2879–2886.

[16] D. Park, D. Ramanan, and C. Fowlkes, Multiresolution models for object detection, Proceedings of European Conference on Computer Vision, 2010, 241-254.

[17] D. Chen, S. Ren, Y. Wei, X. Cao, and J. Sun, Joint cascade face detection and alignment, Proceedings of European Conference on Computer Vision, 2014, 109-122.

[18] K. Simonyan, and A. Zisserman, Very deep convolutional networks for large-scale image recognition, Proceedings of International Conference on Learning Representations, 2015, 1-14.