Robust Face Detector with Fully Convolutional Networks

Yingcheng Su, Xiaopei Wan, and Zhenhua Guo(✉)

Graduate School at Shenzhen, Tsinghua University, Shenzhen, China
zhenhua.guo@sz.tsinghua.edu.cn

Abstract. Many of the exist face detection algorithms are based on the generic object detection methods and have achieved desirable results. However, these methods still struggle in solving the problem of partial occluded face detection. In this paper, we introduce a simple and effective face detector which uses a fully convolutional networks (FCN) for face detection in a single stage. The proposed FCN model is used for pixel-wise prediction instead of anchor mechanism. In addition, we also apply a long short term memory (LSTM) architecture to enhance the contextual infomation of feature maps, making the model more robust to occlusion. Besides, we use a light-weighted neural network PVANet as the backbone, which greatly reduces the computational burden. Experimental results show that the proposed method achieves competitive results with state-of-the-art face detectors on the common face detection benchmarks, including the FDDB, WIDER FACE and MAFA datasets, what's more, it is much more robust to the detection of occluded faces.

Keywords: Face detection · FCN · LSTM · Occlusion

1 Introduction

Face detection has always been a research hotspot as it is a crucial step of many facial applications, such as face alignment, face recognition, etc. Since the pioneering work of Viola-Jones face detector [1], a lot of face detection methods have been proposed. The hand-crafted features [2,3] usually rely on prior knowledge leading to poor performance in complex scenes, especially faces with occlusion.

In recent years, convolutional neural networks (CNNs) have great success in the field of computer vision, including image classification [4,5] and object detection [6–9], etc. The Object detection algorithms such as fast [6]/faster [7] R-CNN, SSD [9], YOLO [8] continue to make new breakthroughs in both speed and precision. Face detection is a special case of object detection. Many face detection approaches are based on object detection methods [10–13] and achieve promising results. However, these anchor-based methods are badly rely on the

Z. Guo—The work is partially supported by the Natural Science Foundation for China (NSFC) (No. 61772296) and Shenzhen fundamental research fund (Grant Nos. JCYJ20160531194840025 and JCYJ20170412170438636).

J.-H. Lai et al. (Eds.): PRCV 2018, LNCS 11258, pp. 207–218, 2018.
https://doi.org/10.1007/978-3-030-03338-5_18

Fig. 1. Our face detector is robust to heavy occlusion and large appearance.

number of matching proposals. If the faces are partial occluded, it's very likely that the models would miss the proposals of occluded faces or be confused by the features of occluded faces. The cascaded network [17,18] is another type of CNNs-based face detection approach. Several small CNNs are cascaded to detect faces in a coarse-to-fine manner. In spite of very fast speed, these shallow networks failed to represent robust image features to handle faces with occlusion.

Inspired by [20], we consider face detection problem as the combination of binary classification and bounding box regression. In this paper, we propose a fast and efficient face detector that only need two steps for face detection. First, a FCN is used to do the pixel-wise classification and bounding box regression. Then, the produced face predictions are sent to Non-Maximum Suppression (NMS) to yield final results. By making such dense predictions, the model has strong robustness to faces with occlusion. In addition, considering the highly-correlated of adjacent regions of the feature map, we use an in-network recurrent architecture to encode rich context information of the feature map. Even if the face is partial occluded, the model can make the correct predictions from the non-occluded part. An example of our detection results can be found in Fig. 1.

The main contribution of this paper can be summarized as:

- We propose a novel FCN-based face detection method that directly make dense predictions in feature maps. The proposed method is fast, accurate and quite simple, which only consist of two step: a forward propagation of the FCN and a NMS merging.
- We use a recurrent architecture to connect the context information of the feature maps, improving the model's capacity of detecting faces with occlusion.

– The proposed method achieves competitive results in FDDB, WDIER Face datasets, and outperforms state-of-the-art methods in occluded faces datasets like MAFA.

2 Related Work

Before the revolution of deep learning, Face detection has been widely studied. Numerous face detector are based on traditional machine learning methods. The pioneering work of Viola-Jones [1] utilizes Adaboost with Haar-like feature to train a cascade model to detect face and get real-time performance. Since then the studies of face detection focus on designing more efficient features [22,23] and more powerful classifiers [26,27]. Deformable pattern models (DPM) [25] are employed for face detection task and achieve promising results. Liao et al. [24] proposed normalized pixel difference (NPD) features and constructed a deep quadratic tree to handle unconstrained face detection. However, these hand-crafted features always require prior assumptions which would be untenable in complex scenarios, leading to low precision in the challenging face datasets, such as WIDER Face and MAFA.

In recent years, the CNN-based face detectors achieved remarkable performance. Li et al. [17] use cascaded CNNs for face detection. Zhang et al. [18] propose Multi-task cascaded CNNs (MTCNN) to detect face and align face, simultaneously. Qin et al. [19] integrate the training of cascaded CNNs into a framework for end-to-end training, which greatly improves the performance of cascaded networks. Faceness [28] generates face parts responses from attribute-aware networks to detect faces under occlusion and unconstrained pose variation. However, this method needs to label facial attributes of different facial parts and generate face proposals according to facial part response maps, which is complicated and time consuming.

There are also a variety of face detection methods that inherit the achievements from generic object detection methods. Face R-CNN [12] is based on Faster R-CNN and adopts center loss [29] to minimize the intra-class distances of the deep features. It also utilizes some training tricks such as online hard example mining and multi-scale training. CMS-RCNN [10] uses contextual information for face detection. DeepIR [13] concatenate features of multiple layers to improve face detection performance. Hu et al. [16] build image pyramids and defines multiple templates to find tiny faces. SSH [14] establishes detection modules on different feature maps to detect face in a single stage. SFD [15] focuses on scale-invariance by using a new anchor matching strategy. Zhu et al. [30] analyze the anchor matching mechanism with the proposed expected max overlap (EMO) score and introduce new designed anchors to find more tiny faces. All these anchor-based methods have obtained promising results. However, we know that the scale of faces is continuous. The anchor mechanism makes the scale discrete, which may lead to the low matching rate of hard samples, especially occluded faces. A naive way to increase the number of matching anchors is to increase the total number of anchors. But this will result in heavily computational burden.

DenseBox [20] is another kind of object detection method. Different from the above anchor-based methods, DenseBox utilizes a FCN to perform pixel-wise predictions. By doing the upsampling operation to keep a high-resolution output, it has great advantages in handling the detection of small objects. The approach of dense prediction can also improve the robustness of detecting heavy occluded objects. UnitBox [21] further presents a new intersection-over-union (IoU) loss for bounding box prediction. Yet there are some drawbacks of UnitBox. On one hand, an up-sample layer is used to perform linear interpolation to resize the feature map to the original image size. Although it can detects smaller faces, the computational cost is unacceptable. On the other hand, the feature maps are upsampled 16 times for pixel-wise classification, which may bring artifacts. In this paper, we propose a novel face detector that utilizes a FCN framework to do the dense prediction on the feature maps whose size is just 1/4 of the original image size. The FCN architecture consists of a bottom-up path and a top-down path similar to [20,31]. Inspired by [32], we further employ an in-network recurrence mechanism to explore meaningful information of the convolutional feature maps and improve the robustness of detecting faces with occlusion, leading to state-of-the-art detection performance.

3 Proposed Method

The proposed face detector is trained to directly predict the existence of faces and their locations from full images instead of dividing the detection task into bounding box proposal and classification. A fully convolutional neural network is used to do the pixel-wise dense prediction of faces. The post-processing of our method is quit simple, which only contains thresholding and NMS.

3.1 Base Framework

As we know from [33] that feature maps of different layer represent different semantic information. The shallow layers have high spatial resolution responding to corners and edge/color conjunctions, which is good for spatial localization. The deep layers have lower spatial resolution but more class-specific which is good for classification. Inspired by recent works [20,31,34], we adopt a neural network that contains a top-down architecture with lateral connection to fuse features from different layers.

Our network architecture is shown in Fig. 2. We use PVANet [35] as the backbone. The bottom-up pathway is the feed-forward computation of the backbone ConvNet generating four levels of feature maps, whose sizes are 1/4, 1/8, 1/16 and 1/32 of the original image, respectively. We define that layers producing the output maps of the same size are in the same network stage. Since the deeper layer should have stronger features, the last layer of each stage is chosen to connect with deeper layer with the same output size. It is very difficult to detect tiny object by low resolution features. The top-down pathway increases the resolution by upsampling operations while keeps the semantic information. Each

upsample operation is at a scaling step of 2. The top-down pathway features are enhanced by features from the bottom-top pathway via lateral connections. By doing such lateral connections, the network can maintain both geometrical and semantic information. As shown in Fig. 2, we use a 1×1 conv layer to preprocess the lateral features and merge different features by concat layer. Then a 1×1 conv layer and a 3×3 conv layer are used to further cut down half of the number of channels and produce the output of this merging stage, respectively. The size of the final feature maps is only 1/4 of the original image, making the network computation-efficient. The network is then split into two branches, one for classification and the other one for bounding box regression.

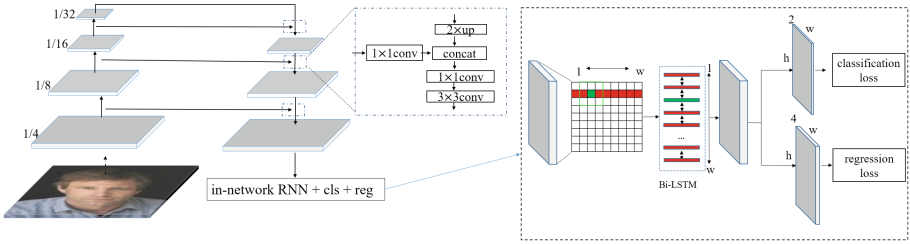


Fig. 2. An overview of our network architecture

3.2 In-Network Recurrence Architecture

Recurrent neural network (RNN) is often applied in scenarios with sequences of inputs such as video, audio, text lines to encode the contextual information. Recent work [32] has shown that the sequential context information is good for text detection. Motivated from this work, we believe that RNN may also benefit for face detection, especially detecting faces with occlusion. We note that features of the face area are highly-correlated, so we can use this correlation via recurrent structure to make correct predictions of the occluded part of face. Besides, the regression task predicts a 4-D distance vector (the distances between the current pixel and the four bounds of the ground truth box), and there is also a strong correlation among the distance vectors of adjacent pixels. RNN can encode these contextual information recurrently using its hidden layers. Formally, The internal state of RNN at t moment is given by

$$H_t = \varphi(H_{t-1}, X_t) \tag{1}$$

where $X_t \in R^{3\times3\times C}$ is the input sequential features from t-th sliding-window (3×3) as shown in Fig. 2. The sliding window slides from left to right at a stride of 1, generating $t = 1, 2, ...W$ sequential inputs for each row. W is the width of the input feature map. In this paper, we adopt the bi-directional long short-term memory (Bi-LSTM) architecture for the RNN layer just as [32] do. The Bi-LSTM allows the model to encode the contextual features in both directions. The outputs of the two inverse LSTMs is then merged by a concat layer, followed by a 1×1 conv layer to cut down the number of channels.

3.3 Label Generation

We consider the face area is a rectangle. The classification task is to predict a binary score map $\in \{0, 1\}$ which indicates the negative area and positive area. The positive area of the rectangle on the score map is designed to be roughly a shrunk version of the original rectangle. For each edge, we shrink it by moving its two endpoints inward along by 0.2 of its length, illustrated in Fig. 3(a). The regression task is to predict a 4 channels of distance map as shown in Fig. 3(d). The ground truth distance map is generated by calculating a 4-D distance vector for each pixel with a value of 1 on the score map, illustrated in Fig. 3(c).

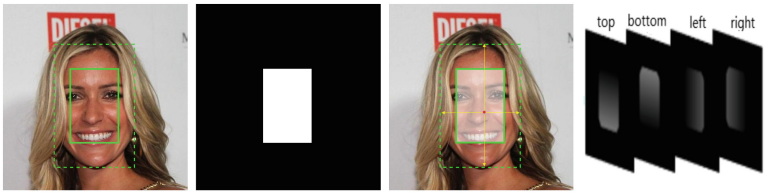


Fig. 3. Label generation. (a) Face bounding box (green dashed) and the shrunk rectangle (green solid); (b) score map; (c) pixel-wise distances generation; (d) 4 channels of distances of each pixel to rectangle boundaries. (Color figure online)

4 Training

In this section, we introduce our training details, including loss function, training dataset, data augmentation and other implementation details.

4.1 Loss Functions

Considering that there is a class imbalance problem, we restrict the number of positive pixels and negative pixels during training, making them numerically equal. This can be done by hard examples mining. We simply use softmax loss for the classification. The regression task is optimized by IoU loss, more details can be found in [21]. These two tasks are joint optimized equally. The multi-task loss is formulated as

$$L = L_{cls} + L_{IoU} \tag{2}$$

We empirical note that model optimized by Eq. 2 has a problem in locating tiny faces, leading to lots of false positives. We solve this problem by employing a focal loss to focus training on locating tiny face. The new loss function can be rewritten as

$$L = L_{cls} + \alpha S^{-\gamma} L_{IoU} \tag{3}$$

where S is the face area, α and γ are two constant. In our experiments, we empirically set $\alpha = 4$,$\gamma = 0.5$.

4.2 Training Dataset and Data Augmentation

We use the WIDER FACE training set which contrains 12,880 images to train our model. In order to get better results, we also apply the following data augmentation techniques: (1) **Scale modification.** Each image is random scaling in a range between [0.6, 2] via bilinear interpolation. (2) **Random crop.** We randomly crop a square patch from the image. And the size of the image patch is 640×640. For images with shorter side less than 640 pixels, we firstly pad the images with 0, making their shorter side greater than 640. (3) **Horizontal flip.** After random crop, we obtain 640×640 image patch, and then we horizontally flip it with probability of 0.5.

4.3 Other Implementation Details

Online hard examples mining is employed to boost the performance of the model. For the parameter initialization, the parameters of the backbone are initialized from the corresponding pre-trained models. We use PVANet as the backbone in our experiments. Other additional layers are randomly initialized with the "xavier" method. All models are trained by SGD with a single GPU. The mini-batch sizes of models are 6, because of the GPU memory limitation. Weight decay is 1e–5 and momentum is 0.9. Our networks are trained for 500 K iterations. The initial learning rate is 0.001 and drops by a factor of 5 after 200 K iterations. During inference, the score threshold is set to 0.01 and NMS with a threshold of 0.3 is performed on the predicted bounding boxes.

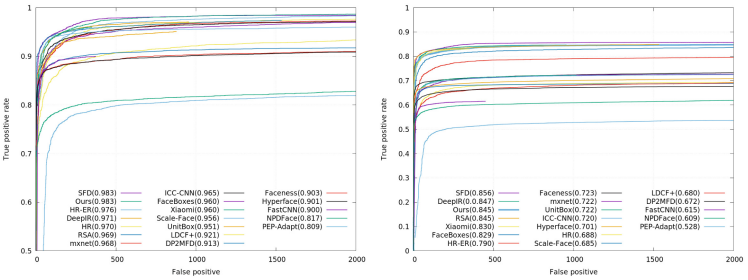
5 Experiments

5.1 Evaluation on Benchmark

We compare the proposed method with existing methods on two common face detection benchmarks: FDDB, WIDER FACE.

FDDB. It contains 2845 images with 5171 annotated faces. The Evaluation criteria include discrete score and continuous score. We compare our face detector against the state-of-the-art methods. Figure 4 shows the results. Our Face detector achieves competitive results with SFD [15] and outperforms other methods, indicating that our method can robustly detect unconstrained faces.

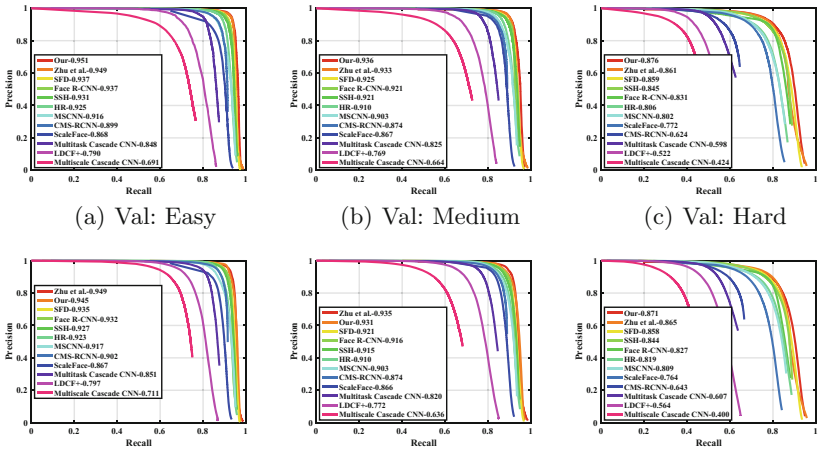
WIDER FACE. It contains 32203 images with a total of 393703 annotated faces with different scales, poses and occlusions. The data set is divided into training (40%), testing (50%) and validation (10%) set. Faces in the testing and validation set are split into three kinds of difficulty (easy, medium and hard). It is one of the most challenging face data sets. Our face detector is trained on WIDER FACE training set and tested on both validation and test set. We set the long side of the test image to 800, 1120, 1400, 1760 and 1920 for multi-scale



(a) Discrete score (b) Continuous score

Fig. 4. Evaluation on FDDB

testing. Figure 5 illustrates the precision-recall curves along with AP scores. Our face detector outperforms other recent published methods including Zhu et al. [30], SFD [15], SSH [14] on the validation set and achieves competitive results with Zhu et al.'s [30], which demonstrate that the proposed method has a strong capacity in detecting small and hard faces.



(a) Val: Easy (b) Val: Medium (c) Val: Hard

(d) Test: Easy (e) Test: Medium (f) Test: Hard

Fig. 5. Precision-recall curves on WIDER FACE validation and test sets.

5.2 Robustness to Occlusion

We further explore the ability of our detector in detecting occluded faces. To demonstrate the effectiveness of LSTM, we carry out comparative experiments with Two models: PVA, PVA+LSTM, where PVA uses PVANet [35] as the

backbone without Bi-LSTM architecture. Two occluded face data sets are used for this purpose, i.e. WIDER FACE validation set with artificial occlusion and MAFA with real occlusion. We also compare our method with other algorithms that release their trained models and testing codes such as MTCNN [18], SFD [15], SSH [14].

Faces with Artificial Occlusion. In this experiment, We generate a new occluded face data set by blacking a rectangle area on every faces of the WIDER FACE validation set. The rectangle black is randomly distributed in the left, right and bottom side of the face, accounting for 40% area of the face annotated box. Examples of occluded images are shown in Fig. 6. Table 1 shows the results of different methods. It's clear that our two models outperform other methods. We note that adding LSTM or not makes little difference. The main reason is that the WIDER Face contain lots of tiny face, the role of encoding the context information of the RNN structure is weakened after adding the artificial occlusion.

Fig. 6. Examples of WIDER FACE validation set with Occlusion

Table 1. Comparison of different models on the WDIER FACE validation set with artificial occlusion.

Methods	AP (easy)	AP (medium)	AP (hard)
MTCNN [18]	0.565	0.526	0.361
SSH [14]	0.801	0.768	0.625
SFD [15]	0.835	0.798	0.621
PVA	0.881	0.850	0.723
PVA+LSTM	0.881	0.851	0.720

Faces with Real Occlusion. MAFA data set contains 30,811 image with 35,806 faces collected from the Internet. Most of the faces are occluded by mask. We only use the testing set which contains 4,935 images to evaluate our face

detector. The long side of all testing images is set to 1280. Table 2 shows the results of different methods. Our base models without LSTM have already outperform other methods. And the LSTM structure further improves the robustness of our face detectors in detecting faces with real occlusion.

Table 2. Comparison of different models on the MAFA data set.

Methods	MTCNN [18]	SSH [14]	SFD [15]	LLE-CNNs [36]	PVA	PVA+LSTM
AP	0.570	0.643	0.724	0.764	0.768	0.781

5.3 Inference Time

Although our method achieves great performance, its speed is not compromised. We employ PVANet, a light-weighted neural network, as the backbone, which greatly reduces the computational burden. We measure the speed using a GTX 1080Ti GPU and Intel Xeon E5-2620 v4@2.1 GHz CPU. Table 3 shows the inference time and AP with respect to different input sizes of our face detector. The max size stands for the long side of the input image while keeping the aspect ratio.

Table 3. The inference time and AP with respect to different input sizes

Max size	800	1120	1440	1760	1920
AP (hard)	0.723	0.829	0.863	0.873	0.872
Time (ms)	60.7	83.9	124.9	172.9	195.0

6 Conclusions

In this paper, we propose a novel FCN-based face detector which is simple and efficient. Unlike other anchor-based methods, our face detector performs dense prediction on a single feature map, which is inherent robust in detecting occluded faces. By using the in-network RNN structure, our face detector is superior to handle the detection of occluded faces. Besides, the size of the final feature map is only 1/4 of the original image, reducing the computational cost while achieving remarkable results in detecting small faces. The experiments demonstrate that the proposed method achieves the state-of-the-art performance on the challenging face detection benchmarks, especially for small faces and occluded faces.

References

1. Viola, P., Jones, M.J.: Robust real-time face detection. IJCV **57**(2), 137–154 (2004)
2. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR, vol. (1), pp. 886–893. IEEE (2005)
3. Ojala, T., Pietikainen, M., Harwood, D.: Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. In: 1994 Proceedings of the 12th IAPR International Conference on Pattern Recognition, Computer Vision and Image Processing, vol. 1, pp. 582–585. IEEE (1994)
4. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
5. Szegedy, C., Vanhoucke, V., Ioffe, S., et al.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)
6. Girshick, R.: Fast R-CNN. arXiv preprint arXiv:1504.08083 (2015)
7. Ren, S., He, K., Girshick, R., et al.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in neural information processing systems, pp. 91–99 (2015)
8. Redmonm, J., Divvala, S., Girshick, R., et al.: You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788 (2016)
9. Liu, W., Anguelov, D., Erhan, D., et al.: SSD: single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
10. Zhu, C., Zheng, Y., Luu, K., Savvides, M.: CMS-RCNN: contextual multi-scale region-based CNN for unconstrained face detection. In: Bhanu, B., Kumar, A. (eds.) Deep Learning for Biometrics. ACVPR, pp. 57–79. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61657-5_3
11. Jiang, H., Learned-Miller, E.: Face detection with the faster R-CNN. In: 2017 12th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2017), pp. 650–657. IEEE (2017)
12. Wang, H., Li, Z., Ji, X., et al.: Face R-CNN. arXiv preprint arXiv:1706.01061 (2017)
13. Sun, X., Wu, P., Hoi, S.C.H.: Face detection using deep learning: an improved faster RCNN approach. arXiv preprint arXiv:1701.08289 (2017)
14. Najibi, M., Samangouei, P., Chellappa, R., et al.: SSH: single stage headless face detector. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4875–4884 (2017)
15. Zhang, S., Zhu, X., Lei, Z., et al.: S^3FD: single shot scale-invariant face detector. arXiv preprint arXiv:1708.05237 (2017)
16. Hu, P., Ramanan, D.: Finding tiny faces. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1522–1530 (2017)
17. Li, H., Lin, Z., Shen, X., et al.: A convolutional neural network cascade for face detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5325–5334 (2015)
18. Zhang, K., Zhang, Z., Li, Z.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Sig. Process. Lett. **23**(10), 1499–1503 (2016)
19. Qin, H., Yan, J., Li, X., et al.: Joint training of cascaded CNN for face detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3456–3465 (2016)

20. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
21. Yu, J., Jiang, Y., Wang, Z., et al.: UnitBox: an advanced object detection network. In: Proceedings of the 2016 ACM on Multimedia Conference, pp. 516–520. ACM (2016)
22. Yang, B., Yan, J., Lei, Z., et al.: Aggregate channel features for multi-view face detection. In: IEEE International Joint Conference on Biometrics (IJCB), pp. 1–8 (2014)
23. Zhu, Q., Yeh, M.C., Cheng, K.T., et al.: Fast human detection using a cascade of histograms of oriented gradients. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 1491–1498 (2006)
24. Liao, S., Jain, A.K., Li, S.Z.: A fast and accurate unconstrained face detector. IEEE Trans. Pattern Anal. Mach. Intell. **38**(2), 211–223 (2016)
25. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., et al.: Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. **32**(9), 1627–1645 (2010)
26. Brubaker, S.C., Wu, J., Sun, J., et al.: On the design of cascades of boosted ensembles for face detection. Int. J. Comput. Vis. **77**(1–3), 65–86 (2008)
27. Pham, M.T, Cham, T.J.: Fast training and selection of HAAR features using statistics in boosting-based face detection. In: IEEE International Conference on Computer Vision (ICCV), pp. 1–7 (2007)
28. Yang, S., Luo, P., Loy, C.C., et al.: From facial parts responses to face detection: a deep learning approach. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3676–3684 (2015)
29. Wen, Y., Zhang, K., Li, Z., Qiao, Y.: A discriminative feature learning approach for deep face recognition. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9911, pp. 499–515. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46478-7_31
30. Zhu, C., Tao, R., Luu, K., et al.: Seeing small faces from robust anchor's perspective. arXiv preprint arXiv:1802.09058 (2018)
31. Zhou, X., Yao, C., Wen, H., et al.: EAST: an efficient and accurate scene text detector. arXiv preprint arXiv:1704.03155 (2017)
32. Tian, Z., Huang, W., He, T., He, P., Qiao, Y.: Detecting text in natural image with connectionist text proposal network. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 56–72. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_4
33. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 818–833. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10590-1_53
34. Lin, T.Y., Dollr, P., Girshick, R., et al.: Feature pyramid networks for object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, no. 2, p. 4 (2017)
35. Kim, K.H., Hong, S., Roh, B., et al.: PVANET: deep but lightweight neural networks for real-time object detection. arXiv preprint arXiv:1608.08021 (2016)
36. Ge, S., Li, J., Ye, Q., et al.: Detecting masked faces in the wild with LLE-CNNS. In: The IEEE Conference on Computer Vision and Pattern Recognition (2017)