



A Face Detection Method Based on Cascade Convolutional Neural Network

Wankou Yang^{1,2} · Lukuan Zhou^{1,2} · Tianhuang Li^{1,2} · Haoran Wang³

Received: 5 June 2018 / Revised: 28 October 2018 / Accepted: 28 November 2018 /

Published online: 4 January 2019

© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract

Cascade has been widely used in face detection where classifier with low computational cost can be firstly used to shrink most of the background while keeping the recall. In this paper, a new cascaded convolutional neural network method consisting of two main steps is proposed. During the first stage, low-pixel candidate window is used as an input such that the shallow convolutional neural network quickly extracts the candidate window. In the second stage, the window from the former stage is resized and used as an input to the corresponding network layer respectively. During the training period, joint online training is conducted for hard samples and the soft non-maximum suppression algorithm is used to test on the dataset. The whole network achieves improved performance on the FDDB and PASCAL face datasets.

Keywords Face detection · Cascade convolution structure · Soft non-maximum suppression

1 Introduction

Face detection [30, 33] has attracted much attention in the past decades and has been widely applied to all aspects of human life, e.g. video surveillance, face recognition, human machine interface and image retrieval. However, many variations in illumination, pose, occlusion and shot angle pose great challenges for face detection in real-world applications.

In the early twentieth century, the cascade face detection method developed by Viola and Jones (VJ) [24] was the first work with real-time speed, which is trained using Adaboost and Haar-like feature. Haar-like feature could be efficiently calculated using integral image. However, Haar-like feature is too simple to handle the faces in the uncontrolled environment and it only well works on

✉ Wankou Yang
wkyang@seu.edu.cn

¹ School of Automation, Southeast University, Nanjing 210096, China

² Key Lab of Measurement and Control of Complex Systems of Engineering, Ministry of Education, Nanjing 210096, China

³ College of Information Science and Engineering, Northeastern University, Shenyang 110819, China

the frontal unobstructed face detection. Many improved works based on VJ framework have been proposed in the past decades [13, 21, 31]. They can be divided into two categories: feature-based methods [25, 37] and classifier-based method [1]. Meanwhile, deformable part models (DPM) [28, 36] are developed for face detection and achieve tremendous success, whereas the time complexity is very high with undesirable efficiency. Dollar et al. [3] proposed the integral channel feature (ICF) to detect pedestrian with real time speed. ICF is built by combing HOG and LUV color space to well represent pedestrian. Based on ICF, Dollar et al. [4] aggregate channel feature (ACF) for pedestrian detection. Although ACF inherits the ten channel maps of ICF, ACF utilizes different strategies to extract feature that are subsequently fed into Adaboost with random decision trees. It is suggested that ACF is more robust than ICF. Some techniques in image retrieval [14, 27, 35, 38, 39] could be utilized to perform face detection.

In recent years, with the dramatic progress in convolutional neural networks (CNN) [11, 12], CNN has achieved remarkable success in computer vision and multimedia, e.g. image classification [11], objection detection [6, 7, 15, 19] and image retrieval [17, 21, 22]. Massive face detection methods based on CNN are developed. Yang et al. [31] designed deep convolutional neural network for detecting facial part to yield face candidate. However, the method is insufficiently efficient since the CNN structure is complex. Li et al. [13] first proposed cascaded CNNs for face detection using CNN instead of Haar-like feature and adaboost of VJ framework. Huang et al. [9] proposed DenseBox to detect face by using a single fully convolutional neural network (FCN) to directly predict the bounding box and the object class confidences. The detection accuracy is further improved by performing landmark localization. Yang et al. [32] proposed a coarse-to-fine method, named Faceness-Net. It fuses the scores of facial parts using several DCNN to obtain the face region followed by a refining network to achieve the face detection. Zhan et al. [34] proposed a multi-task cascade convolutional networks (MTCNN) method for joint face detection and alignment. MTCNN contains three parts, P-Net, R-Net and O-Net and is a coarse-to-fine method. Hu et al. [18] proposed a hybrid-resolution model (HR) for small face detection by fusing scale, resolution and context. Wang et al. [26] utilized multi-task loss function, online hard example mining and multi-scale training strategy to improve Faster R-CNN to build a face detection method, named Face R-CNN. Najibi et al. [16] proposed to simultaneously detect faces with different scales from different layers in a single network, named single stage headless (SSH). Tang et al. [23] proposed a context-aware face detection method, named PyramidBox. Several works [2, 13, 34] show that simultaneously performing face detection and landmark localization could further improve the performance of face detection. Inspired by cascade CNN and MTCNN, we propose a face detection method based on the cascaded convolutional structure. The joint training of different convolutional stages in the cascade is conducted for hard samples. The proposed method is verified on FDDB and PASCAL face datasets.

Our contribution includes:

- (1) We propose a cascade structure based CNN for face detection. A shallow FCN is utilized to generate the face proposals while a multi-scale CNN is utilized to refine the face.
- (2) We utilize hard sample mining technique and soft non-maximum suppression to further improve the performance.

The rest of the paper is organized as follows: Section 2 reviews the related work. Section 3 details our proposed work. Section 4 conducts experiments to evaluate the proposed method. Section 5 gives the conclusions.

2 Related work

2.1 Fully Convolution Network

Typical classification neural networks, such as AlexNet [11] and R-CNN [7], need a fixed size window by cropping or scaling the image as the input of the network due to the fully connected layers. The output from the fully connected layers does not contain the spatial information of the input image since the fully connected layer converts the feature map output from the convolutional layer into a vector. Shelhamer et al. [20] proposed fully convolutional networks (FCN) that addressed the limitations of the size of the input image and preserved the spatial information of the image. The heat map is generated to locate the object location by the activation function.

As shown in Fig. 1 and 2, the difference between a fully connected structure and the full convolutional structure manifests itself in the image size in each layer. The resolution of the original image changes from 227×227 to 55×55 after twice operations of convolution and pooling, resulting in an image of size 27×27 after the second pooling. Output by the fifth layer, the image size is reduced to 13×13 . However, in FCN, with an image of size $H \times W$ as input, it decreases to a quarter of the original image after twice operations of convolution and pooling. Then, after each pooling layer, the length and width of the image are reduced by half. Thus, the convolutional feature is one sixteenth of the original size output by the fifth layer. At last, the feature is reduced to one thirty-twoth of the original size. It is shown that there is a significant decrease in the image size after multiple convolution and pooling operations. The heat map with the smallest size can be obtained by the last layer as mentioned above. It can be viewed as important high-dimensional feature map. Subsequently, the image is up-sampled and enlarged to the original image size and the pixel results for the position correspond to the classification result. Due to the significant advantages by the unlimited image size, we adopt full convolutional layer in three multi-resolution networks respectively such that the input image size is no longer limited.

2.2 Spatial Pyramid Pooling

R-CNN [7] object detection method requires that the size of the input image is fixed. Therefore, the input image is usually preprocessed by cropping or resizing. The preprocessing operation will result in information loss and geometric distortion of the object. Thus, detection and recognition accuracy will decrease.

To solve this problem, SPP-Net [8] was proposed to run the CNN model only once on the entire image. Then, the candidate regions obtained by selective search are mapped to the feature maps. Spatial pyramid pooling and SVMs are used to classify candidate targets. It is

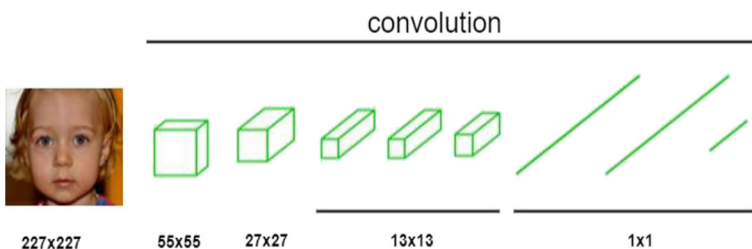


Fig. 1 Fully connected structure

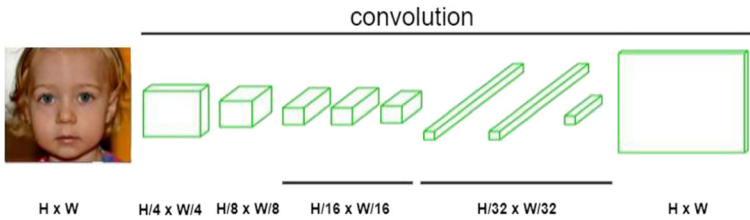


Fig. 2 Fully convolution network structure

possible to obtain convolutional features of arbitrary size through input images of unfixed size and it is only necessary to ensure that the size of the input to the full connected layer is fixed. We will use FCN structure such that the size of the input image cannot be limited anymore. It will generate an output with fixed size. Therefore, the overall structure is different from R-CNN. The flowchart of the spatial pyramid pooling layer structure is shown in Fig. 3.

2.3 The Cascade Structure

Cascade structure has been widely used in face detection, where the classifier with low computational cost can be firstly used to remove most of the background while keeping the recall. The cascaded classifiers work on multiple AdaBoost weak classifiers or strong classifiers to sequentially process different features. The cascade structure is shown in Fig. 4. The flowchart not only creates a strong cascaded classifier by combing multiple weak classifiers, but also increases the speed. However, each stage of the previous methods is trained independently. Therefore, the optimization for different CNNs is independent of each other.

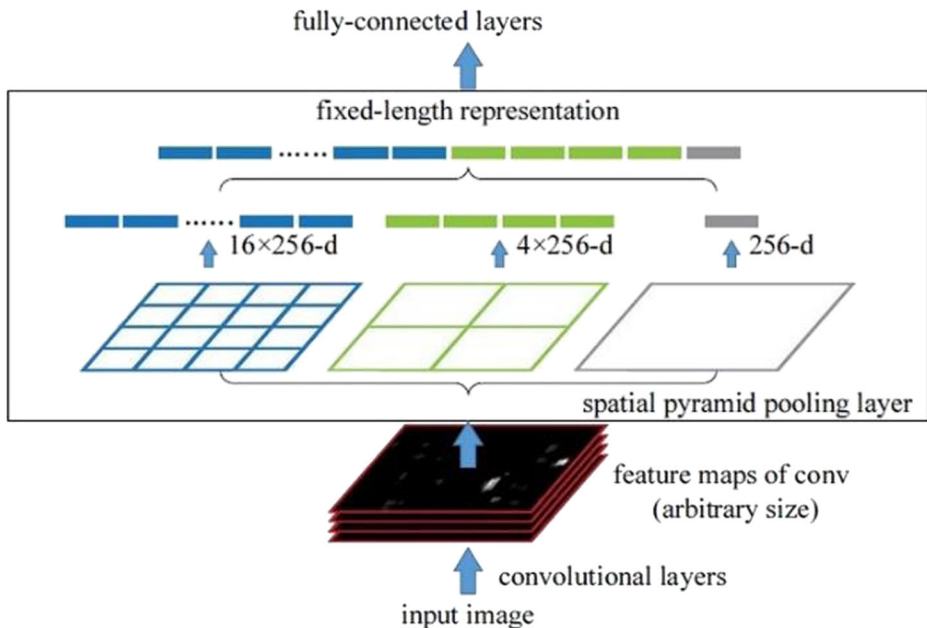


Fig. 3 Spatial pyramid pooling layer structure flow chart

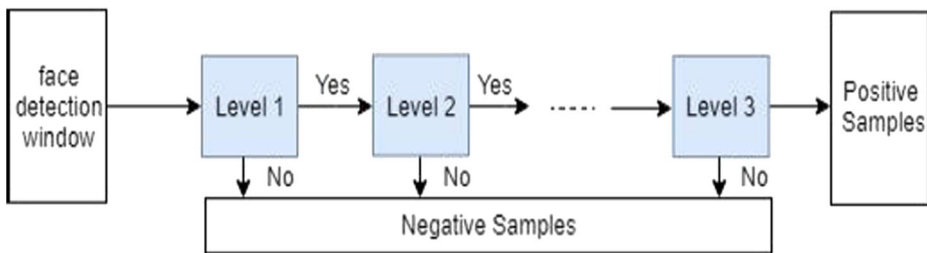


Fig. 4 Cascading classifier structure diagram

In recent years cascade structure has been widely utilized in CNN based detection methods. With the development of CNN, multi-stage CNN is becoming popular and many advanced object detection algorithms also adopt a multi-stage system. The first stage is the Region Proposal Network (RPN). One or more following steps are the networks used for detection. Cascaded CNN and Faster R-CNN use this mechanism. However, previous methods are divided into training and optimization using greedy algorithms. Unlike boosting methods, CNN is a deep neural network optimized using back propagation. Therefore, different network layers can be jointly optimized with computing information shared. MTCNN is an algorithm for face detection and alignment based on cascaded structure. It consists of three networks: Proposal Network (P-Net), Refinement network (R-Net) and Output network (O-Net). First, candidate windows are fed into a shallow CNN network (P-Net) such that a large number of candidate windows are rapidly generated; Then, as for the R-Net, a large number of non-face windows are eliminated through a more complex network refining candidate windows again. Finally, the O-Net is used to finalize the bounding box and the positions of the faces are obtained to achieve face detection and alignment.

2.4 Face detection correlation algorithm based on convolutional neural network

With the rapid development of CNN, many studies have been completed for face detection, for example R-CNN [7], Cascade CNN [13] and DDFD [5].

R-CNN is one of the most classic algorithms for target detection. The general process is to use the Selective Search method to find out 2000 candidates approximately. After the determination, feature extraction is performed on all candidate frame regions using CNN. After the extracted feature vectors are obtained, SVM or other classifiers are used to classify the feature vectors. The performance of R-CNN is unsatisfactory in face detection. Firstly, selective search may miss some face areas and result in low recall rate. Secondly, the bounding box regression cannot align with the ground truth completely, resulting in poor positioning accuracy. In addition, due to the calculation of the eigenvalues of all the candidate boxes, the feature extraction through CNN is the main bottleneck and the speed is slow. Fast R-CNN [7] and Faster R-CNN [6] are developed based on RCNN.

Cascade CNN is proposed for face detection. Cascade CNN consists of three stages. At each stage, they use a detection network and a calibration network, a total of six CNNs. In practice, it makes the training process quite complicated. We must carefully prepare training samples for each stage and optimize the networks one by one. One question that naturally arises is how to train a network at all stages. Firstly, the detection network and the calibration network can share a multi-loss network for detection and bounding box regression. Multi-loss optimization has been proven effective in general target detection. Secondly, if multi-resolution

is used in later training, as in Cascade CNN, the convolutional network is divided into three phases. The latter network contains the previous network. Meanwhile, sharing the convolutional layer can reduce the computational cost. Thirdly, in a cascaded network, the separated first stage is used for generating candidate windows, but is optimized on a separate network, and in a federated network, it can be optimized for branch offices on a larger scale. This allows each network branch benefits from other networks.

DDFD (Deep Dense Face Detector) [5] is a face detection method based on deep neural network. It can detect faces with different angles and can solve the problem of face occlusion to some extent. Compared with R-CNN, DDFD only requires a deep convolutional neural network model to detect faces with different angles. Therefore, the computational complexity is relatively low. In addition, the training data when training the model does not need to mark face poses and information about alignment of facial features. The goal of DDFD is to use a simple classifier to detect faces in different poses with minimal computational complexity. It solves the fixed size problem of the input image through the idea of a full convolutional network and obtains better detection results. The algorithm first fine tunes a well-trained model such as AlexNet, then converts the fully-connected layer of the model to a convolutional layer to obtain an output heat map, and locates the face position by non-maximal suppression.

3 Our proposed work

Due to the effectiveness of the cascade structure, cascade structure is often recommended in the traditional methods. Here, we propose a face detection method based cascade structure and CNN using the effective model architecture, more reasonable data sampling strategies and training data.

3.1 Structural design

In this section, we will describe a cascade CNN for face detection by using three input images with different resolutions (12×12 , 24×24 and 48×48). The input image is resized to different scales to form an image pyramid. Firstly, a large number of non-face windows are eliminated through a micro network (Fully convolutional proposal network, FCPN). Then, the rest of candidate window is fed to the second stage (Multi-scale network, MSN). MSN-24 represents the branch with input size 24×24 , while MSN-48 represents the branch with input size 48×48 . The convolutional features (i.e. probability distribution information) in fifth layer of MSN-24 are fused with MSN-48's later. Hard sample mining and joint training for different cascade stages will be conducted to complete two tasks: face classification and bounding box regression. The overall pipeline of our proposed method is shown in Fig. 5.

In our proposed work, the input image is resized to different scales to create an image pyramid. The detection process is divided into two stages. The first stage is the Full Convolution Proposal Network (FCPN) which uses a low resolution shallow convolutional neural network structure to effectively eliminate a large number of background windows quickly, as shown in Fig. 6. The second stage is

Multi-scale Network (MSN), which combines the features of two higher resolution convolutional neural networks by weighted threshold to further filter out hard samples and refine the bounding box. The structure is shown in Fig. 7. The two stages will be described in details as follows.

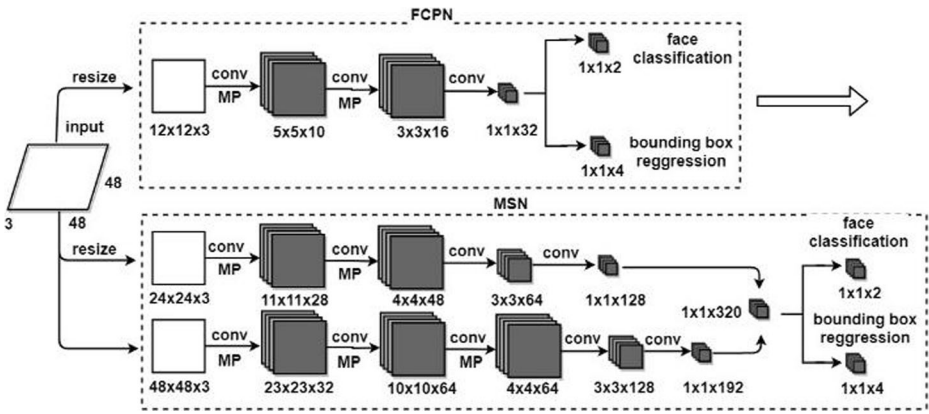


Fig. 5: The overall framework of our proposed work

Stage 1: FCPN is a micro FCN, which is mainly used to generate region proposals and bounding box regression vectors. First, the input image is resized to 12×12 . Then, it passes through a 3×3 convolution kernel with a step size of 1 to get a 10×10 feature map. This feature map is pooled with a 2×2 maximum template to derive a feature map of 5×5 . Then, the feature maps are successively convolved with 3×3 convolution kernels. At last, a two-dimensional vector and a four-dimensional vector are produced as output. The two-dimensional vector represents the probability of being a face or non-face. The four-dimensional vector represents the position information of the Bounding Box (i.e. the horizontal and vertical coordinates of the upper left corner, the length and width of the rectangles). Overlapping candidate boxes are merged using non-maximum suppression (NMS). Meanwhile, candidate windows are corrected by bounding box regression.

The number of faces in the detected image is limited while the remaining candidate windows are background images. It can be seen that the number of positive samples is extremely limited in the training process and the negative samples tend to be infinite. In order to avoid the detector decision biased towards negative characteristics, we cannot use all negative samples for training. We need to ensure the number of positive samples and negative samples are balanced. The network quickly eliminates a large number of background windows at the beginning of the detection and is suitable for being incorporated into one or more stages of the neural network, which is consistent with RPN in the Faster R-CNN.

Stage 2: MSN consists of two branches which are MSN-24 and MSN-48. The candidate window filtered by FCPN is resized as the input of the MSN. Through a 3×3 convolution

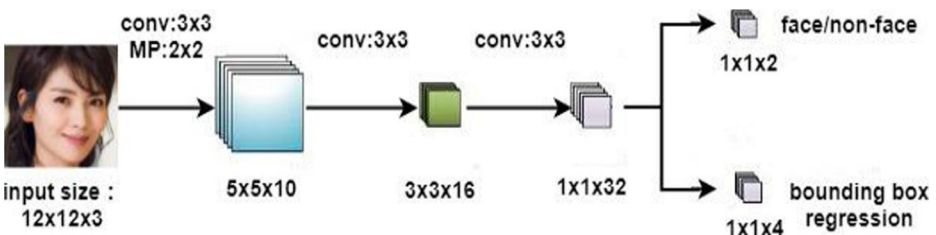


Fig. 6 The architecture of FCPN

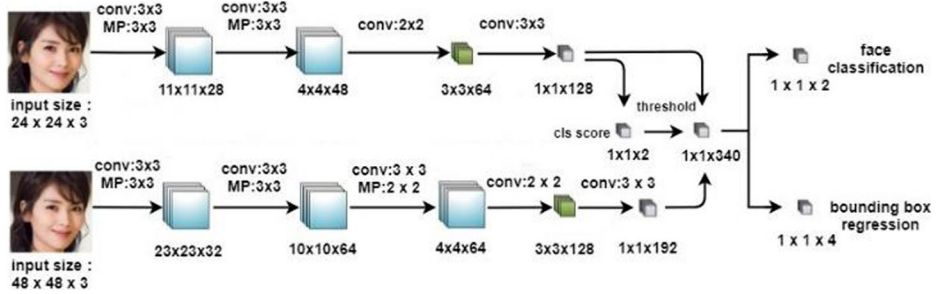


Fig. 7 The architecture of MSN

kernel, the step size is 1 and a 22×22 map is obtained. After 3×3 pooling, a heat map is generated. Similarly, we can observe the size of the later feature map. As for MSN-24, the face that is affected by partial occlusion, uneven light, posture, etc. could report higher detection confidence score. It is possible to result in false detection, so another stage (MSN-48) with higher pixel input in parallel is added. We choose a classification score threshold for the score threshold layer. Only the above proposals contribute to the loss of final layers. In our experiments, 0.15 is an appropriate threshold.

The second-stage convolution step is similar to the previous work. Considering the higher resolution and expensive computation costs, a maximum pooled template with 2×2 and step size of 1 is added when the third layer is convoluted. The size of output is the same as that of the upper layer. After the classification score are weighted through the training threshold, the bounding box regression and the NMS are combined to produce the final output.

3.2 The training

The cascade face classifiers are trained through two tasks: face and non-face classification and bounding box regression.

- Symbol notation

For convenience, we list the symbols used in the paper in Table 1.

- Face classification

Table 1 Important Symbols used in the paper

x_i	the sample
z_j^i	the 2D feature
p_j^i	the probability
y_i^{det}	the class label
y_i^{box}	the box coordinate
L_i^{det}	the classification loss fuction
L_i^{box}	the bounding box regression loss fuction
α_j	the weight parameter
β_i^j	the type parameter

The learning target is expressed as a dichotomous problem. For each sample x_i , we first compute a two-dimensional vector z_j^i , where $feature_j^i$ represents the feature of the sample x_i in the j th pooling layer and $\mathcal{O}_j(\cdot)$ represents the nonlinear transformation function in j th pooling layer.

$$z_j^i = \mathcal{O}_j(feature_j^i) \tag{1}$$

Then, we use the nonlinear activation function to calculate the probability that the sample x_i may be a face p_j^i using Eq. (2),

$$p_j^i = \frac{e^{z_{j,1}^i}}{e^{z_{j,1}^i} + e^{z_{j,2}^i}} \tag{2}$$

where $e^{z_{j,1}^i}$ represents the first element of two-dimensional vector z_j^i , $e^{z_{j,2}^i}$ represents the second element of two-dimensional vector z_j^i .

The cross-entropy function is used as the loss function as shown in Eq.(3).

$$L_i^{det} = -(y_i^{det} \log(p_i) + (1-y_i^{det})(1-\log(p_i))) \tag{3}$$

where p_i represents the probability that the sample x_i is a face based on the network, $L_i^{det} \in \{0, 1\}$ represents the ground truth.

- Bounding Box Regression

At last, the bounding box which is predicted according to each candidate is compared with the ground truth. The objective function can be summarized as regression problems. For each sample x_i , we use the Euclidean distance to calculate the following loss,

$$L_i^{box} = \|\hat{y}_i^{box} - y_i^{box}\|^2 \tag{4}$$

where \hat{y}_i^{box} is the result of the network calculation of the target; y_i^{box} is the ground truth coordinates (a total of four coordinates: the vertical and horizontal coordinates of the upper left corner and the height and width of the detection window), $y_i^{box} \in R^4$.

- Joint Training

Cascade structure is inconvenient for joint training directly, so it compromises the end-to-end learning of CNN. Since the traditional cascade training is usually optimized separately, the results cannot be compared with joint multi-stage. Thus, we propose to perform joint training through back propagation. For this cascaded structure, the learning objective function can be written as Eq. (5). For the background image, we only calculate L_i^{det} and set the loss of the other to zero.

$$L_{joint} = \sum_{i=1}^N \sum_{j \in \{det, box\}} \alpha_j \beta_i^j L_i^j \tag{5}$$

where N is the number of training samples and α_j represents the importance of the task. In the experiments the parameters α_{det} and α_{box} are set as $\alpha_{det} = 1$, $\alpha_{box} = 0.5$ when training FCPN, α_{det} and α_{box} are set as $\alpha_{det} = 0.5$, $\alpha_{box} = 1$ when training MSN, where $\beta_i^j \in \{0, 1\}$ represents the sample type, Positive:1, Negative:0.

- Hard Sample Mining

Different from hard sample mining during traditional classifier training, we adaptively select hard samples during training. In each batch, the loss function of the candidate region is calculated and they are sorted according to the loss value. The target area with the top 70% of the loss value is selected as hard samples, and the remaining 30% of simple samples are ignored. In order to assess the benefits of this method, we train two different models for comparison (w/ and w/o the hard online training of samples) and evaluate the performance on the test set. Fig. 8 shows two different network results. The solid line shows the performance of mining with hard samples. The dotted line shows the effect of not using this method. The experimental results show that online training of difficult samples can help improve the detection performance, providing performance gain of 1.5% on FDDb.

3.3 Soft non-maximum suppression

In the first stage, FCPN generates a large number of candidate regions and MSN refines the regions. Non-maximal suppression is used for post-processing. The method is to sort the detection results by score as usual and retain the candidates with the highest score. Meanwhile, we remove other boxes whose area of overlap with the box is greater than a certain proportion. As mentioned above, the limitation of this greedy method is that it will force the scores of adjacent detection boxes to zero. In this case, the detector fails to detect if a positive appears in the overlapping area, and thus the average detection rate of the algorithm (Average Precision) decreases. The details are shown in Fig. 9. The red boxes and the green boxes are current detection results, in which the green box score is higher than the red box score. If one follows the traditional NMS process, the green box with the highest score is firstly selected, and then the red box may be removed because the overlapped area is large than the threshold. Secondly, it is not easy to set the threshold of NMS. If the threshold is too small, the red box will be filtered out because of the large overlap area with the green box. If the threshold is too high, it is easy to increase the false detection. Therefore, we adopt a new idea, called Soft non-maximum

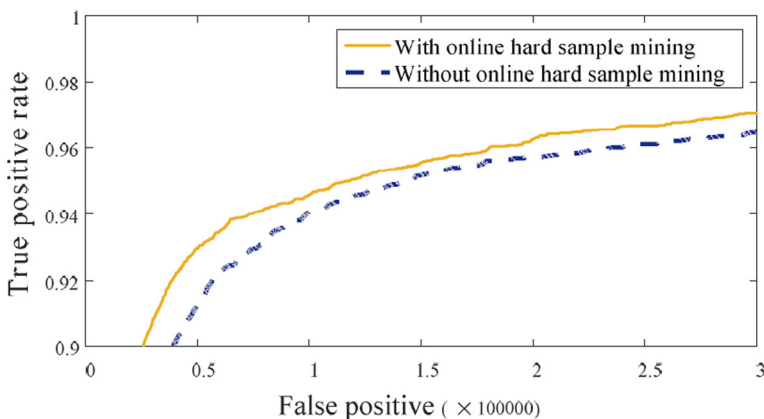


Fig. 8 Hard sample mining performance

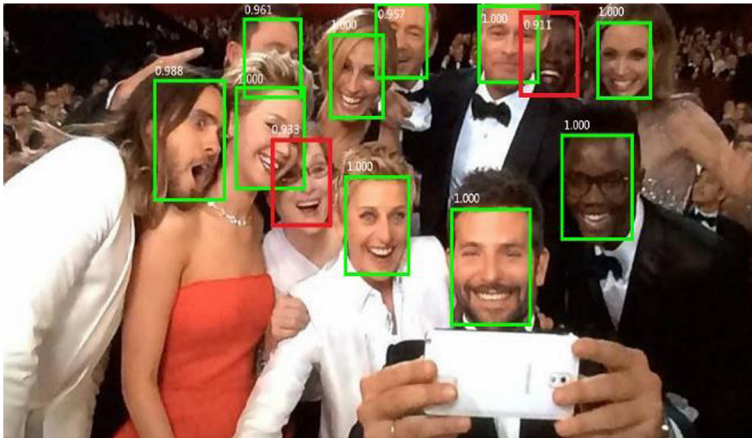


Fig. 9 NMS face detection results

suppression (Soft-NMS). It does not remove all boxes whose IoU (Intersection-over-Union) is larger than the threshold but reduces its confidence through a function based on the degree of overlap with M . M is the current score maximum box. Thus, the score of the adjacent detection frame is decreased instead of being completely eliminated. Although the score is reduced, the adjacent detection result is still in the object detection sequence.

Soft-NMS not only has the same overlapping threshold parameter N_t as the traditional NMS, but also has the parameter σ in the Gaussian weighting method. In the experiment, two non-maximal suppression methods are compared on the Pascal Faces data set based on the proposed cascaded convolutional neural network. The continuous function attenuates the detection score of the non-maximum detection frame rather than completely remove it. Only simple changes to the traditional NMS algorithm are needed and no additional parameters are added. We have compared the effect of the traditional NMS and two Soft-NMS (G for Gaussian weighting, L for linear weighting) on the detection effect under different parameter settings (N_t to 0.3, 0.5, 0.7 respectively in the Pascal Faces data set) and report the best performing parameter σ size. The results are shown in Table 2. We could observe that the Soft-NMS can obviously achieve a 1% performance improvement on average when $N_t=0.5$ and $\sigma=0.6$, whilst the best results are obtained without the extra training cost.

Table 2 The results of Soft-NMS with different N_t on *Pascal Faces*

N_t	Method	AP	σ
0.3	Ours-NMS	92.06	0.5
0.5	Ours-NMS	91.84	0.3
0.7	Ours-NMS	91.02	0.4
0.3	Ours+S-NMS G	93.15	0.6
0.5	Ours+S-NMS G	93.33	0.6
0.7	Ours+S-NMS G	93.10	0.6
0.3	Ours+S-NMS L	92.99	0.5
0.5	Ours+S-NMS L	92.95	0.6
0.7	Ours+S-NMS L	91.84	0.6

4 Experiments

4.1 The dataset and setup

The proposed face detection algorithm is evaluated on two face datasets (Pascal Faces and FDDB). FDDB [10] face dataset includes 5171 annotated face images from 2845 images with 300×400 resolution captured from the wild environment. Pascal Face [29] is widely used face dataset. It includes 1335 annotated face from 851 images.

Our proposed method is compared with some of the classical algorithms mentioned: Faceness, DDFD, HeadHunter, Structured Models, Google Picassa and Face++.

ROC (True Positive Rate-False Positive) curve and PR curve (Precision-Recall) are used as performance measure. The experiments are conducted on a computer server (Ubuntu 14.04.01, Geforce GTX Titan X with $12G \times 4$ and mxnet framework).

4.2 Experimental results

The curve of precision-recall on the Pascal Faces dataset is shown in Fig. 10. It can be seen that the proposed method exceeds Faceness and DDFD by nearly two percentage. It also has certain advantages compared with the face detection results of two commercial systems (Picasa and Face++). As observed from the results, the improved cascaded convolutional neural network can improve the accuracy and increase the speed of the detector due to the fact that non-face window is quickly eliminated through the shallow network.

Fig. 11 and Fig. 12 show the detection results of different methods (e.g. Faceness, DDFD, Cascade CNN, DP2MFD, CCF, HeadHunter and the method proposed in third chapter: ACF-

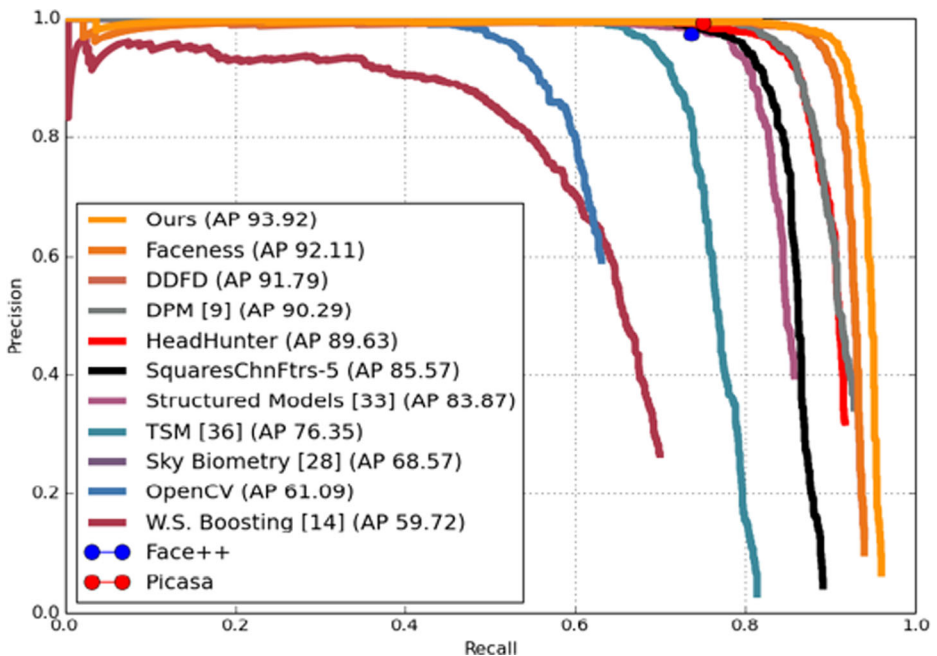


Fig. 10 The detection results on Pascal Faces dataset

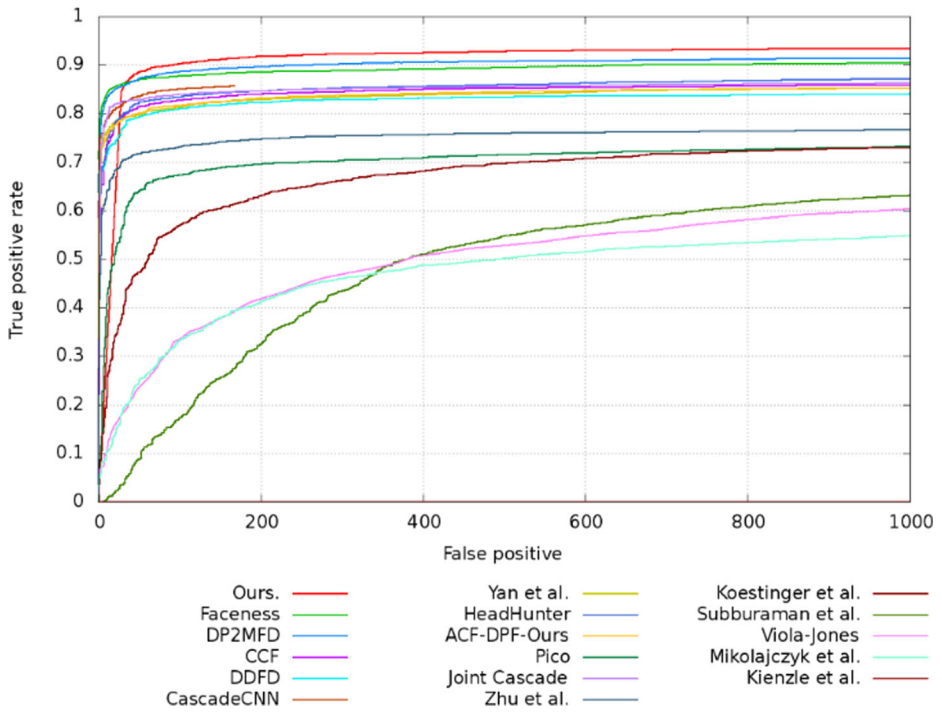


Fig. 11 The discROC curves on FDDB face dataset

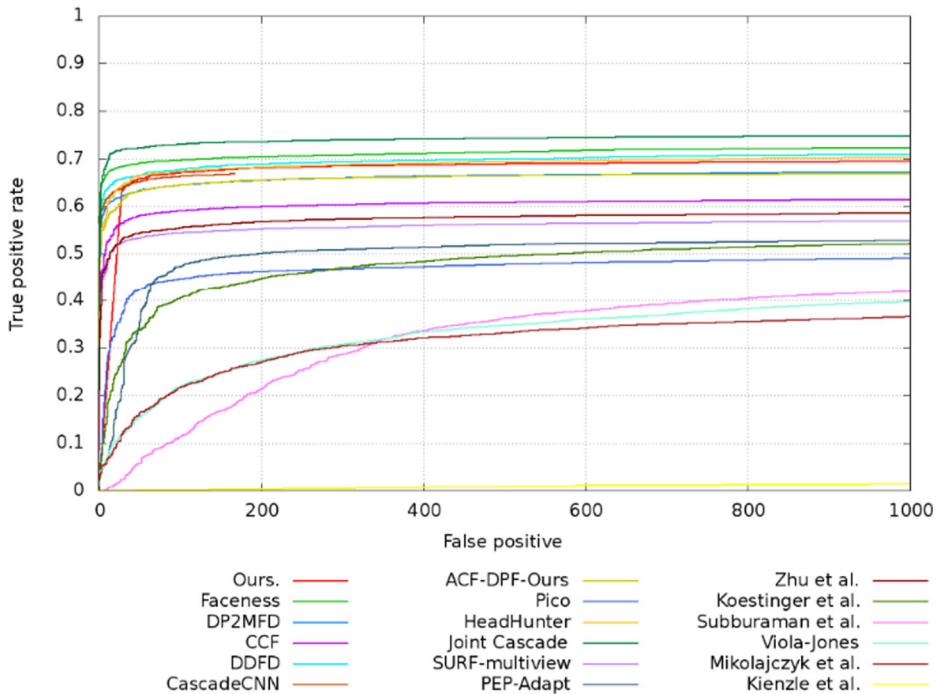


Fig. 12 The contROC curves on FDDB face dataset



Fig. 13 Part of the test results

DPF-Ours) on the Fddb dataset using the two different evaluation criterions respectively: discROC and contROC.

It is shown in Fig. 11 that the detection result of the discrete score on the Fddb data set is 93.4%; Compared with other convolutional neural networks (Faceness: 90.3%, DDFD: 84%,

Cascade CNN: 85.6%), the result is further improved. For some classical methods (DP2MFD: 91.3%, Yan et: 85.2%, ACF-DPF-Ours: 85.41%) and a method that combines convolutional features with traditional features.

(CCF: 85.9%), the result is significantly improved; Fig. 12 shows that the continuous score of the FDDB dataset is 69.5%. The continuous score of the method has no advantage compared with other deeper convolutional neural networks, but it is significantly improved compared with classical method by adding traditional classifiers. From the evaluation results on the FDDB dataset, we can conclude that our proposed face detector achieves better results with discrete scores when the number of false detections is certain, but the continuous score is not promising and remains to be further improved by adjust the positioning of the rectangle.

4.3 Detection examples

Figure 13 qualitatively shows some detection examples of our proposed method. It can be seen that the proposed method can detect some hard faces in uncontrollable natural backgrounds, including faces with changes in occlusion, posture and angles. However, missed detection occasionally occurs particularly if the image is over-shadowed and the face is too small. This is related to the decreasing pixel features of the convolutional layer, which will be addressed in our future research.

5 Conclusion

In this paper, we have presented a new cascaded convolutional neural network for face detection. Firstly, we feed a low-pixel candidate window to a shallow convolutional neural network which can quickly generate the candidate windows. Secondly, the features of two higher pixel convolutional neural networks are combined via weighted threshold to refine the candidate windows. The whole network is further improved by hard sample mining and joint training. We evaluate our proposed method on PASCAL Faces and FDDB face datasets. The experimental results show that our proposed method achieves the superior performance compared to other methods.

Acknowledgements This work is supported by National Natural Science Foundation (NNSF) of China under Grant No. 61473086, 61603080 and 61773117. Jiangsu key R & D plan (No.BE2017157).

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Bourdev L, Brandt J (2005) Robust Object Detection via Soft Cascade, *Computer Vision and Pattern Recognition*, 236–243
2. Chen D, Ren S, Wei Y, Cao X, Sun J (2014) Joint cascade face detection and alignment, in *European Conference on Computer Vision*, 109–122
3. Dollár P, Tu Z, Perona P, Belongie S (2009) Integral channel features, in *BMVA*
4. Dollár P, Appel R, Belongie S, Perona P (2014) Fast feature pyramids for object detection. *IEEE Trans Pattern Anal Mach Intell* 36(8):1532–1545
5. Farfåde SS, Saberian M, Li L, Multi-view face detection using deep convolutional neural networks, *ICMR2015*

6. Girshick R, Fast R-CNN, ICCV2015
7. Girshick R, Donahue J, Darrell T, Malik J, Rich feature hierarchies for accurate object detection and semantic segmentation, IEEE CVPR2014
8. He K, Zhang X, Ren S et al (2015) Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans Pattern Anal Mach Intell* 37(9):1904
9. Huang L, Yang Y, Deng Y, Yu Y (2015) DenseBox: Unifying Landmark Localization with End to End Object Detection arXiv:1509.04874
10. Jain V, Learned-Miller E (2010) FDDB: A benchmark for face detection in unconstrained settings, Tech. Rep. UM-CS-2010-009, University of Massachusetts. In: Amherst
11. Krizhevsky A, Sutskever I, Hinton G (2012) Imagenet classification with deep convolutional neural networks. *NIPS* 1097–1105
12. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521:436–444
13. Li H, Lin Z, Shen X, Brandt J, Hua G (2015) A convolutional neural network cascade for face detection computer vision and pattern recognition
14. Li J, Lu K, Huang Z, Zhu L, Shen HT Transfer independently together: a generalized framework for domain adaption. *IEEE Trans Cybern, Digit Object Identifier*. <https://doi.org/10.1109/TCYB.2018.2820174>
15. Liu L, Ouyang W, Wang X, Fieguth P, Chen J, Liu X, Pietikäinen M (2018) Deep learning for generic object detection: a survey, arXiv:1809.02165v1 [cs.CV] 6 Sep
16. Najibi M, Samangouei P, Chellapa R, Davis LS, SSH: single stage headless face detector, ICCV2007
17. Nie L, Wang X, Zhang J, He X, Zhang H, Hong R, Tian Q, Enhancing micro-video understanding by harnessing external sounds, *ACMM2017*
18. Peiyun H, Ramanan D (2017) Finding tiny faces, *CVPR*
19. Ren S, He K, Girshick R, Sun J, (2016) Faster R-CNN: Towards real-Time object detection with region proposal networks, *IEEE CVPR* 1137–1149
20. Shelhamer E, Long J, Darrell T (2014) Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans Pattern Anal Mach Intell* 39(4):640
21. Shen F, Xu Y, Liu L, Yang Y, Huang Z, Shen HT, Unsupervised Deep Hashing with Similarity-Adaptive and Discrete Optimization, *IEEE Transactions on Pattern Analysis and Machine Intelligence*. <https://doi.org/10.1109/TPAMI.2018.2789887>
22. Song X, Feng F, Han X, Yang X, Liu W, Nie L Neural compatibility modeling with attentive knowledge distillation, *SIGIR2018*
23. Tang X, Du DK, He Z, Liu J, (2018) PyramidBox: A Context-assisted Single Shot Face Detector. arXiv preprint arXiv:1803.07737
24. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features, in *Proceedings of the 19th Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001*, pp. 511–518. *IEEE*
25. Wang X, Han TX, Yan S (2009) An hog-lbp human detector with partial occlusion handling, *IEEE ICCV*
26. Wang H, Li Z, Ji X, Wang Y, Face R-CNN (2017) arXiv preprint arXiv:1706.01061
27. Xie L, Shen J, Han J, Zhu L, Shao L, Dynamic multi-view hashing for online image retrieval, *IJCAI2017*
28. Yan J, Lei Z, Wen L, Li S (2014) “The fastest deformable part model for object detection,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2497–2504
29. Yan J, Zhang X, Lei Z, Li SZ (2014) Face detection by structural models. *Image Vis Comput* 32(10):790–799
30. Yang MH, Kriegman D, Ahuja N (2002) Detecting faces in images: A survey, *IEEE Trans. PAMI*
31. Yang S, Luo P, Loy CC, Tang X (2015) From facial parts responses to face detection: A deep learning approach, in *IEEE International Conference on Computer Vision*, 3676–3684
32. Yang S, Luo P, Loy CC, Tang X (2018) Faceness-Net: face detection through deep facial part response. *IEEE Trans Pattern Anal Mach Intell* 40(8):1845–1859
33. Zafeiriou S, Zhang C, Zhang Z (2015) A survey on face detection in the wild: past, present and future. *Comput Vis Image Underst* 138:1–24
34. Zhan K, Zhang Z, Li Z, Qiao Y (2016) Joint face detection and alignment using multi-task cascade convolutional Networks. *IEEE Signal process lett* 23(10):1499–1503
35. Zheng R, Yao C, Jin H, Zhou L, Zhang Q, Dong W (2015) Parallel key frame extraction for surveillance video service in a smart city. *PLoS One* 10(8):e0135694
36. Zhu X, Ramanan D (2012) “Face detection, pose estimation, and landmark localization in the wild,” in *IEEE Conference on Computer Vision and Pattern Recognition* 2879–2886
37. Zhu Q, Yeh MC, Cheng KT, Avidan S (2006) Fast human detection using a cascade of histograms of oriented gradients, *IEEE CVPR*
38. Zhu L, Huang Z, Chang X, Song J, Shen HT, Exploring consistent preferences: discrete hashing with pair-exemplar for scalable landmark search, *ACMM2017*
39. Zhu L, Huang Z, Li Z, Xie L, Shen HT (2018) Exploring auxiliary context: discrete semantic transfer hashing for scalable image retrieval. *IEEE Trans NNLS* 29(11):5264–5276



Wankou Yang received his B.S., M.S. and Ph.D. degrees at the School of Computer Science and Technology, Nanjing University of Science and Technology (NUST), P.R. China, 2002, 2004 and 2009 respectively. From July 2009 to Sep. 2011, he worked as a postdoctoral fellow at the School of Automation, Southeast University. From Sep. 2011 to March 2016, he worked as an assistant professor at the School of Automation, Southeast University. Now he is an associate professor in the School of Automation, Southeast University, P.R. China. His research interests include pattern recognition, computer vision.



Lukuan Zhou received the Bachelor's degree in the School of South China University of Technology, China in 2017. He is currently a graduate student in the School of Automation, Southeast University, China. He is interested in the research of neuron network, computer vision and object detection.



Tianhuang Li received the Bachelor's degree in the College of Engineering, Nanjing Agricultural University, China in 2015. She is currently a graduate student in the School of Automation, Southeast University, China. Her is interested in the research of neuron network, machine learning and pattern recognition.



Haoran Wang received the B.S. degree from the Department of information science and technology, Northeast University, China, in 2008, and the PhD degree from School of Automation, Southeast University, China, in 2015. Since spring 2015, he has been an Assistant Professor at Northeastern University, China. His research interests include computer vision, pattern recognition, and machine learning.