Multi-Task FaceBoxes: A Lightweight Face Detector Based on Channel Attention and Context Information

Shuaihui Qi[1*], Jungang Yang[1], Xiaofeng Song[1*], Chen Jiang[1]
[1] National University of Defense Technology, Xi'an 710106, Shanxi, China
[e-mail: 18883381539@163.com, xiaofengsong@sina.com]
*Corresponding author: Shuaihui Qi, Xiaofeng Song

Received May 19, 2020; revised August 8, 2020; accepted September 21, 2020; published October 31, 2020

Abstract

In recent years, convolutional neural network (CNN) has become the primary method for face detection. But its shortcomings are obvious, such as expensive calculation, heavy model, etc. This makes CNN difficult to use on the mobile devices which have limited computing and storage capabilities. Therefore, the design of lightweight CNN for face detection is becoming more and more important with the popularity of smartphones and mobile Internet. Based on the CPU real-time face detector FaceBoxes, we propose a multi-task lightweight face detector, which has low computing cost and higher detection precision. First, to improve the detection capability, the squeeze and excitation modules are used to extract attention between channels. Then, the textual and semantic information are extracted by shallow networks and deep networks respectively to get rich features. Finally, the landmark detection module is used to improve the detection performance for small faces and provide landmark data for face alignment. Experiments on AFW, FDDB, PASCAL, and WIDER FACE datasets show that our algorithm has achieved significant improvement in the mean average precision. Especially, on the WIDER FACE hard validation set, our algorithm outperforms the mean average precision of FaceBoxes by 7.2%. For VGA-resolution images, the running speed of our algorithm can reach 23FPS on a CPU device.

Keywords: Multi-Task FaceBoxes, Feature Fusion, Attention, Landmark Detection

http://doi.org/10.3837/tiis.2020.10.009 **ISSN : 1976-7277**

1. Introduction

With the advent of mobile Internet, smart mobile devices are becoming more and more popular. To enhance security, face detection is widely used as a personal identity authentication tools in these devices. Moreover, face detection is also increasingly used in so many scenes, such as age progression [1, 2], facial emotion recognition [3], and face tracking [4]. Therefore, the design of face detectors running mobile devices is attracting more and more attention. The difficulty is that the face detector needs low computation cost and high detection precision.

Modern approaches to face detection can be divided into two ways according to categories of extracting features: hand-crafted features and extracted by CNN. Previous face detection algorithms generally design hand-crafted features and use a classifier to judge whether the proposal is a face or not. Viola and Jones [5, 6] uses simple Haar-like features and cascaded adaboost classifier to construct the face detector. After that, many outstanding works using new features [7, 8], new boosting methods [9, 10], or cascade framework [11, 12] are raised. Besides these, the structural models [13], and DPM [14, 15] are introduced to improve performance. Since these algorithms are structural or detecting component of target, it is not greatly affected by distortion, posture, and angle.

In recent years, convolutional neural networks (CNN) have become the primary choice for face detection. CascadeCNN [16] designs a cascade architecture based on CNN and achieve high performance. UnitBox [17] introduces a new loss function about intersection over-union. PyramidBox [18] proposes a novel context-assisted single shot face detector. DSFD [19] realizes face detection by adding feature enhance module, proposing progressive anchor loss, and designing an improved anchor matching strategy. Although they can achieve higher precision, the fatal problem is that they are time-resuming badly with a heavy model. Besides these large networks, the lightweight algorithms, such as MTCNN [20], FaceBoxes [21], libfacedetecion [22], ULFG [23], DBFace [24], and RetinaFace [25], are developed to solve the contradiction between speed and precision. However, MTCNN is a cascade framework and can not be trained easily. FaceBoxes is an end-to-end fully convolutional neural network and it can achieve a real-time speed on a CPU, however FaceBoxes still has room for improvement in its performance. A super-lightweight model is utilized in ULFG and runs at a fast speed with normal accuracy. DBFace, libfacedetection, and RetinaFace introduce five facial landmarks into networks and get a powerful discriminative capability for small faces. However, the attention between the channels cannot be considered in these algorithms.

To improve the detection performance, we develop a lightweight and multi-task face detector named Multi-task FaceBoxes, which is an improved version of FaceBoxes [21]. Compared with FaceBoxes, we add squeeze and excitation modules to extract attention between channels and modify the network structure to get the enriched feature by fusing context information. Finally, a landmark detection module is added to improve the detection performance for small faces. For our detector adopts a fully connected network, its speed is unrelated to the number of faces. Our face detector can reach to 23FPS(Frame Per Second) on a CPU device. We evaluate our algorithm on the benchmark datasets such as AFW, PASCAL face, FDDB, and WIDER FACE validation datasets and it can achieve the competitive detection performance in contrast to the existed algorithms.

Our contributions can be summarized as follows:

· The network structure of FaceBoxes is optimized by using squeeze and excitation module to extract attention between different channels.

· The rich face features are extracted by fusing the texture information of the shallow networks and semantic information of the deep networks.

· The recall rate of small faces is improved by adding the landmark detection module and the multi-task learning is realized because facial landmarks are provided at the same time.

· The aspect ratio of the train dataset is counted and taken into training procedure to make the detection performance more stable.

2. Multi-Task FaceBoxes

2.1 FaceBoxes

FaceBoxes uses a single-stage network structure [26, 27], which is used to achieve real-time and high accuracy on the CPU. It is mainly divided into three parts: rapid digestion convolution layer, multi-scale convolution layer, positioning and classification layer. The network is shown in **Fig. 1**.

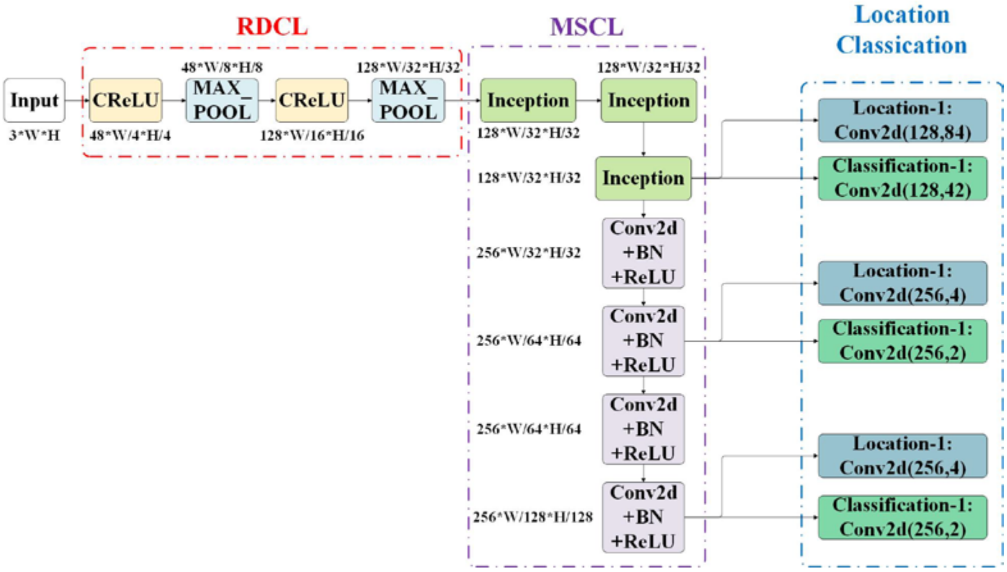


Fig. 1. Architecture of FaceBoxes.

Rapid Digestion Convolution Layer: this module is designed to accelerate the reduction of feature map size, retain more feature information and make this algorithm running in real-time. This module includes two C.ReLU layers and two max_pooling layers. The C.ReLU layer consists of the Convolution and Batch Normalization layer and then utilizes C.ReLU activation function to double the number of output channels by concatenating negated outputs before applying ReLU [21]. The stride size of these four layers is 4, 2, 2, and 2, respectively and the kernel size is 7*7, 3*3, 5*5, and 3*3. However, a fast shrink of the feature map causes the loss of feature information.

Multi-Scale Convolution Layer: To solve weak features and resolution of RPN, Inception [29], basic convolution operation, which consists of Convolution, Batch Normalization, and

ReLU activation function, are utilized to generate more default anchors with different resolutions. And Inception modules are used to learn visual patterns for different scales of faces, it consists of multiple convolution branches with different kernels and enriches the receptive fields [21].

Location and Classification Layer: Convolution layers are used to get scores and regression results of anchors. Cross-entropy loss function and smooth L1 loss function are used to calculate classification loss and regression loss during training.

2.2 Network Structure of Multi-Task FaceBoxes

Multi-Task FaceBoxes turns from FaceBoxes and the architecture is shown in **Fig. 2**.

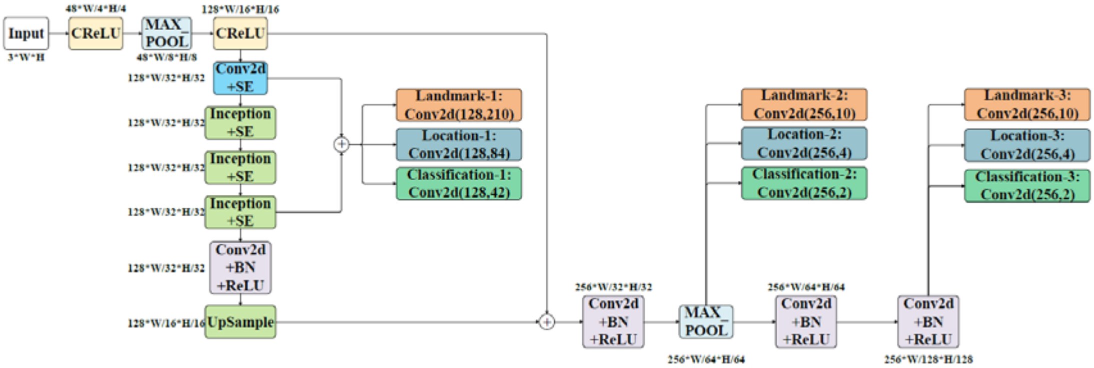


Fig. 2. Architecture of Multi-Task FaceBoxes.

2.2.1 Channel Attention Extraction

Since different channels have different significance, so a squeeze and excitation module [30] is used to extract channel attention for each channel. This module can be divided into squeeze and excitation two parts. The squeeze operation uses global pooling to map input features of size C×H×W to C×1×1 feature descriptors. Generally, the maximum pooling or average pooling method is used; the excitation operation includes the fully connected layers and sigmoid activation function. The fully connected layer is used to fuse all the input feature information, and the sigmoid function can map the input to 0~1, that is, the weight of each channel. Finally, the weight is multiplied by the input features to filter out the attention of each channel. The squeeze and excitation module in Multi-Task FaceBoxes architecture is labeled with “SE” in **Fig. 2** and its specific definition is shown in **Fig. 3**. There is a hyper-parameter needs to be set in the squeeze and excitation module that is the *reduction* and it is set to 6 in RDCL and 16 in MSCL.

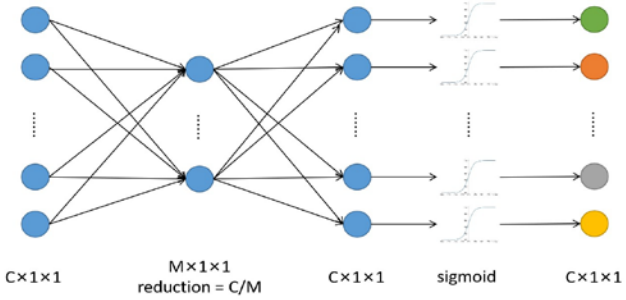


Fig. 3. Squeeze and excitation module.

2.2.2. Context Feature Fusion

In order to achieve fast convolution, the input image is quickly digested by 32 times after through RDCL and MSCL, which leads to a lot of information losing. Therefore, we can design a module to improve the completeness of features.

As we know, the texture information of the shallow network and the semantic information of the deep network are rich. Therefore, we add two branches to fuse the features. One branch is realized by fusing the output of the RDCL and the output by the third Inception module after the squeeze and excitation module. Compared to FaceBoxes, the texture information of the shallow network is directly added to provide abundant features for face location and classification. The other branch is fusing the output of the second C.ReLU module and the output of the upsampling module. The fused features are propagated along the backbone to get more stable and accurate deep features so that achieve better detection performance. As shown in **Fig. 2**, One branch is realized by fusing the output of the RDCL and the output by the third Inception module after the squeeze and excitation module. The other branch is fusing the output of after the second C.ReLU module and the output of the upsampling module. In order to make the network to learn more deep features and keep more features, the Max-Pooling layer after the second C.ReLU module is replaced with a convolution layer. In the fusing module, an upsampling module is used to deconvolution feature maps, so a max-pooling layer is used to downsample.

2.2.3. Landmark Detection

In [20], the author believes that there is a potential relationship between face detection and landmark detection. Therefore, in order to make full use of this potential connection, the author proposes a multi-task cascading face detection framework to realize face detection and landmark detection simultaneously. Therefore, in order to verify this potential connection with FaceBoxes, this paper adds a landmark detection module to assist detection performance and provide landmark data for face alignment. The landmark detection utilizes a convolutional layer to accomplish like location and classification in **Fig. 2**.

2.3. Loss function

In Multi-Task FaceBoxes, the loss function includes three parts: location loss, classification loss, and landmark loss. The loss function is as follows:

$$\mathrm{Loss} = \frac{\lambda_1}{N}\sum_{i=1} L_{classify}(label_i, label_i^{gt}) + \frac{\lambda_2}{N}\sum_{j=1} L_{locate}(bbox_j, bbox_j^{gt}) + \frac{\lambda_3}{N}\sum_{j=1} L_{landmark}(landmark_j, landmark_j^{gt}) \quad (1)$$

Where N is batch size in training， $L_{classify}$ represents classification loss which use cross-entropy loss function, $label_i$ represents the score of anchor, $label_i^{gt}$ represents the ground truth of this anchor, which value is 0 or 1. L_{locate} and $L_{landmark}$ represent the bounding box location loss and landmark loss [25] respectively, and they are calculated using the smooth L1 loss function under the condition that $bbox_j$ and $landmark_j$ are positive. $bbox_j^{gt}$ and $landmark_j^{gt}$ represent translation parameters between anchor and ground truth. The landmark parameters need to learn is defined as follows:

$$t_x = (G_x - a_{x\_center}) / a_w \quad , \quad t_y = (G_y - a_{y\_center}) / a_h \quad (2)$$

(G_x, G_y) represents landmark coordinate， $(a_{x\_center}, a_{y\_center}, a_w, a_y)$ represents anchor coordinate. λ_1, λ_2 and λ_3 represents weight of $L_{classify}$、 L_{locate} and $L_{landmark}$.

2.4. Training

Training dataset: Our model is trained on the WIDER FACE training dataset, which contains 12880 images. Facial landmarks training dataset is labeled by RetinaFace [25].

Training details: In order to verify the effectiveness of our algorithm, the same parameter with the FaceBoxes is used:

(1) The same method of data augmentation is applied to process training images and the most important is the face boxes whose height and width are less than 20 pixels are abandoned.

(2) Batch size is set to 32, λ_1=2, λ_2 =1, λ_3=0.1.

(3) "Xavier" is used to initialize the model parameters and momentum stochastic gradient descent optimizer (momentum = 0.9, decay = 5e-4) is utilized to optimize model.

(4) We run 200 epochs with the learning rate is 0.001, and then run the last 100 epochs with the learning rate decayed 10 times after every 50 epochs.

(5) Before we train our model on WIDER FACE training dataset, the minimum between width and height and the aspect ratio of all ground truth in FDDB and WIDER FACE training dataset is counted and statistical normalized results are shown in **Fig. 4**. These two datasets have many small faces that aspect ratio is about 0.8. So, we set the width to 0.8 times of height when generating anchors.

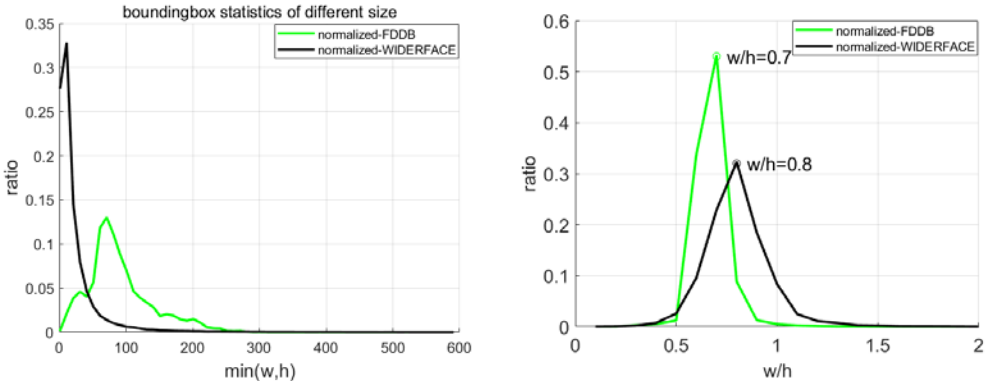


Fig. 4. Statistical result of min(w,h) and w/h.

3. Experiments

The source code and pre-trained model of FaceBoxes can be downloaded at this address: *https://github.com/zisianw/FaceBoxes.PyTorch*. The data reproduced in this article are tested on ’FaceBoxesProd.pth’ provided by the official website. Our model is tested on the FDDB dataset [31] for analyzing model performance and take the true positive rate at 1000 false positives as the evaluation criterion.

3.1. Parameter Selection

3.1.1. Filter size

From [21], we can know that the author filters out face boxes whose height and width are less than 20 pixels and it reduces the scale of the training dataset. Maybe it will cause algorithm degradation. So a control-experiment is made to select an optimal training parameter. As shown in **Table 1**, when we do not filter any faces, it can make better results than FaceBoxes on FDDB datasets.

Table 1. Comparison of different filter size.

Filter Size	FDDB_Disc	FDDB_Cont
min(w,h) <= 20	0.953	0.724
min(w,h) <= 0	0.955	0.724

3.1.2. Reduction parameter

When adding squeeze and excitation module, because the number of feature channels after the first C.ReLU module is only 48 layers, and the number of channels after the inception module is 128 layers (reduction=16). So we reduced reduction to 6 and 16 respectively in RDCL. The results are shown in **Table 2**.

Table 2. Comparison of different reductions.

Reduction	FDDB_Disc	FDDB_Cont
SE_Module(6)	0.958	0.728
SE_Module(16)	0.955	0.724

Experiments show that it has a better result when the *reduction* is 6. The reason is that the excitation operation is a fully connected layer, and reduction represents the scale factor of input size, which is inversely proportional to the number of hidden layer neurons. Therefore, the smaller the reduction, the more hidden layer neurons, the stronger the ability to express the channel attention, the better the effect.

3.2. Ablative Results

To prove effectiveness of our contributions, we carried out ablative experiments to analyze our model based on the FaceBoxes network. The results are shown in **Table 3**.

Table 3. Results of Multi-Task FaceBoxes on FDDB dataset.

Module	FaceBoxes	Multi-Task FaceBoxes			
Attention		√	√	√	√
Feature Fusion			√	√	√
Landmark				√	√
w/h=0.8					√
FDDB_Disc	0.953	0.958	0.960	**0.963**	**0.963**
FDDB_Cont	0.724	0.728	0.731	0.731	**0.732**

Experiments show that all of our contributions can improve performance and when adding all of the modules, the result indicates that our Multi-Task FaceBoxes can improve 1% and 0.8% on discontinuous and continuous ROC curves, respectively.

3.3. Evaluation on Benchmark

We test our algorithm on the AFW [32], FDDB [31], PASCAL face [13] and WIDER FACE validation set [33]. There is a question needs to pay attention to that is since our algorithm adds the context feature fusion module, the size of the input image needs to be expanded when sending it into the network, and the width and height of the input image must be divisible by 64. because the picture is reduced by 32 times before the upsampling, so expanding input image can avoid inconsistency of feature map size when fusing features.

AFW dataset [32]: it contains 205 images, including 473 faces. Compared our work with classical face detection algorithm [21, 8, 34, 24, 35, 13, 36, 32], commercial algorithm (such as Face.com, Picasa and Face ++) and famous lightweight algorithms [22, 23, 24] in GitHub, as shown in **Fig. 5**, our algorithm achieves the best performance.

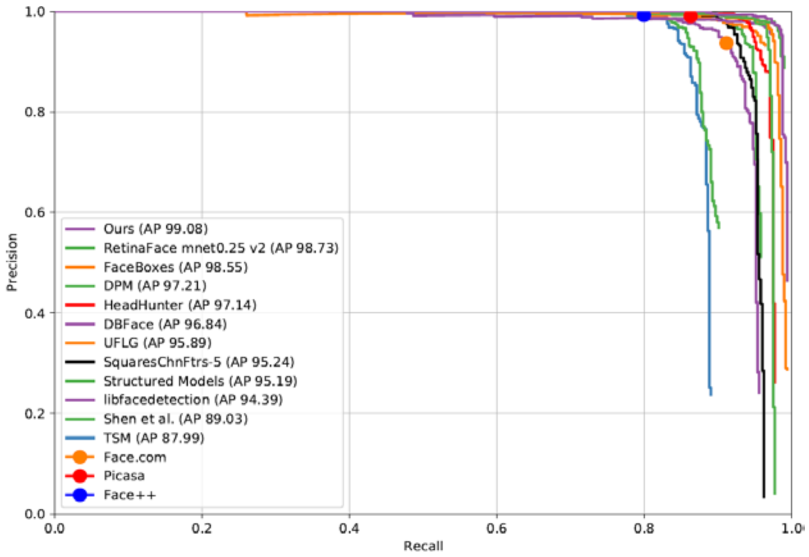
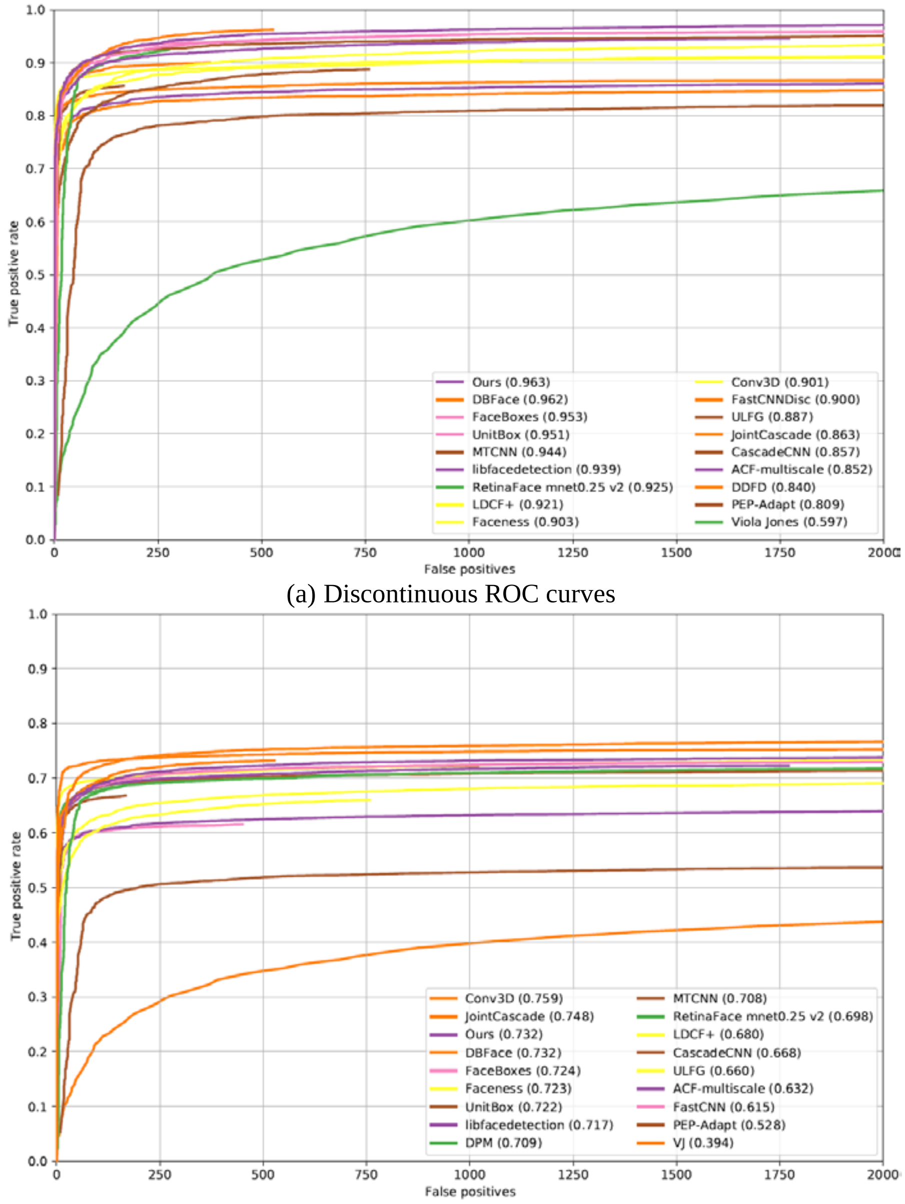


Fig. 5. Precision-recall curves on AFW dataset.

FDDB dataset [31]: it contains 5171 faces from 2845 images obtained by Faces in Wild Dataset. We compared the algorithm against the well-known algorithms [24, 21, 17, 20, 22, 37, 38, 39, 40, 23, 41, 16, 7, 42, 43, 5]. The results are shown in **Fig. 6**. Our algorithm realizes a remarkable performance.



(a) Discontinuous ROC curves

(b) Continuous ROC curves

Fig. 6. Evaluation on the FDDB dataset.

PASCAL face dataset [13]: it has a total of 851 images and 1335 faces. Our model can outperform other algorithms [22, 24, 23, 8, 34, 36, 32, 44] and some commercial algorithms(such as Sky Biometry, Picasa, and Face++). But it is lower than FaceBoxes. The reason can be seen from section 3.4.

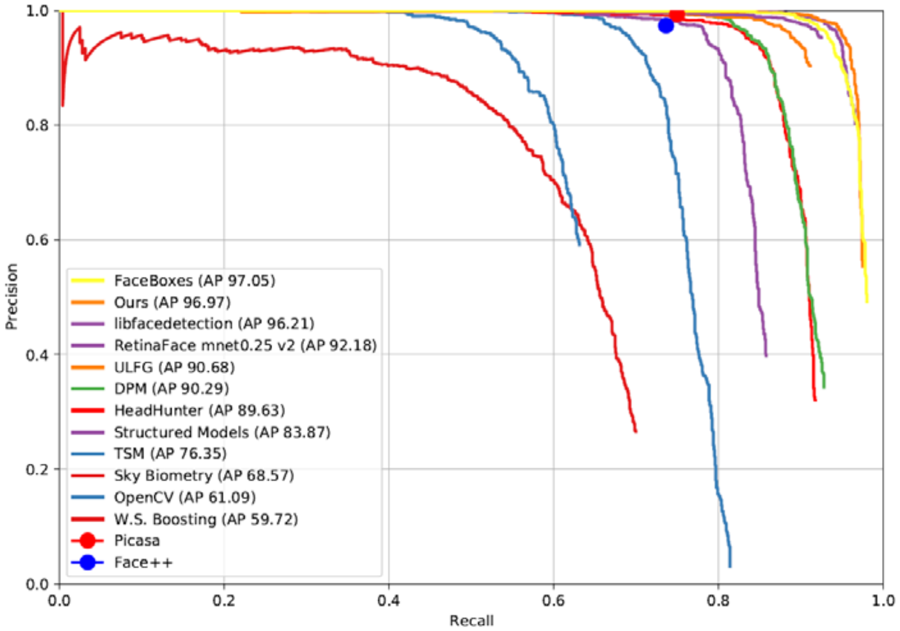
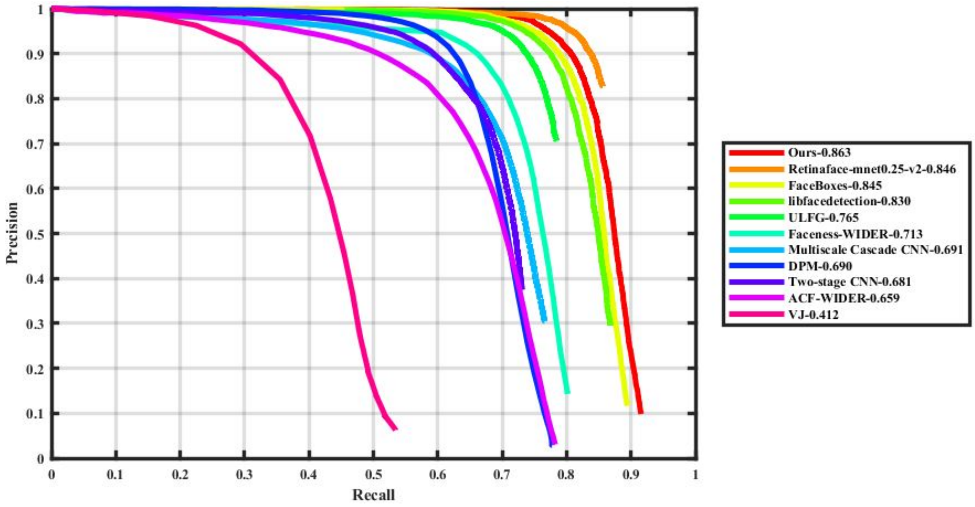
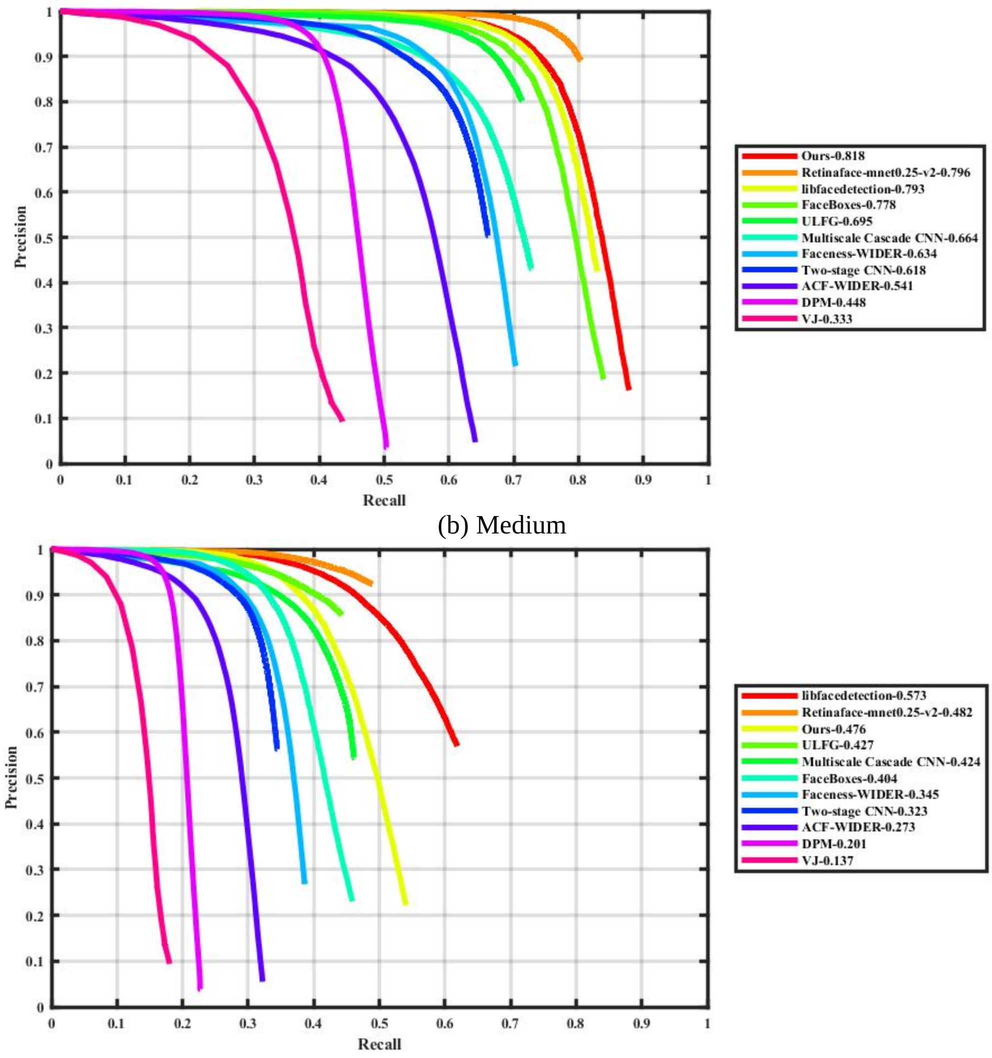


Fig. 7. Precision-recall curves on PASCAL face dataset.

WIDER FACE [33] validation dataset: it contains 3226 images. This dataset is divided into three parts: Easy, Medium, and Hard, according to the size of face boxes. Results are shown in **Fig. 8**, and the performance is better than [25, 21, 22, 23, 38, 33, 8, 7, 5].



(a) Easy



(b) Medium

(c) Hard

Fig. 8. Precision-recall curves on WIDER FACE dataset.

Compared to original FaceBoxes on the WIDER FACE validation dataset, the results of our work are improved by 1.8%, 4.0%, and 7.2% on the Easy, Medium, and Hard validation sets. It proves that our work has competitive performance.

3.4. Result analysis

Combined with the results on the WIDER FACE, we can find that the increments on Easy, Medium, and Hard datasets are increasing, especially on the Hard dataset, the improvement is very obvious. But the performance of our algorithm on the AFW dataset is not clear with an increment of 0.53% and negative growth occurs on PASCAL face dataset. Therefore, we perform statistics for the labeled face on the AFW and PASCAL face datasets, the normalized results are shown in **Fig. 9**. In the AFW and PASCAL datasets, because the proportion of small-sized faces and faces number are small. Besides, the aspect ratio mainly distributed around 1.0. These may be the reasons that our model can perform the best performance. Because FDDB and WIDER FACE datasets are larger than AFW and PASCAL face datasets, the scale of small faces is too much and the aspect ratio is set to fit FDDB and WIDER FACE datasets. So the performance improvement is more obvious. The performance on the FDDB dataset is improved by 1%, and the results on Easy, Medium, and Hard datasets are increased by 1.8%, 4%, and 7.2%, respectively. We select some representative results shown in **Fig. 10** and find our algorithm can reduce false negative effectively on FDDB datasets and upgrade the quality of detection results.

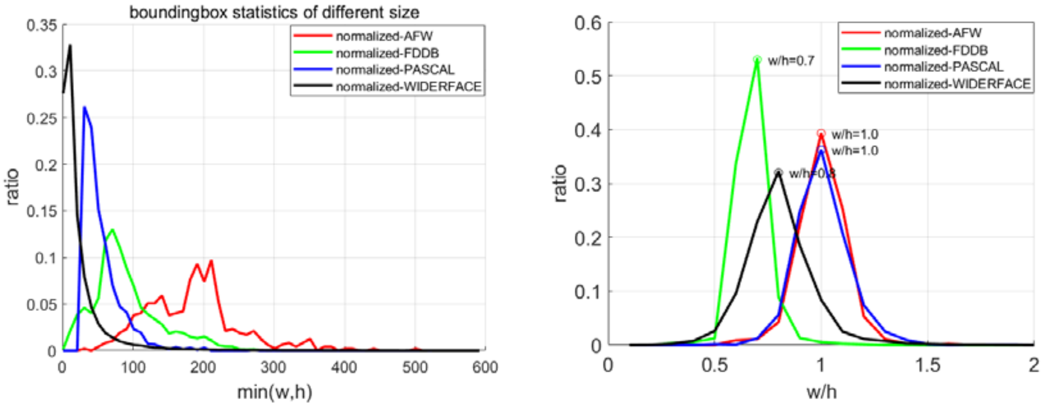


Fig. 9. Distribution of different sizes of boundingbox.

3.5. Runtime Efficiency

Because the model produces so many anchors that depend on image size, we set the score threshold 0.05 to filter anchors, and retain 400 anchors by their confidence, set the threshold of IoU to 0.3 for non-maximum suppression, and then keep 200 anchors as the detection results. We use the CPU configured by i7-8750H@2.20Hz to test video data with VGA-resolution. The results are shown in **Table 4**.

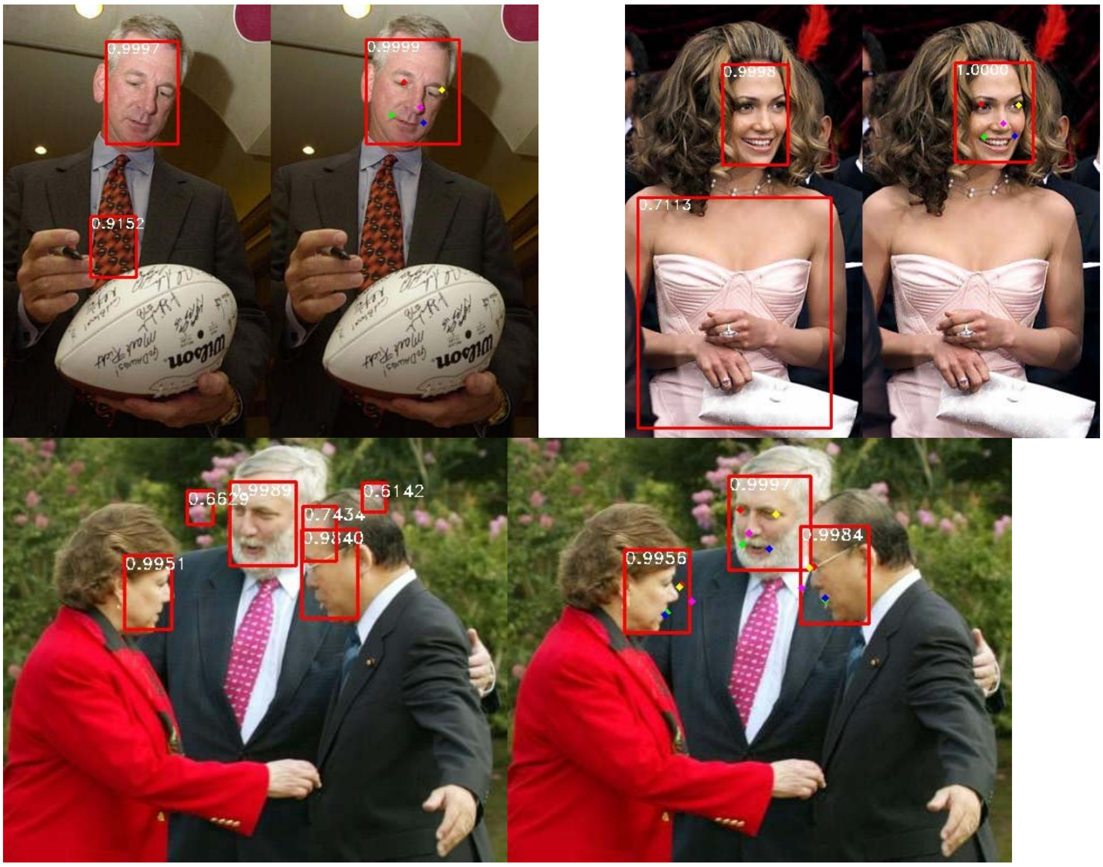
We can find that Multi-Task FaceBoxes outperforms other lightweight algorithms on detection performance. Its speed can reach to 23fps on CPU, and it can guarantee the accuracy better than the original algorithm. Meanwhile, our model is the only 5.7MB.

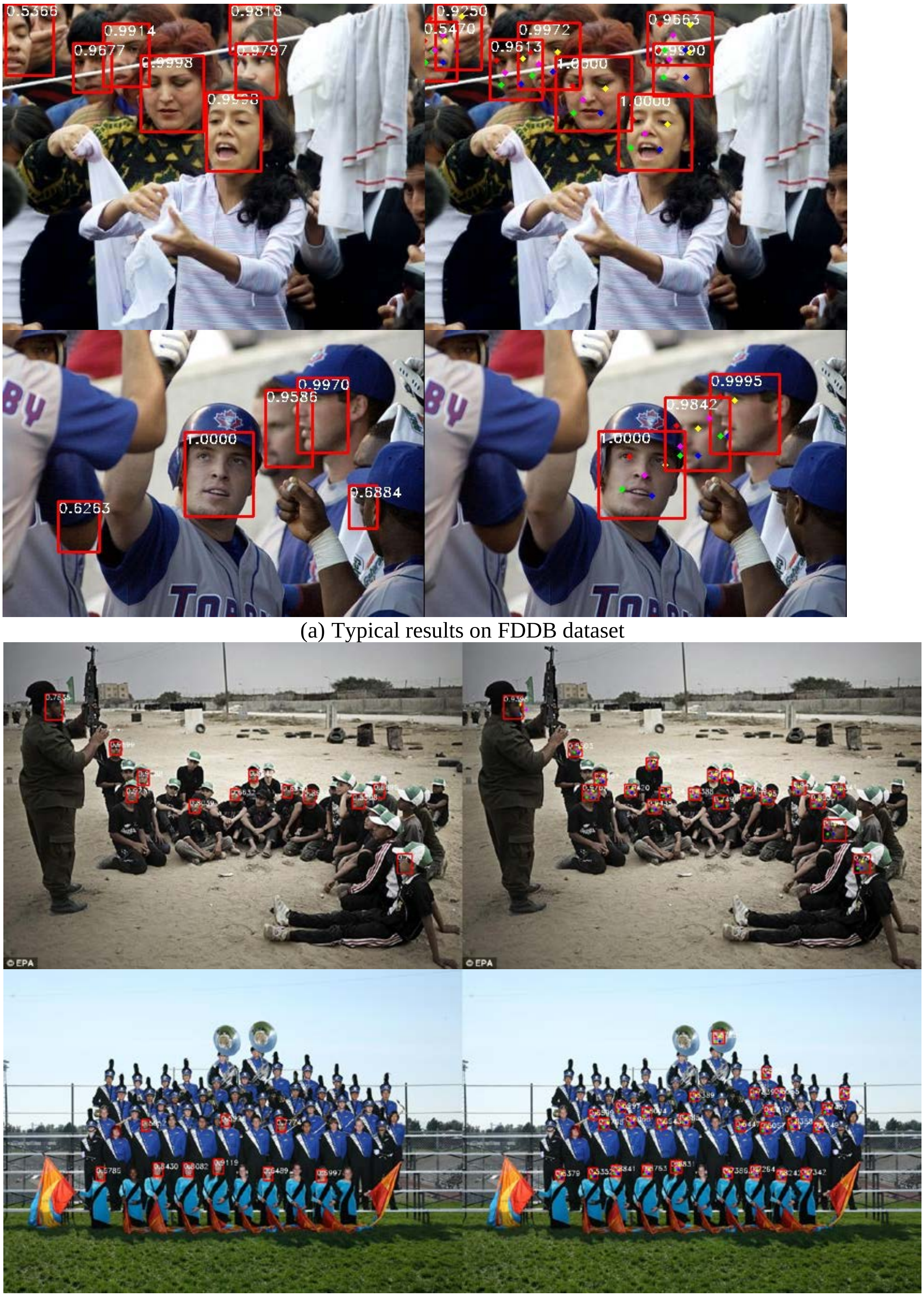
Table 4. Performance comparison.

Algorithm	CPU-Model	Weights	FDDB	Easy	Medium	Hard	FPS
Libface detection	i7-8750H@2.20	9.3M	0.939	0.830	0.793	**0.573**	15
ULFG	i7-8750H@2.20	**1.2M**	0.887	0.765	0.695	0.427	unstable
RetinaFace-mnet0.25-v2	i7-9750H@2.60	2.0M	0.925	0.846	0.796	0.482	4.5
FaceBoxes	i7-8750H@2.20	4.1M	0.953	0.845	0.778	0.404	**29**
Multi-Task FaceBoxes	i7-8750H@2.20	5.7M	**0.963**	**0.863**	**0.818**	0.476	23

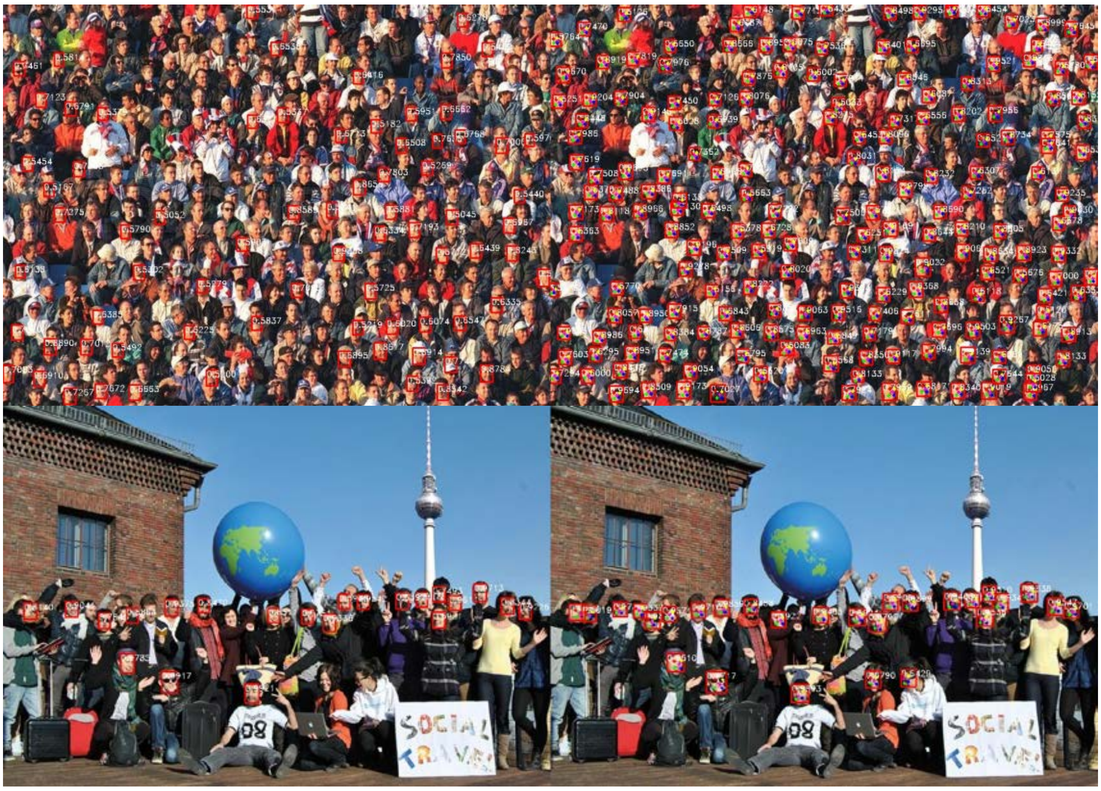
4. Conclusion

With the popularity of mobile intelligent devices, face detection is becoming a mainstream authentication method for system login, mobile payment, and so on. Moreover, face detection is also increasingly used in the other scenes, such as age progression, facial emotion recognition, face tracking, etc. Therefore, the design of lightweight models for face detection is also becoming important. The two main challenges for face detection in mobile devices are the running speed and detection precision. This paper proposes a multi-task face detector based on channel attention and context information. Moreover, the landmark detection module is integrated into the network and it not only improves the performance for the small face but also provides a landmark data for face alignment. In the experiments, the typical datasets such as AFW, FDDB, PASCAL face, and WIDER FACE are used to evaluate the detection performances of the proposed face detector. The experimental results show the proposed face detector can achieve the competitive performances in contrast to the state-of-the-art techniques. Our detector can run at the speed of 23FPS on the CPU device for a VGA-resolution image.





(a) Typical results on FDDB dataset



(b) Typical results on WIDER FACE validation dataset

Fig. 10. Results comparison between FaceBoxes and Ours.

References

[1] Shu, Xiangbo, et al., “Personalized Age Progression with Aging Dictionary,” *IEEE International Conference on Computer Vision*, pp. 3970-3978, October 8-16, 2016. Article (CrossRefLink)

[2] X. Shu, J. Tang, Z. Li, H. Lai, L. Zhang and S. Yan, “Personalized Age Progression with Bi-Level Aging Dictionary Learning,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 905-917, April, 2018. Article (CrossRefLink)

[3] H. Yang, U. Ciftci and L. Yin, “Facial Expression Recognition by De-expression Residue Learning,” in *Proc. of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2168-2177, June 18-23, 2018. Article (CrossRefLink)

[4] Ishii, Idaku, et al, “500-Fps Face Tracking System,” *Journal of Real-Time Image Processing*, vol. 8, no. 4, pp. 379–388, December, 2013. Article (CrossRefLink)

[5] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *Proc. of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 1, pp. 511–518, December 8-14, 2001. Article (CrossRefLink)

[6] Viola, P., & Jones, M, “Robust real-time face detection,” in *Proc. of Eighth IEEE International Conference on Computer Vision*, vol. 57, pp. 137-154, 2004. Article (CrossRefLink)

[7] Bin Yang, J. Yan, Z. Lei and S. Z. Li, “Aggregate channel features for multi-view face detection,” in *Proc. of IEEE International Joint Conference on Biometrics*, pp. 1-8, September 29-October 2, 2014. Article (CrossRefLink)

[8] S. Liao, A. K. Jain and S. Z. Li, “A Fast and Accurate Unconstrained Face Detector,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 211-223, Feb, 2016. Article (CrossRefLink)

[9] Brubaker, S. C., Wu, J., Sun, J., Mullin, M. D., and Rehg, J. M, “On the Design of Cascades of Boosted Ensembles for Face Detection,” *International Journal of Computer Vision*, vol. 77, pp. 65-86, September, 2008. Article (CrossRefLink)

[10] M. Pham and T. Cham, “Fast training and selection of Haar features using statistics in boosting-based face detection,” in *Proc. of 2007 IEEE 11th International Conference on Computer Vision*, pp. 1-7, October 14-21, 2007. Article (CrossRefLink)

[11] L. Bourdev and J. Brandt, “Robust object detection via soft cascade,” in *Proc. of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 236-243, June 20-25, 2005. Article (CrossRefLink)

[12] Li, S. Z., Zhu, L., Zhang, Z., Blake, A., Zhang, H., and Shum, H., “Statistical Learning of Multi-view Face Detection,” in *Proc. of ECCV ʹ02 Proceedings of the 7th European Conference on Computer Vision-Part IV*, pp. 67–81, May 28-31, 2002. Article (CrossRefLink)

[13] Junjie Yan, Xucong Zhang, Zhen Lei, and S. Z. Li, “Face detection by structural models,” in *Proc. of 2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition*, vol. 32, no 10, pp. 790-799, October, 2014. Article (CrossRefLink)

[14] P. Felzenszwalb, D. McAllester and D. Ramanan, “A discriminatively trained, multiscale, deformable part model,” in *Proc. of 2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-8, June 23-28, 2008. Article (CrossRefLink)

[15] J. Yan, Z. Lei, L. Wen and S. Z. Li, “The Fastest Deformable Part Model for Object Detection,” in *Proc. of 2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2497-2504, June 23-28, 2014. Article (CrossRefLink)

[16] Li, H., Lin, Z., Shen, X., Brandt, J., and Hua, “A convolutional neural network cascade for face detection,” in *Proc. of 2015 IEEE Conference on Computer Vision and Pattern Recognition* , pp. 5325–5334, June 7-12, 2015. Article (CrossRefLink)

[17] Yu, J., Jiang, Y., Wang, Z., Cao, Z., and Huang, T, “UnitBox: An Advanced Object Detection Network,” in *Proc. of the 24th ACM international conference on Multimedia*, pp. 516–520, October 15-19, 2016. Article (CrossRefLink)

[18] Tang, X., Du, D. K., He, Z., and Liu, J., “PyramidBox: A Context-assisted Single Shot Face Detector,” in *Proc. of the European Conference on Computer Vision*, pp. 812–828, September 8-14, 2018. Article (CrossRefLink)

[19] J. Li et al., “DSFD: Dual Shot Face Detector,” in *Proc. of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5055-5064, June 15-20, 2019. Article (CrossRefLink)

[20] K. Zhang, Z. Zhang, Z. Li and Y. Qiao, “Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499-1503, October, 2016. Article (CrossRefLink)

[21] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang and S. Z. Li, “FaceBoxes: A CPU real-time face detector with high accuracy,” in *Proc. of 2017 IEEE International Joint Conference on Biometrics*, pp. 1-9, October 1-4, 2017. Article (CrossRefLink)

[22] https://github.com/ShiqiYu/libfacedetection.train. Article (CrossRefLink)

[23] https://github.com/Linzaer/Ultra-Light-Fast-Generic-Face-Detector-1MB. Article (CrossRefLink)

[24] https://github.com/dlunion/DBFace. Article (CrossRefLink)

[25] Deng, J., Guo, J., Zhou, Y., Yu, J., Kotsia, I., and Zafeiriou, S., “RetinaFace: Single-stage Dense Face Localisation in the Wild,” in *Proc. of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. Article (CrossRefLink)

[26] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. of 2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779-788, June 26-30, 2016. Article (CrossRefLink)

[27] Liu, Wei, et al. "SSD: Single Shot MultiBox Detector," in *Proc. of European Conference on Computer Vision*, vol. 9905, pp. 21–37, October 8-16, 2016. Article (CrossRefLink)

[28] Shang, W., Sohn, K., Almeida, D., and Lee, H., "Understanding and improving convolutional neural networks via concatenated rectified linear units," in *Proc. of the 33rd International Conference on International Conference on Machine Learning*, vol. 48, pp. 2217-2225, June 19-24, 2016. Article (CrossRefLink)

[29] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in *Proc. of 2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818-2826, June 27-30, 2016. Article (CrossRefLink)

[30] J. Hu, L. Shen and G. Sun, "Squeeze-and-Excitation Networks," in *Proc. of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7132-7141, June 18-23, 2018. Article (CrossRefLink)

[31] Vidit Jain and Erik Learned-Miller, "FDDB: A Benchmark for Face Detection in Unconstrained Settings," *Technical Report UM-CS-2010-009, Dept. of Computer Science, University of Massachusetts, Amherst*, 2010. Article (CrossRefLink)

[32] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Proc. of 2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2879-2886, June 16-21, 2012. Article (CrossRefLink)

[33] Yang, Shuo, et al., "WIDER FACE: A Face Detection Benchmark," in *Proc. of 2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5525-5533, June 27-30, 2016. Article (CrossRefLink)

[34] Mathias, M., Benenson, R., Pedersoli, M., & Gool, L. J. V., "Face Detection without Bells and Whistles," in *Proc. of European Conference on Computer Vision*," vol 8692, 720-735, September 5-12, 2014. Article (CrossRefLink)

[35] R. Benenson, M. Mathias, T. Tuytelaars and L. Van Gool, "Seeking the Strongest Rigid Detector," in *Proc. of 2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3666-3673, June 23-28, 2013. Article (CrossRefLink)

[36] X. Shen, Z. Lin, J. Brandt and Y. Wu, "Detecting and Aligning Faces by Image Retrieval," in *Proc. of 2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3460-3467, June 23-28, 2013. Article (CrossRefLink)

[37] E. Ohn-Bar and M. M. Trivedi, "To boost or not to boost? On the limits of boosted trees for object detection," in *Proc. of 2016 23rd International Conference on Pattern Recognition*, pp. 3350-3355, October 4-8, 2016. Article (CrossRefLink)

[38] S. Yang, P. Luo, C. Loy and X. Tang, "From Facial Parts Responses to Face Detection: A Deep Learning Approach," in *Proc. of 2015 IEEE International Conference on Computer Vision* , pp. 3676-3684, December 7-13, 2015. Article (CrossRefLink)

[39] Li, Y., Sun, B., Wu, T., and Wang, Y., "Face Detection with End-to-End Integration of a ConvNet and a 3D Model," in *Proc. of European Conference on Computer Vision*, vol. 9907, pp. 420–436, October 8-16, 2016. Article (CrossRefLink)

[40] Triantafyllidou, D., and Tefas, A., "A Fast Deep Convolutional Neural Network for Face Detection in Big Visual Data," in *Proc. of INNS Conference on Big Data*, vol. 529, pp. 61–70, 2016. Article (CrossRefLink)

[41] Chen, D., Ren, S., Wei, Y., Cao, X., and Sun, J., "Joint Cascade Face Detection and Alignment," in *Proc. of European Conference on Computer Vision*, vol. 8694, pp. 109-122, 2014. Article (CrossRefLink)

[42] Farfade, S. S., Saberian, M. J., and Li, L.-J., “Multi-view Face Detection Using Deep Convolutional Neural Networks,” in *Proc. of the 5th ACM on International Conference on Multimedia Retrieval*, pp. 643-650, June 9-12, 2015. Article (CrossRefLink)

[43] H. Li, G. Hua, Z. Lin, J. Brandt and J. Yang, “Probabilistic Elastic Part Model for Unsupervised Face Detector Adaptation,” in *Proc. of 2013 IEEE International Conference on Computer Vision*, pp. 793-800, December 1-8, 2013. Article (CrossRefLink)

[44] Kalal, Z., Matas, J., and Mikolajczyk, K., “Weighted Sampling for Large-Scale Boosting,” in *Proc. of British Machine Vision Conference*, pp. 42.1-42.1, September 1-4, 2008. Article (CrossRefLink)

Shuaihui Qi received his BE degree from the School of Automation, Chongqing University, Chongqing, China, in 2016, his ME degree from the College of Intelligence Science and Technology, National University of Defense Technology, Changsha, China, in 2018. He is currently an assistant with the School of Information and Communication, National University of Defense Technology, Xi’an, China. His current research interests include object detection, object tracking, and semantic segmentation.

Jungang Yang received his BE degree from Institute of Communication Engineering, Nanjing, China, in 1996, his ME degree and PhD from the Institute of Communication Engineering, Xidian University, Xi’an, China, in 2003 and in 2008, respectively. He is currently a professor with the School of Information and Communication, National University of Defense Technology, Xi’an, China. His current research interests include artificial intelligence security, intelligent combat, and research of unmanned platform.

Xiaofeng Song received his BS degree from the School of Information and Technology, Zhengzhou University, Zhengzhou, China, in 2002, his MS degree from the School of Computer Science, Xidian University, Xi’an, China, in 2009, and his PhD from Zhengzhou Science and Technology Institute, Zhengzhou, China, in 2016. He is currently an associate professor with the School of Information and Communication, National University of Defense Technology, Xi’an, China. His current research interests include digital image forensics, image steganography and steganalysis.

Chen Jiang received the BE degree from School of Computer Science, Xi’an Polytechnic University, Zhengzhou, China, in 2008. Currently, he is a lecturer of the School of Information and Communication, National University of Defense Technology, Xi’an, China. His research interest includes image processing, and graphic design.