
Robust and High Performance Face Detector

Yundong Zhang, Xiang Xu, Xiaotao Liu

Vimicro AI Chip Technology Corporation, Beijing, China

State Key Laboratory of Digital Multi-media Chip Technolgy, Beijing, China

{raymond, xuxiang, liuxiaotao}@vimicro.com

Abstract

In recent years, face detection has experienced significant performance improvement with the boost of deep convolutional neural networks. In this report, we reimplement the state-of-the-art detector [1] and apply some tricks proposed in the recent literatures to obtain an extremely strong face detector, named VIM-FD. In specific, we exploit more powerful backbone network like DenseNet-121 [2], revisit the data augmentation based on data-anchor-sampling proposed in [3], and use the max-in-out label and anchor matching strategy in [4]. In addition, we also introduce the attention mechanism [5, 6] to provide additional supervision. Over the most popular and challenging face detection benchmark, *i.e.*, WIDER FACE [7], the proposed VIM-FD achieves state-of-the-art performance.

1 Introduction

Because face detection serves as a specific task for generic object detection, the development of generic object detection significantly promotes the development of face detection. Deep convolutional neural network (CNN) based object detectors have become more and more developed and achieved great success in recent years, owing to the significant progress of network architecture such as VGG [8], Inception [9, 10], ResNet [11] and DenseNet [2]. Advanced object detection frameworks can be divided into two categories: one-stage detector and two-stage detector. Most state-of-the-art methods use two-stage detectors, *e.g.*, Faster R-CNN [12], R-FCN [13], FPN [14] and Cascade R-CNN [15]. These approaches first obtain a manageable number of region proposals called region of interest (RoI) and then pool out the corresponding features. In the second stage, R-CNN classifies and regresses each RoI again. By contrast, one-stage detectors have the advantage of simple structures and high speed. SSD [16] and YOLO [17, 18] have achieved good speed/accuracy trade-off. However, they can hardly surpass the accuracy of two-stage detectors. RetinaNet [19] is the state-of-the-art one-stage detector that achieves comparable performance to two-stage detectors. It adopts an architecture modified from RPN [20] and focuses on addressing the class imbalance during training.

As a long-standing problem in computer vision, face detection has extensive applications including face alignment, face analysis, face recognition, etc. Starting from the pioneering work of Viola-Jones [21], face detection has also made great progress. The milestone work of Viola-Jones uses Haar feature and AdaBoost to train a cascade of face/non-face classifiers that achieves a good accuracy with real-time efficiency. After that, lots of works have focused on improving the performance with more sophisticated hand-crafted features [22] and more powerful classifiers [23]. Besides the cascade structure, [24] introduces deformable part models (DPM) into face detection tasks and achieves remarkable performance. However, these methods highly depend on the robustness of hand-crafted features and optimize each component separately, making face detection pipeline sub-optimal. Recent years have witnessed the advance of CNN-based face detectors. CascadeCNN [25] develops a cascade architecture built on CNNs with powerful discriminative capability and high performance. Qin et al. [26] propose to jointly train CascadeCNN to realize end-to-end optimization. Faceness [27] trains a series of CNNs for facial attribute recognition to detect partially occluded faces. MTCNN [28] proposes to jointly solve face detection and alignment using several multi-task CNNs. UnitBox [29]

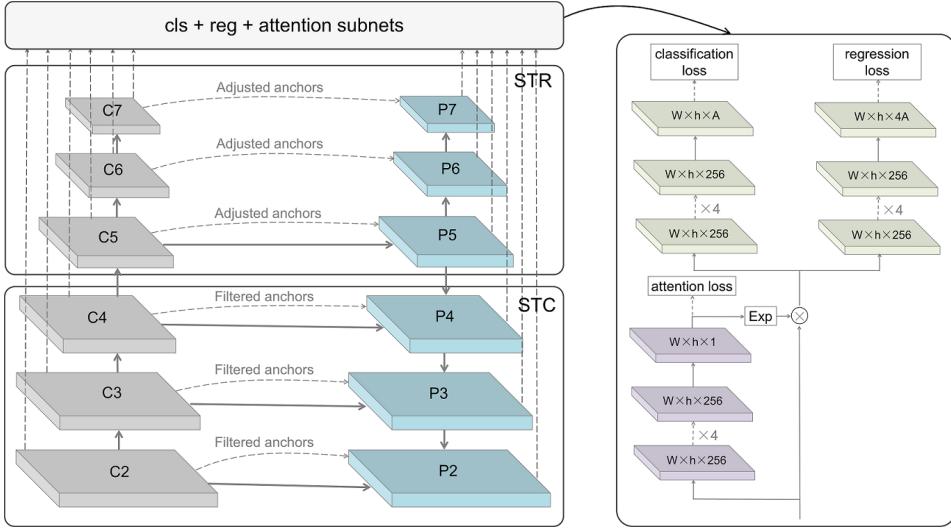


Figure 1: Network structure of VIM-FD. It consists of STC, STR, and prediction subnets. STC uses the first-step classifier to filter out most simple negative anchors from low level detection layers to reduce the search space for the second-step classifier. STR applies the first-step regressor to coarsely adjust the locations and sizes of anchors from high level detection layers to provide better initialization for the second-step regressor. Prediction subnets perform classification, regression and attention map prediction jointly.

introduces a new intersection-over-union loss function. The performances on several well-known datasets have been improved consistently, even tend to be saturated.

In this paper, we reimplement the state-of-the-art detector [1] and utilize it as our baseline model. Then we revisit several proposed tricks from the following aspects: (1) Data augmentation method; (2) Matching and classification strategy; (3) Impact of the backbone network; (4) Attention mechanism in face detection. Through the organic integration of these tricks, we obtain a very powerful face detector, which achieves state-of-the-art result on WIDER FACE dataset.

2 Related Work

Face detection has been a challenging research field since its emergence in the 1990s. Viola and Jones pioneer to use Haar features and AdaBoost to train a face detector with promising accuracy and efficiency [21], which inspires several different approaches afterwards [22, 23]. Apart from those, another important work is the introduction of Deformable Part Model (DPM) [24].

Recently, face detection has been dominated by the CNN-based methods. CascadeCNN [25] improves detection accuracy by training a series of interleaved CNN models and following work [26] proposes to jointly train the cascaded CNNs to realize end-to-end optimization. EMO [30] proposes an Expected Max Overlapping score to evaluate the quality of anchor matching. SAFD [31] develops a scale proposal stage which automatically normalizes face sizes prior to detection. S²AP [32] pays attention to specific scales in image pyramid and valid locations in each scales layer. PCN [33] proposes a cascade-style structure to rotate faces in a coarse-to-fine manner. Recent work [34] designs a novel network to directly generate a clear super-resolution face from a blurry small one.

Additionally, face detection has inherited some achievements from generic object detectors, such as Faster R-CNN [12], SSD [16], FPN [14], RefineDet [35] and RetinaNet [19]. Face R-CNN [36] combines Faster R-CNN with hard negative mining and achieves promising results. FaceBoxes [37] introduces a CPU real-time detector based on SSD. Face R-FCN [38] applies R-FCN in face detection and makes according improvements. The face detection model for finding tiny faces [39] trains separate detectors for different scales. S³FD [4] presents multiple strategies onto SSD to compensate for the matching problem of small faces. SSH [40] models the context information by large filters on each prediction module. PyramidBox [3] utilizes contextual information with

improved SSD network structure. FAN [5] proposes an anchor-level attention into RetinaNet to detect the occluded faces.

3 Proposed Approach

The overall framework of VIM-FD is shown in Figure 1, we describe each component as follows.

3.1 Backbone

The original detector [1] adopts ResNet-50 [11] as the backbone network, which is a little bit obsolete currently. We explore several more powerful backbone network, such as ResNeXt [41], DenseNet [2] and NASNet [42]. Finally we adopt DenseNet-121 with 6-level feature pyramid structure as the backbone network for VIM-FD. The feature maps extracted from those four blocks are denoted as C2, C3, C4, and C5, respectively. C6 and C7 are just extracted by two simple down-sample 3×3 convolution layers after C5. The lateral structure between the bottom-up and the top-down pathways is the same as [14]. P2, P3, P4, and P5 are the feature maps extracted from lateral connections, corresponding to C2, C3, C4, and C5 that are respectively of the same spatial sizes, while P6 and P7 are just down-sampled by two 3×3 convolution layers after P5.

3.2 STC and STR Module

The STC module aims to filter out most simple negative anchors from low level detection layers to reduce the search space for the subsequent classifier, which selects C2, C3, C4, P2, P3, and P4 to perform two-step classification. While the STR module is designed to coarsely adjust the locations and sizes of anchors from high level detection layers to provide better initialization for the subsequent regressor, which selects C5, C6, C7, P5, P6, and P7 to conduct two-step regression. The loss function of these two modules is also same with [1], which is described in detail as below:

$$\begin{aligned} \mathcal{L}_{\text{STC}}(\{p_i\}, \{q_i\}) = & \frac{1}{N_{s_1}} \sum_{i \in \Omega} \mathcal{L}_{\text{FL}}(p_i, l_i^*) \\ & + \frac{1}{N_{s_2}} \sum_{i \in \Phi} \mathcal{L}_{\text{FL}}(q_i, l_i^*), \end{aligned} \quad (1)$$

where i is the index of anchor in a mini-batch, p_i and q_i are the predicted confidence of the anchor i being a face in the first and second steps, l_i^* is the ground truth class label of anchor i , N_{s_1} and N_{s_2} are the numbers of positive anchors in the first and second steps, Ω represents a collection of samples selected for two-step classification, and Φ represents a sample set that remains after the first step filtering. The binary classification loss \mathcal{L}_{FL} is the sigmoid focal loss over two classes (face vs. background).

$$\begin{aligned} \mathcal{L}_{\text{STR}}(\{x_i\}, \{t_i\}) = & \sum_{i \in \Psi} [l_i^* = 1] \mathcal{L}_r(x_i, g_i^*) \\ & + \sum_{i \in \Phi} [l_i^* = 1] \mathcal{L}_r(t_i, g_i^*), \end{aligned} \quad (2)$$

where g_i^* is the ground truth location and size of anchor i , x_i is the refined coordinates of the anchor i in the first step, t_i is the coordinates of the bounding box in the second step, Ψ represents a collection of samples selected for two-step regression, l_i^* and Φ are the same as defined in STC. Similar to Faster R-CNN [12], we use the smooth L_1 loss as the regression loss \mathcal{L}_r . The Iverson bracket indicator function $[l_i^* = 1]$ outputs 1 when the condition is true, i.e., $l_i^* = 1$ (the anchor is not the negative), and 0 otherwise. Hence $[l_i^* = 1] \mathcal{L}_r$ indicates that the regression loss is ignored for negative anchors.

3.3 Attention module

Recently, attention mechanism is applied continually in object detection and face detection. DES [6] utilizes the idea of weakly supervised semantic segmentation, to provide high semantic meaningful and class-aware features to activate and calibrate feature map used in the object detection. FAN [5] introduces anchor-level attention to highlight the features from the facial regions and successfully relieving the risk from the false positives.

We apply the attention subnet in FAN to the P2, P3, P4, P5, P6 and P7 layers, the specific structure of which is shown in Figure 1. Specifically, the attention supervision information is obtained by filling the ground-truth box. And the supervised masks are associated to the ground-truth faces assigned to the anchors in the current detection layer. Because the first step and the second step share the same detection subnet, we also apply attention subnet on bottom-up levels, but we do not calculate attention loss on these layers. We define the attention loss function as:

$$\mathcal{L}_{\text{ATT}}(\{m_i\}) = \sum_{i \in X} \mathcal{L}_{\text{sig}}(m_i, m_i^*), \quad (3)$$

where m_i is the predicted attention map generated per level in the second step, m_i^* is the ground truth attention mask of the i th detection layer, X represents the set of detection layers applied to attention mechanism (*i.e.*, P2, P3, P4, P5, P6 and P7), and \mathcal{L}_{sig} is pixel-wise sigmoid cross entropy loss.

3.4 Max-in-out Label

$S^3\text{FD}$ [4] introduces max-out background label to reduce the false positives of small negatives. PyramidBox [3] uses this strategy on both positive and negative samples. In specific, this strategy first predicts $c_p + c_n$ scores for each prediction module, and then selects $\max c_p$ as the positive score. Similarly, it chooses the max score of c_n to be the negative score. In our VIM-FD, we employ the max-in-out label in the classification subnet, and set $c_p = 3$ and $c_n = 3$ to recall more faces and reduce false positives simultaneously.

3.5 Anchor Design and Matching

The design of anchor scale and ratio keeps same with [1]. At each pyramid level, we use two specific scales of anchors as same as [1] (*i.e.*, $2S$ and $2\sqrt{2}S$, where S represents the total stride size of each pyramid level) and one aspect ratios (*i.e.*, 1.25). In total, there are $A = 2$ anchors per level and they cover the scale range 8 – 362 pixels across levels with respect to the network’s input image.

During the training phase, anchors need to be divided into positive and negative samples. Specifically, anchors are assigned to ground-truth face boxes using an intersection-over-union (IoU) threshold of θ_p ; and to background if their IoU is in $[0, \theta_n]$. If an anchor is unassigned, which may happen with overlap in $[\theta_n, \theta_p]$, it is ignored during training. Empirically, we set $\theta_n = 0.3$ and $\theta_p = 0.7$ for the first step, and $\theta_n = \theta_p = 0.35$ for the second step. This setting draws on the scale compensation anchor matching strategy in $S^3\text{FD}$ [4], aiming to improve the recall rate of small faces. The setting is based on the observation that faces whose scale distribute away from anchor scales can not match enough anchors. To solve this issue, we decrease the IoU threshold to increase the average number of matched anchors. The scale compensation anchor matching strategy greatly increases the matched anchors of tiny and outer faces, which notably improves the recall rate of these faces.

3.6 Data Augmentation

We employ the data-anchor-sampling method in PyramidBox [3] to diversify the scale distribution of training samples and construct a robust model. Specifically, we first randomly select a face of size S_{face} in a batch. Let

$$i_{\text{anchor}} = \arg \min_i \text{abs}(S_{\text{anchor}_i} - S_{\text{face}}) \quad (4)$$

be the index of the nearest anchor scale from the selected face, then we choose a random index i_{target} in the set $\{0, 1, \dots, \min(5, i_{\text{anchor}} + 1)\}$, thus we get the image resize scale

$$S^* = \text{random}(S_{i_{\text{target}}} / 2, S_{i_{\text{target}}} * 2) / S_{\text{face}}. \quad (5)$$

By resizing the original image with the scale S^* and cropping a standard size of 640×640 containing the selected face randomly, we get the anchor-sampled training data.

3.7 Loss Function

We append a hybrid loss at the end of the deep architecture to jointly optimize model parameters, which is just the sum of the STC loss, the STR loss and the ATT loss:

$$\mathcal{L} = \mathcal{L}_{\text{STC}} + \mathcal{L}_{\text{STR}} + \mathcal{L}_{\text{ATT}}, \quad (6)$$

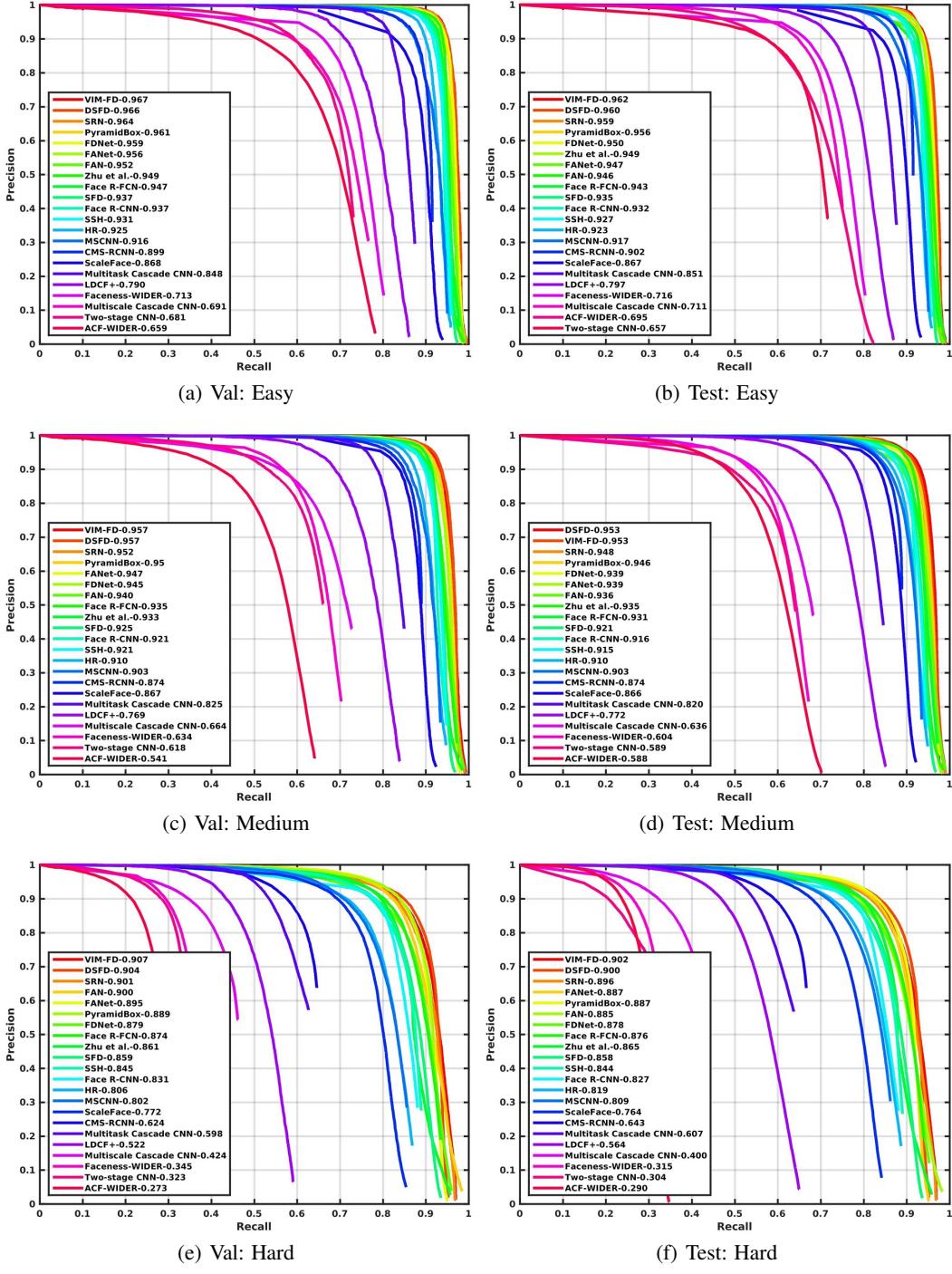


Figure 2: Precision-recall curves on WIDER FACE validation and testing subsets.

4 Experiments

The backbone network is initialized by the pretrained DenseNet-121 model [43] and all the parameters in the newly added convolution layers are initialized by the “xavier” method. We fine-tune the model using SGD with 0.9 momentum, 0.0001 weight decay, and batch size 32. We set the learning rate to 10^{-2} for the first 100 epochs, and decay it to 10^{-3} and 10^{-4} for another 20 and 10 epochs, respectively. We implement VIM-FD using the PyTorch library [44].

4.1 Dataset

The WIDER FACE dataset [7] consists of 393, 703 annotated face bounding boxes in 32, 203 images with variations in pose, scale, facial expression, occlusion, and lighting condition. The dataset is split into the training (40%), validation (10%) and testing (50%) sets, and defines three levels of difficulty: Easy, Medium, Hard, based on the detection rate of EdgeBox [45]. Due to the variability of scale, pose and occlusion, WIDER FACE dataset is one of the most challenge face datasets. Our VIM-FD is trained only on the training set and evaluated on both validation set and testing set.

4.2 Experimental Results

We compare VIM-FD with twenty-one state-of-the-art face detection methods on both the validation and testing sets. To obtain the evaluation results on the testing set, we submit the detection results of VIM-FD to the authors for evaluation. As shown in Figure 2, we find that VIM-FD performs favourably against the state-of-the-art based on the average precision (AP) across the three subsets, especially on the Hard subset which contains a large amount of small faces. Specifically, it produces the best AP scores in all subsets of both validation and testing sets, *i.e.*, 96.7% (Easy), 95.7% (Medium) and 90.7% (Hard) for validation set, and 96.2% (Easy), 95.3% (Medium) and 90.2% (Hard) for testing set. Except that the AP of Medium subset is equal to DSFD [46], our results surpass all approaches, which demonstrates the superiority of the proposed detector. We first show an impressive qualitative result of the World Largest Selfie¹ in Figure 3 and VIM-FD successfully finds 890 faces out of the reported 1000 faces. We also demonstrate some qualitative results on WIDER FACE in Figure 4, indicating the proposed VIM-FD is robust to scale, blur, expression, illumination, makeup, occlusion and pose.

5 Conclusion

In this report, we reimplement the state-of-the-art detector [1] and revisit several tricks proposed in the recent literatures to obtain an extremely strong face detector, named VIM-FD. In specific, we make some explorations in the following aspects: (1) Data augmentation method; (2) Matching and classification strategy; (3) Impact of the backbone network; (4) Attention mechanism in face detection. Extensive experiments on the WIDER FACE dataset demonstrate that VIM-FD achieves the state-of-the-art detection performance.

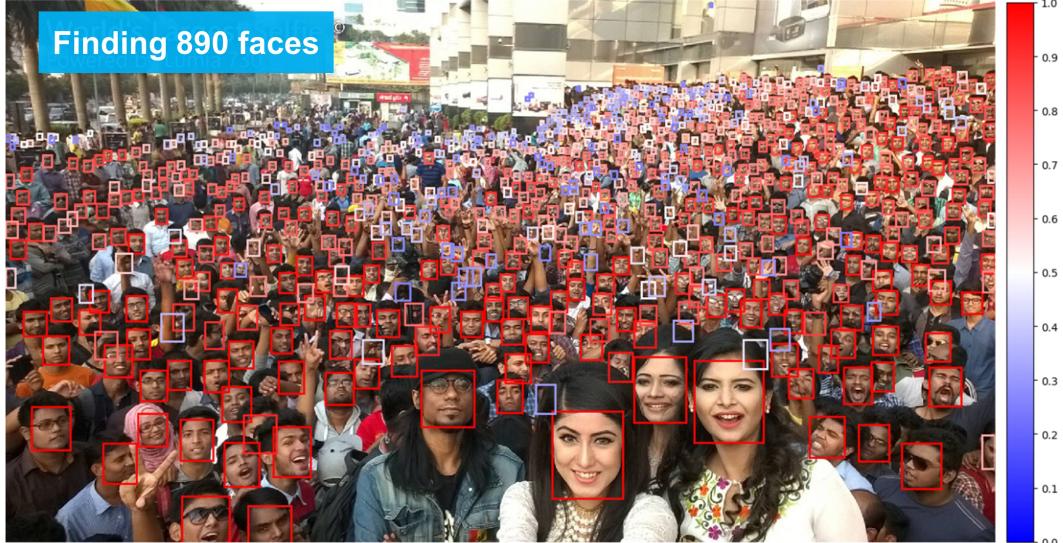


Figure 3: Impressive qualitative result. VIM-FD finds 890 faces out of the reported 1000 faces. The confidences of the detections are presented in the color bar on the right hand. Best viewed in color.

¹<https://www.cs.cmu.edu/~peiyunh/tiny/>



(a) Scale attribute. Our VIM-FD is able to detect faces at a continuous range of scales.



(b) Our VIM-FD is robust to blur, expression, illumination, makeup, occlusion and pose.

Figure 4: Qualitative results on WIDER FACE. We visualize some examples for each attribute. Please zoom in to see small detections.

References

- [1] Cheng Chi, Shifeng Zhang, Junliang Xing, Zhen Lei, Stan Z. Li, and Xudong Zou. Selective refinement network for high performance face detection. In *AAAI*, 2019.
- [2] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *CVPR*, 2017.
- [3] Xu Tang, Daniel K Du, Zeqiang He, and Jingtuo Liu. Pyramidbox: A context-assisted single shot face detector. In *ECCV*, 2018.
- [4] Shifeng Zhang, Xiangyu Zhu, Zhen Lei, Hailin Shi, Xiaobo Wang, and Stan Z. Li. S³FD: Single shot scale-invariant face detector. In *ICCV*, 2017.
- [5] Jianfeng Wang, Ye Yuan, and Gang Yu. Face attention network: An effective face detector for the occluded faces. *CoRR*, 2017.
- [6] Zhishuai Zhang, Siyuan Qiao, Cihang Xie, Wei Shen, Bo Wang, and Alan L. Yuille. Single-shot object detection with enriched semantics. In *CVPR*, 2018.
- [7] Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. WIDER FACE: A face detection benchmark. In *CVPR*, 2016.
- [8] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, 2014.
- [9] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.
- [10] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016.
- [11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- [12] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *TPAMI*, 2017.
- [13] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-FCN: object detection via region-based fully convolutional networks. In *NIPS*, 2016.
- [14] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.
- [15] Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: delving into high quality object detection. In *CVPR*, 2018.
- [16] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. In *ECCV*, 2016.
- [17] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016.
- [18] Joseph Redmon and Ali Farhadi. YOLO9000: better, faster, stronger. *CoRR*, 2016.
- [19] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017.
- [20] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *NIPS*, 2015.
- [21] Paul A. Viola and Michael J. Jones. Robust real-time face detection. *IJCV*, 2004.
- [22] Shengcui Liao, Anil K. Jain, and Stan Z. Li. A fast and accurate unconstrained face detector. *TPAMI*, 2016.
- [23] S. Charles Brubaker, Jianxin Wu, Jie Sun, Matthew D. Mullin, and James M. Rehg. On the design of cascades of boosted ensembles for face detection. *IJCV*, 2008.
- [24] Markus Mathias, Rodrigo Benenson, Marco Pedersoli, and Luc J. Van Gool. Face detection without bells and whistles. In *ECCV*, 2014.
- [25] Haoxiang Li, Zhe Lin, Xiaohui Shen, Jonathan Brandt, and Gang Hua. A convolutional neural network cascade for face detection. In *CVPR*, 2015.

- [26] Hongwei Qin, Junjie Yan, Xiu Li, and Xiaolin Hu. Joint training of cascaded CNN for face detection. In *CVPR*, 2016.
- [27] Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. From facial parts responses to face detection: A deep learning approach. In *ICCV*, 2015.
- [28] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *SPL*, 2016.
- [29] Jiahui Yu, Yuning Jiang, Zhangyang Wang, Zhimin Cao, and Thomas S. Huang. Unitbox: An advanced object detection network. In *ACMMM*, 2016.
- [30] Chenchen Zhu, Ran Tao, Khoa Luu, and Marios Savvides. Seeing small faces from robust anchors perspective. In *CVPR*, 2018.
- [31] Zekun Hao, Yu Liu, Hongwei Qin, Junjie Yan, Xiu Li, and Xiaolin Hu. Scale-aware face detection. In *CVPR*, 2017.
- [32] Guanglu Song, Yu Liu, Ming Jiang, Yujie Wang, Junjie Yan, and Biao Leng. Beyond trade-off: Accelerate fcn-based face detector with higher accuracy. In *CVPR*, 2018.
- [33] Xuepeng Shi, Shiguang Shan, Meina Kan, Shuzhe Wu, and Xilin Chen. Real-time rotation-invariant face detection with progressive calibration networks. In *CVPR*, 2018.
- [34] Yancheng Bai, Yongqiang Zhang, Mingli Ding, and Bernard Ghanem. Finding tiny faces in the wild with generative adversarial network. In *CVPR*, 2018.
- [35] Shifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, and Stan Z. Li. Single-shot refinement neural network for object detection. In *CVPR*, 2018.
- [36] Hao Wang, Zhifeng Li, Xing Ji, and Yitong Wang. Face r-cnn. *CoRR*, 2017.
- [37] Shifeng Zhang, Xiangyu Zhu, Zhen Lei, Hailin Shi, Xiaobo Wang, and Stan Z. Li. Faceboxes: A CPU real-time face detector with high accuracy. In *IJCB*, 2017.
- [38] Yitong Wang, Xing Ji, Zheng Zhou, Hao Wang, and Zhifeng Li. Detecting faces using region-based fully convolutional networks. *CoRR*, 2017.
- [39] Peiyun Hu and Deva Ramanan. Finding tiny faces. In *CVPR*, 2017.
- [40] Mahyar Najibi, Pouya Samangouei, Rama Chellappa, and Larry S. Davis. SSH: single stage headless face detector. In *ICCV*, 2017.
- [41] Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017.
- [42] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V. Le. Learning transferable architectures for scalable image recognition. In *CVPR*, 2018.
- [43] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *IJCV*, 2015.
- [44] Adam Paszke, Sam Gross, Soumith Chintala, and Gregory Chanan. Pytorch, 2017.
- [45] C. Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014.
- [46] Jian Li, Yabiao Wang, Changan Wang, Ying Tai, Jianjun Qian, Jian Yang, Chengjie Wang, Ji-Lin Li, and Feiyue Huang. DSFD: dual shot face detector. *CoRR*, 2018.