DSFD: Dual Shot Face Detector

Jian Li[†] Yabiao Wang[‡] Changan Wang[‡] Ying Tai[‡]
Jianjun Qian[†*] Jian Yang[†*] Chengjie Wang[‡] Jilin Li[‡] Feiyue Huang[‡]
[†]PCA Lab, Key Lab of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education
[†]Jiangsu Key Lab of Image and Video Understanding for Social Security
[†]School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China
[‡]Youtu Lab, Tencent
[†]lijiannuist@gmail.com, {csjqian, csjyang}@njust.edu.cn
[‡]{casewang, changanwang, yingtai, jasoncjwang, jerolinli, garyhuang}@tencent.com

Figure 1: **Visual results**. Our method is robust to various variations on scale, blurry, illumination, pose, occlusion, reflection and makeup.

Abstract

In this paper, we propose a novel face detection network with three novel contributions that address three key aspects of face detection, including better feature learning, progressive loss design and anchor assign based data augmentation, respectively. First, we propose a Feature Enhance Module (FEM) for enhancing the original feature maps to extend the single shot detector to dual shot detector. Second, we adopt Progressive Anchor Loss (PAL) computed by two different sets of anchors to effectively facilitate the features. Third, we use an Improved Anchor Matching (IAM) by integrating novel anchor assign strategy into data augmentation to provide better initialization for the regressor. Since these techniques are all related to the two-stream design, we name the proposed network as Dual Shot Face Detector (DSFD). Extensive experiments on popular benchmarks, WIDER FACE and FDDB, demonstrate the superiority of DSFD over the state-of-the-art face detectors.

1. Introduction

Face detection is a fundamental step for various facial applications, like face alignment [26], parsing [3], recognition [34], and verification [6]. As the pioneering work for face detection, Viola-Jones [29] adopts AdaBoost algorithm with hand-crafted features, which are now replaced by deeply learned features from the convolutional neural network (CNN) [10] that achieves great progress. Although

*Jianjun Qian and Jian Yang are corresponding authors. This work was supported by the National Science Fund of China under Grant Nos. 61876083, U1713208, and Program for Changjiang Scholars. This work was done when Jian Li was an intern at Tencent Youtu Lab.

978-1-7281-3293-8/19/$31.00 ©2019 IEEE
DOI 10.1109/CVPR.2019.00520

the CNN based face detectors have being extensively studied, detecting faces with high degree of variability in scale, pose, occlusion, expression, appearance and illumination in real-world scenarios remains a challenge.

Previous state-of-the-art face detectors can be roughly divided into two categories. The first one is mainly based on the Region Proposal Network (RPN) adopted in Faster RCNN [24] and employs two stage detection schemes [30, 33, 36]. RPN is trained end-to-end and generates high-quality region proposals which are further refined by Fast R-CNN detector. The other one is Single Shot Detector (SSD) [20] based one-stage methods, which get rid of RPN, and directly predict the bounding boxes and confidence [4, 27, 39]. Recently, one-stage face detection framework has attracted more attention due to its higher inference efficiency and straightforward system deployment.

Despite the progress achieved by the above methods, there are still some problems existed in three aspects:

Feature learning Feature extraction part is essential for a face detector. Currently, Feature Pyramid Network (FPN) [17] is widely used in state-of-the-art face detectors for rich features. However, FPN just aggregates hierarchical feature maps between high and low-level output layers, which does not consider the current layer's information, and the context relationship between anchors is ignored.

Loss design The conventional loss functions used in object detection include a regression loss for the face region and a classification loss for identifying if a face is detected or not. To further address the class imbalance problem, Lin *et al.* [18] propose Focal Loss to focus training on a sparse set of hard examples. To use all original and enhanced features, Zhang *et al.* propose Hierarchical Loss to effectively learn the network [37]. However, the above loss functions do not consider progressive learning ability of feature maps in both of different levels and shots.

Anchor matching Basically, pre-set anchors for each feature map are generated by regularly tiling a collection of boxes with different scales and aspect ratios on the image. Some works [27, 39] analyze a series of reasonable anchor scales and anchor compensation strategy to increase positive anchors. However, such strategy ignores random sampling in data augmentation, which still causes imbalance between positive and negative anchors.

In this paper, we propose three novel techniques to address the above three issues, respectively. First, we introduce a Feature Enhance Module (FEM) to enhance the discriminability and robustness of the features, which combines the advantages of the FPN in PyramidBox and Receptive Field Block (RFB) in RFBNet [19]. Second, motivated by the hierarchical loss [37] and pyramid anchor [27] in PyramidBox, we design Progressive Anchor Loss (PAL) that uses progressive anchor sizes for not only different levels, but also different shots. Specifically, we assign smaller anchor sizes in the first shot, and use larger sizes in the second shot. Third, we propose Improved Anchor Matching (IAM), which integrates anchor partition strategy and anchor-based data augmentation to better match anchors and ground truth faces, and thus provides better initialization for the regressor. The three aspects are *complementary* so that these techniques can work together to further improve the performance. Besides, since these techniques are all related to two-stream design, we name the proposed network as Dual Shot Face Detector (DSFD). Fig. 1 shows the effectiveness of DSFD on various variations, especially on extreme small faces or heavily occluded faces.

In summary, the main contributions of this paper include:

• A novel Feature Enhance Module to utilize different level information and thus obtain more discriminability and robustness features.

• Auxiliary supervisions introduced in early layers via a set of smaller anchors to effectively facilitate the features.

• An improved anchor matching strategy to match anchors and ground truth faces as far as possible to provide better initialization for the regressor.

• Comprehensive experiments conducted on popular benchmarks FDDB and WIDER FACE to demonstrate the superiority of our proposed DSFD network compared with the state-of-the-art methods.

2. Related work

We review the prior works from three perspectives.

Feature Learning Early works on face detection mainly rely on hand-crafted features, such as Harr-like features [29], control point set [1], edge orientation histograms [13]. However, hand-crafted features design is lack of guidance. With the great progress of deep learning, hand-crafted features have been replaced by Convolutional Neural Networks (CNN). For example, Overfeat [25], Cascade-CNN [14], MTCNN [38] adopt CNN as a sliding window detector on image pyramid to build feature pyramid. However, using an image pyramid is slow and memory inefficient. As the result, most two stage detectors extract features on single scale. R-CNN [7, 8] obtains region proposals by selective search [28], and then forwards each normalized image region through a CNN to classify. Faster R-CNN [24], R-FCN [5] employ Region Proposal Network (RPN) to generate initial region proposals. Besides, ROI-pooling [24] and position-sensitive RoI pooling [5] are applied to extract features from each region.

More recently, some research indicates that multi-scale features perform better for tiny objects. Specifically, SSD [20], MS-CNN [2], SSH [23], S3FD [39] predict boxes on multiple layers of feature hierarchy. FCN [22], Hypercolumns [9], Parsenet [21] fuse multiple layer features in segmentation. FPN [15, 17], a top-down architecture, integrate high-level semantic information to all scales.

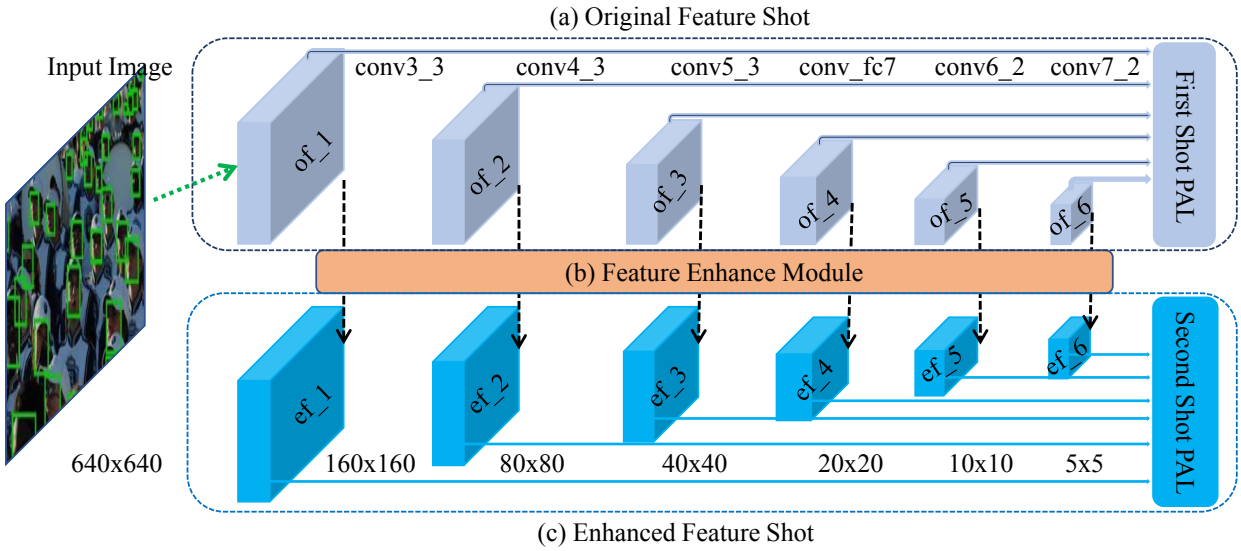

Figure 2: **Our DSFD framework** uses a Feature Enhance Module (b) on top of a feedforward VGG/ResNet architecture to generate the enhanced features (c) from the original features (a), along with two loss layers named first shot PAL for the original features and second shot PAL for the enchanted features.

FPN-based methods, such as FAN [31], PyramidBox [27] achieve significant improvement on detection. However, these methods do not consider the current layers information. Different from the above methods that ignore the context relationship between anchors, we propose a feature enhance module that incorporates multi-level dilated convolutional layers to enhance the semantic of the features.

Loss Design Generally, the objective loss in detection is a weighted sum of classification loss (*e.g.* softmax loss) and box regression loss (*e.g.* L_2 loss). Girshick *et al.* [7] propose smooth L_1 loss to prevent exploding gradients. Lin *et al.* [18] discover that the class imbalance is one obstacle for better performance in one stage detector, hence they propose focal loss, a dynamically scaled cross entropy loss. Besides, Wang *et al.* [32] design RepLoss for pedestrian detection, which improves performance in occlusion scenarios. FANet [37] create a hierarchical feature pyramid and presents hierarchical loss for their architecture. However, the anchors used in FANet are kept the same size in different stages. In this work, we adaptively choose different anchor sizes in different stages to facilitate the features.

Anchor Matching To make the model more robust, most detection methods [20,35,39] do data augmentation, such as color distortion, horizontal flipping, random crop and multi-scale training. Zhang *et al.* [39] propose an anchor compensation strategy to make tiny faces to match enough anchors during training. Wang *et al.* [35] propose random crop to generate large number of occluded faces for training. However, these methods ignore random sampling in data augmentation, while ours combines anchor assign to provide better data initialization for anchor matching.

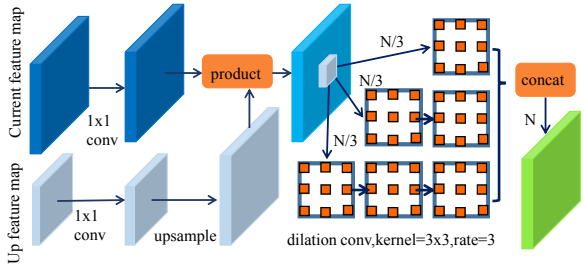

Figure 3: **Illustration on Feature Enhance Module**, in which the current feature map cell interactives with neighbors in current feature maps and up feature maps.

3. Dual Shot Face Detector

We firstly introduce the pipeline of our proposed framework DSFD, and then detailly describe our feature enhance module in Sec. 3.2, progressive anchor loss in Sec. 3.3 and improved anchor matching in Sec. 3.4, respectively.

3.1. Pipeline of DSFD

The framework of DSFD is illustrated in Fig. 2. Our architecture uses the same extended VGG16 backbone as PyramidBox [27] and S3FD [39], which is truncated before the classification layers and added with some auxiliary structures. We select conv3_3, conv4_3, conv5_3, conv_fc7, conv6_2 and conv7_2 as the first shot detection layers to generate six original feature maps named $of_1, of_2, of_3, of_4, of_5, of_6$. Then, our proposed FEM transfers these original feature maps into six enhanced feature maps named $ef_1, ef_2, ef_3, ef_4, ef_5, ef_6$, which have the same sizes as the original ones and are fed into SSD-style head to construct the second shot detection layers. Note that

the input size of the training image is 640, which means the feature map size of the lowest-level layer to highest-level layer is from 160 to 5. Different from S3FD and PyramidBox, after we utilize the receptive field enlargement in FEM and the new anchor design strategy, its unnecessary for the three sizes of stride, anchor and receptive field to satisfy equal-proportion interval principle. Therefore, our DSFD is more flexible and robustness. Besides, the original and enhanced shots have two different losses, respectively named First Shot progressive anchor Loss (FSL) and Second Shot progressive anchor Loss (SSL).

3.2. Feature Enhance Module

Feature Enhance Module is able to enhance original features to make them more discriminable and robust, which is called FEM for short. For enhancing original neuron cell $oc_{(i,j,l)}$, FEM utilizes different dimension information including upper layer original neuron cell $oc_{(i,j,l)}$ and current layer non-local neuron cells: $nc_{(i-\varepsilon,j-\varepsilon,l)}$, $nc_{(i-\varepsilon,j,l)}$, ..., $nc_{(i,j+\varepsilon,l)}$, $nc_{(i+\varepsilon,j+\varepsilon,l)}$. Specially, the enhanced neuron cell $ec_{(i,j,l)}$ can be mathematically defined as follow:

$$\begin{aligned} ec_{(i,j,l)} &= f_{concat}(f_{dilation}(nc_{(i,j,l)})) \\ nc_{i,j,l} &= f_{prod}(oc_{(i,j,l)}, f_{up}(oc_{(i,j,l+1)})) \end{aligned} \tag{1}$$

where $c_{i,j,l}$ is a cell located in (i, j) coordinate of the feature maps in the l-th layer, f denotes a set of basic dilation convolution, elem-wise production, up-sampling or concatenation operations. Fig. 3 illustrates the idea of FEM, which is inspired by FPN [17] and RFB [19]. Here, we first use 1×1 convolutional kernel to normalize the feature maps. Then, we up-sample upper feature maps to do element-wise product with the current ones. Finally, we split the feature maps to three parts, followed by three sub-networks containing different numbers of dilation convolutional layers.

3.3. Progressive Anchor Loss

Different from the traditional detection loss, we design *progressive* anchor sizes for not only different levels, but also different shots in our framework. Motivated by the statement in [24] that low-level features are more suitable for small faces, we assign smaller anchor sizes in the first shot, and use larger sizes in the second shot. First, our Second Shot anchor-based multi-task Loss function is defined as:

$$\begin{aligned} \mathcal{L}_{SSL}(p_i, p_i^*, t_i, g_i, a_i) =& \frac{1}{N_{conf}}(\Sigma_i L_{conf}(p_i, p_i^*) \\ &+ \frac{\beta}{N_{loc}}\Sigma_i p_i^* L_{loc}(t_i, g_i, a_i)), \end{aligned} \tag{2}$$

where N_{conf} and N_{loc} indicate the number of positive and negative anchors, and the number of positive anchors respectively, L_{conf} is the softmax loss over two classes (face vs. background), and L_{loc} is the smooth L_1 loss between the parameterizations of the predicted box t_i and ground-truth box g_i using the anchor a_i. When $p_i^* = 1$ ($p_i^* = \{0, 1\}$), the anchor a_i is positive and the localization loss is activated. β is a weight to balance the effects of the two terms. Compared to the enhanced feature maps in the same level, the original feature maps have less semantic information for classification but more high resolution location information for detection. Therefore, we believe that the original feature maps can detect and classify smaller faces. As the result, we propose the First Shot multi-task Loss with a set of smaller anchors as follows:

$$\begin{aligned} \mathcal{L}_{FSL}(p_i, p_i^*, t_i, g_i, sa_i) =& \frac{1}{N_{conf}}\Sigma_i L_{conf}(p_i, p_i^*) \\ &+ \frac{\beta}{N_{loc}}\Sigma_i p_i^* L_{loc}(t_i, g_i, sa_i), \end{aligned} \tag{3}$$

where sa indicates the smaller anchors in the first shot layers, and the two shots losses can be weighted summed into a whole Progressive Anchor Loss as follows:

$$\mathcal{L}_{PAL} = \mathcal{L}_{FSL}(sa) + \lambda\mathcal{L}_{SSL}(a). \tag{4}$$

Note that anchor size in the first shot is half of ones in the second shot, and λ is weight factor. Detailed assignment on the anchor size is described in Sec. 3.4. In prediction process, we only use the output of the second shot, which means no additional computational cost is introduced.

3.4. Improved Anchor Matching

Current anchor matching method is bidirectional between the anchor and ground-truth face. Therefore, anchor design and face sampling during augmentation are collaborative to match the anchors and faces as far as possible for better initialization of the regressor. Our IAM targets on addressing the contradiction between the discrete anchor scales and continuous face scales, in which the faces are augmented by $S_{input} * S_{face}/S_{anchor}$ (S indicates the spatial size) with the probability of 40% so as to *increase* the positive anchors, *stabilize* the training and thus improve the results. Table 1 shows details of our anchor design on how each feature map cell is associated to the fixed shape anchor. We set anchor ratio 1.5:1 based on face scale statistics. Anchor size for the original feature is one half of the enhanced feature. Additionally, with

Table 1: The stride size, feature map size, anchor scale, ratio, and number of six original/enhanced features for two shots.

Feature	Stride	Size	Scale	Ratio	Number
ef_1 (of_1)	4	160×160	16 (8)	1.5 : 1	25600
ef_2 (of_2)	8	80×80	32 (16)	1.5 : 1	6400
ef_3 (of_3)	16	40×40	64 (32)	1.5 : 1	1600
ef_4 (of_4)	32	20×20	128 (64)	1.5 : 1	400
ef_5 (of_5)	64	10×10	256 (128)	1.5 : 1	100
ef_6 (of_6)	128	5×5	512 (256)	1.5 : 1	25

Table 2: Effectiveness of Feature Enhance Module on the AP performance.

Component	Easy	Medium	Hard
FSSD+VGG16	92.6%	90.2%	79.1%
FSSD+VGG16+FEM	**93.0**%	**91.4**%	**84.6**%

Table 3: Effectiveness of Progressive Anchor Loss on the AP performance.

Component	Easy	Medium	Hard
FSSD+RES50	93.7%	92.2%	81.8%
FSSD+RES50+FEM	95.0%	94.1%	88.0%
FSSD+RES50+FEM+PAL	**95.3**%	**94.4**%	**88.6**%

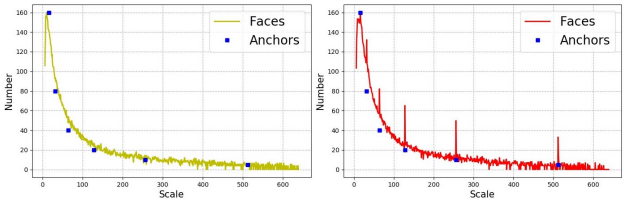

Figure 4: **The number distribution of different scales of faces** compared between traditional anchor matching (**Left**) and our improved anchor matching (**Right**).

probability of $2/5$, we utilize anchor-based sampling like data-anchor-sampling in PyramidBox, which randomly selects a face in an image, crops sub-image containing the face, and sets the size ratio between sub-image and selected face to 640/rand $(16, 32, 64, 128, 256, 512)$. For the remaining $3/5$ probability, we adopt data augmentation similar to SSD [20]. In order to improve the recall rate of faces and ensure anchor classification ability simultaneously, we set Intersection-over-Union (IoU) threshold 0.4 to assign anchor to its ground-truth faces.

4. Experiments

4.1. Implementation Details

First, we present the details in implementing our network. The backbone networks are initialized by the pretrained VGG/ResNet on ImageNet. All newly added convolution layers' parameters are initialized by the 'xavier' method. We use SGD with 0.9 momentum, 0.0005 weight decay to fine-tune our DSFD model. The batch size is set to 16. The learning rate is set to 10^{-3} for the first 40k steps, and we decay it to 10^{-4} and 10^{-5} for two 10k steps.

During inference, the first shot's outputs are ignored and the second shot predicts top 5k high confident detections. Non-maximum suppression is applied with jaccard overlap of 0.3 to produce top 750 high confident bounding boxes per image. For 4 bounding box coordinates, we round down top left coordinates and round up width and height to expand the detection bounding box. The official code has been released at: https://github.com/TencentYoutuResearch/FaceDetection-DSFD.

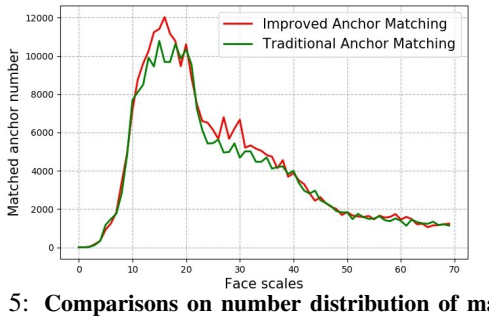

Figure 5: **Comparisons on number distribution of matched anchor for ground truth faces** between traditional anchor matching (blue line) and our improved anchor matching (red line). we actually set the IoU threshold to 0.35 for the traditional version. That means even with a higher threshold (*i.e.*, 0.4), using our IAM, we can still achieve more matched anchors. Here, we choose a slightly higher threshold in IAM so that to better balance the *number* and *quality* of the matched faces.

4.2. Analysis on DSFD

In this subsection, we conduct extensive experiments and ablation studies on the WIDER FACE dataset to evaluate the effectiveness of several contributions of our proposed framework, including feature enhance module, progressive anchor loss, and improved anchor matching. For fair comparisons, we use the same parameter settings for all the experiments, except for the specified changes to the components. All models are trained on the WIDER FACE training set and evaluated on validation set. To better understand DSFD, we select different baselines to ablate each component on how this part affects the final performance.

Feature Enhance Module First, We adopt anchor designed in S3FD [39], PyramidBox [27] and six original feature maps generated by VGG16 to perform classification and regression, which is named Face SSD (FSSD) as the baseline. We then use VGG16-based FSSD as the baseline to add feature enchance module for comparison. Table 2 shows that our feature enhance module can improve VGG16-based FSSD from 92.6%, 90.2%, 79.1% to 93.0%, 91.4%, 84.6%.

Progressive Anchor Loss Second, we use Res50-based FSSD as the baseline to add progressive anchor loss for comparison. We use four residual blocks' ouputs in ResNet to replace the outputs of conv3_3, conv4_3, conv5_3, conv_fc7 in VGG. Except for VGG16, we do not perform layer normalization. Table 3 shows our progressive anchor loss can improve Res50-based FSSD using FEM from 95.0%, 94.1%, 88.0% to 95.3%, 94.4%, 88.6%.

Improved Anchor Matching To evaluate our improved anchor matching strategy, we use Res101-based FSSD without anchor compensation as the baseline. Table 4 shows that our improved anchor matching can improve Res101-based FSSD using FEM from 95.8%, 95.1%, 89.7% to 96.1%, 95.2%, 90.0%. Finally, we can improve our DSFD to 96.6%, 95.7%, 90.4% with ResNet152 as the backbone.

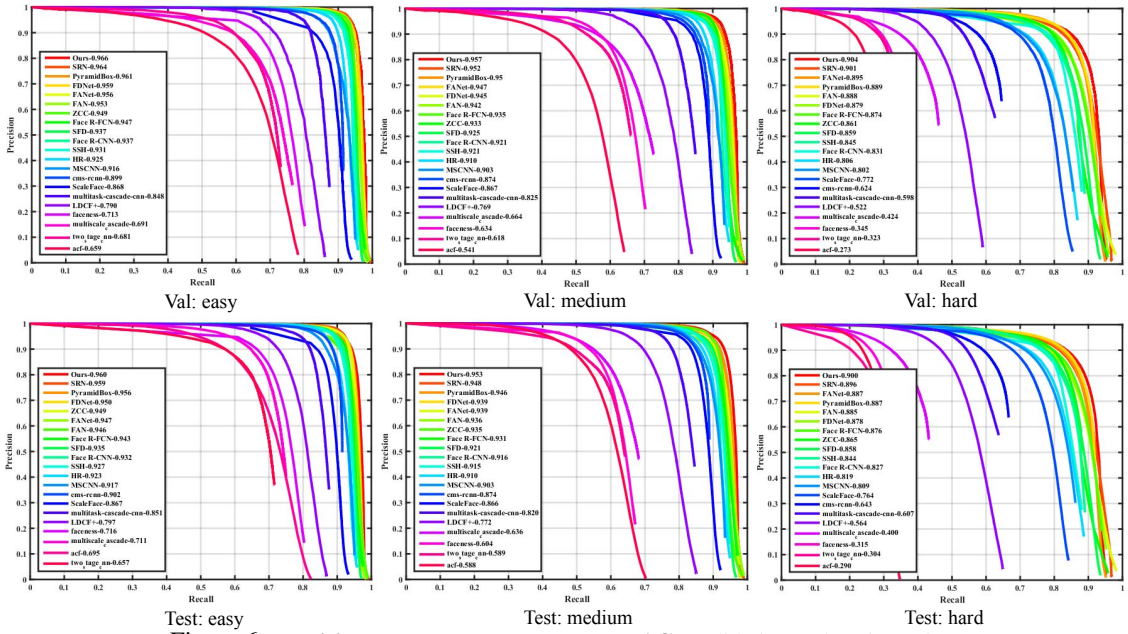

Figure 6: **Precision-recall curves on WIDER FACE** validation and testing subset.

Table 4: Effectiveness of Improved Anchor Matching on the AP performance.

Component	Easy	Medium	Hard
FSSD+RES101	95.1%	93.6%	83.7%
FSSD+RES101+FEM	95.8%	95.1%	89.7%
FSSD+RES101+FEM+IAM	96.1%	95.2%	90.0%
FSSD+RES101+FEM+IAM+PAL	96.3%	95.4%	90.1%
FSSD+RES152+FEM+IAM+PAL	**96.6**%	**95.7**%	90.4%
FSSD+RES152+FEM+IAM+PAL+LargeBS	96.4%	**95.7**%	**91.2**%

Table 5: Effectiveness of different backbones.

Component	Params	ACC@Top-1	Easy	Medium	Hard
FSSD+RES101+FEM+IAM+PAL	399M	77.44%	96.3%	95.4%	90.1%
FSSD+RES152+FEM+IAM+PAL	459M	78.42%	**96.6**%	**95.7**%	**90.4**%
FSSD+SE-RES101+FEM+IAM+PAL	418M	78.39%	95.7%	94.7%	88.6%
FSSD+DPN98+FEM+IAM+PAL	515M	79.22%	96.3%	95.5%	**90.4**%
FSSD+SE-RESNeXt101_32×4d+FEML+IAM+PA	416M	80.19%	95.7%	94.8%	88.9%

Table 6: **FEM vs. RFB** on WIDER FACE.

Backbone - ResNet101 (%)	Easy	Medium	Hard
DSFD (RFB)	96.0	94.5	87.2
DSFD (FPN) / (FPN+RFB)	96.2 / 96.2	95.1 / 95.3	89.7 / 89.9
DSFD (FEM)	**96.3**	**95.4**	**90.1**

Besides, Fig. 4 shows that our improved anchor matching strategy greatly increases the number of ground truth faces that are closed to the anchor, which can reduce the contradiction between the discrete anchor scales and continuous face scales. Moreover, Fig. 5 shows the number distribution of matched anchor number for ground truth faces, which indicates our improved anchor matching can significantly increase the matched anchor number, and the averaged number of matched anchor for different scales of faces can be improved from 6.4 to about 6.9.

Comparison with RFB Our FEM differs from RFB in two aspects. First, our FEM is based on FPN to make full use of feature information from different spatial levels, while RFB ignores. Second, our FEM adopts stacked dilation convolutions in a multi-branch structure, which efficiently leads to larger Receptive Fields (RF) than RFB that only uses one dilation layer in each branch, *e.g.*, R^3 in FEM compared to R in RFB where indicates the RF of one dilation convolution. Tab. 6 clearly demonstrates the superiority of our FEM over RFB, even when RFB is equipped with FPN.

From the above analysis and results, some promising conclusions can be drawn: 1) Feature enhance is crucial. We use a more robust and discriminative feature enhance module to improve the feature presentation ability, especially for hard face. 2) Auxiliary loss based on progressive

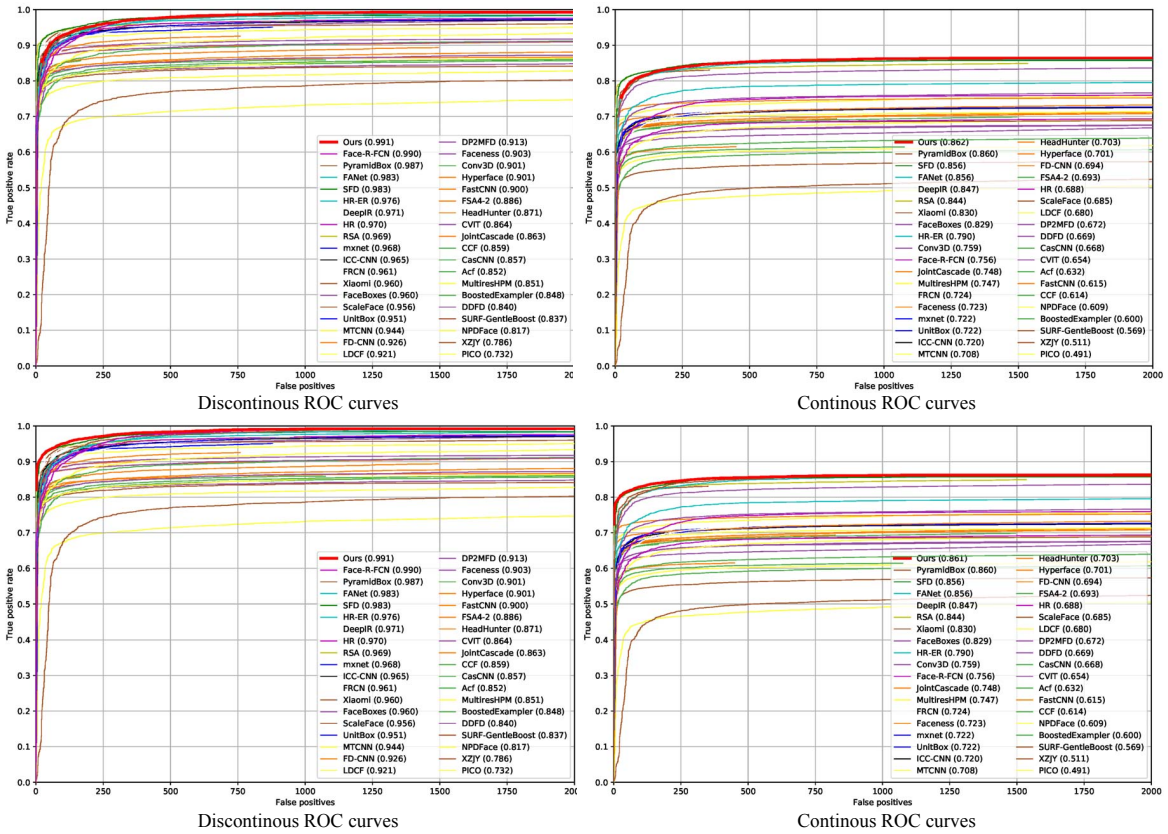

Figure 7: **Comparisons with popular state-of-the-art methods on the FDDB dataset**. The first row shows the ROC results without additional annotations, and the second row shows the ROC results with additional annotations.

anchor is used to train all 12 different scale detection feature maps, and it improves the performance on easy, medium and hard faces simultaneously. 3) Our improved anchor matching provides better initial anchors and ground-truth faces to regress anchor from faces, which achieves the improvements of 0.3%, 0.1%, 0.3% on three settings, respectively. Additionally, when we enlarge the training batch size (*i.e.*, LargeBS), the result in hard setting can get 91.2% AP.

Effects of Different Backbones To better understand our DSFD, we further conducted experiments to examine how different backbones affect classification and detection performance. Specifically, we use the same setting except for the feature extraction network, we implement SE-ResNet101, DPN-98, SE-ResNeXt101_32$\times$4d following the ResNet101 setting in our DSFD. From Table 5, DSFD with SE-ResNeXt101_32$\times$4d got 95.7%, 94.8%, 88.9%, on easy, medium and hard settings respectively, which indicates that more complexity model and higher Top-1 ImageNet classification accuracy may not benefit face detection AP. Therefore, in our DSFD framework, better performance on classification are not necessary for better performance on detection, which is consistent to the conclusion claimed in [11, 16]. Our DSFD enjoys high inference speed benefited from simply using the second shot detection results. For VGA resolution inputs to Res50-based DSFD, it runs 22 FPS on NVIDA GPU P40 during inference.

4.3. Comparisons with State-of-the-Art Methods

We evaluate the proposed DSFD on two popular face detection benchmarks, including WIDER FACE [35] and Face Detection Data Set and Benchmark (FDDB) [12]. Our model is trained only using the training set of WIDER FACE, and then evaluated on both benchmarks without any further fine-tuning. We also follow the similar way used in [31] to build the image pyramid for multi-scale testing and use more powerful backbone similar as [4].

WIDER FACE Dataset It contains $393,703$ annotated faces with large variations in scale, pose and occlusion in total $32,203$ images. For each of the 60 event classes, 40%, 10%, 50% images of the database are randomly selected as training, validation and testing sets. Besides, each subset is further defined into three levels of difficulty: 'Easy', 'Medium', 'Hard' based on the detection rate of a baseline detector. As shown in Fig. 6, our DSFD achieves the best performance among all of the state-of-the-art face detectors based on the average precision (AP) across the three subsets, *i.e.*, 96.6% (Easy), 95.7% (Medium) and 90.4% (Hard) on validation set, and 96.0% (Easy), 95.3% (Medium) and

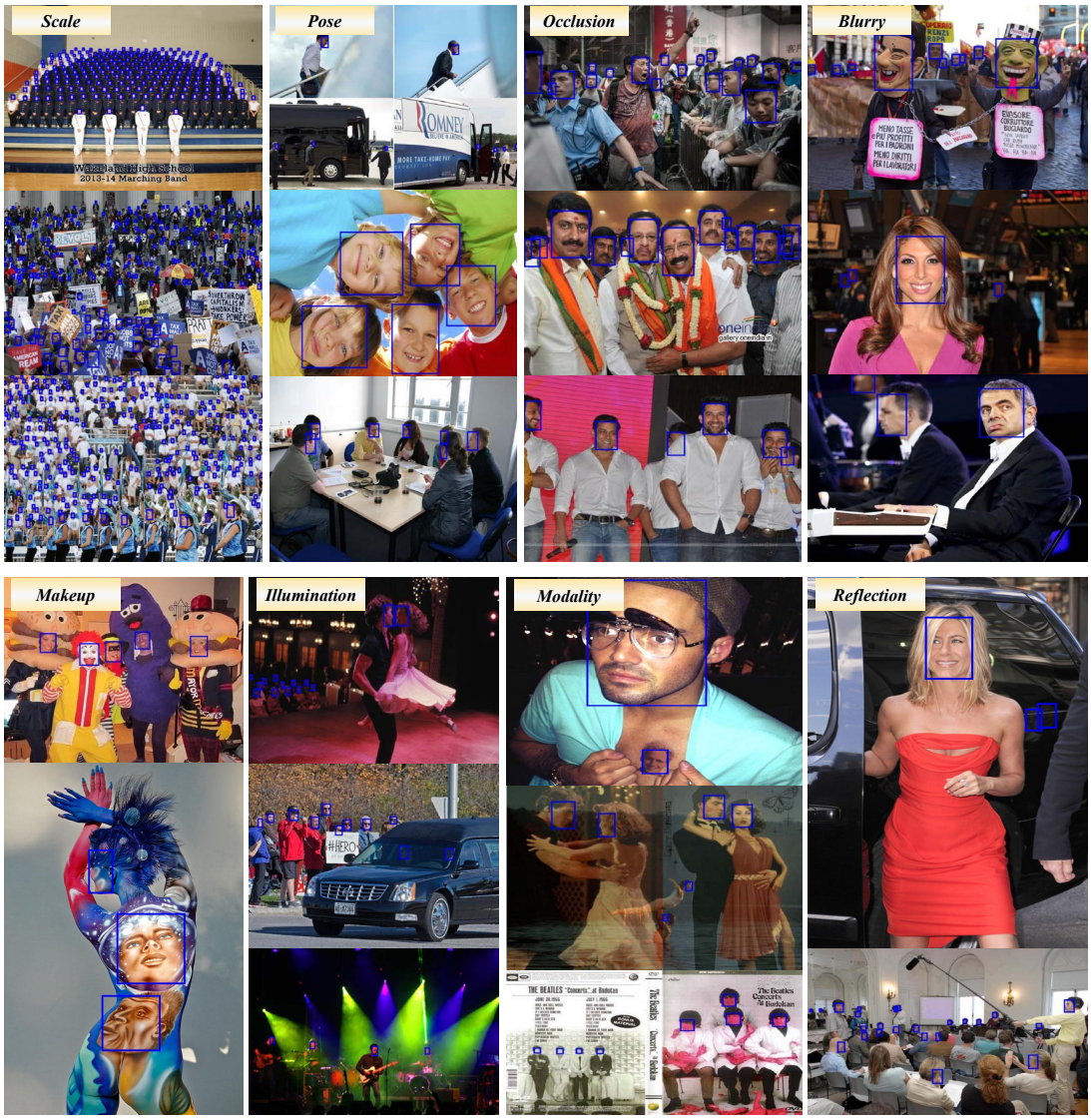

Figure 8: **Illustration of our DSFD to various large variations** on scale, pose, occlusion, blurry, makeup, illumination, modality and reflection. Blue bounding boxes indicate the detector confidence is above 0.8.

90.0% (Hard) on test set. Fig. 8 shows more examples to demonstrate the effects of DSFD on handling faces with various variations, in which the blue bounding boxes indicate the detector confidence is above 0.8.

FDDB Dataset It contains 5,171 faces in 2,845 images taken from the faces in the wild data set. Since WIDER FACE has bounding box annotation while faces in FDDB are represented by ellipses, we learn a post-hoc ellipses regressor to transform the final prediction results. As shown in Fig. 7, our DSFD achieves state-of-the-art performance on both discontinuous and continuous ROC curves, *i.e.* 99.1% and 86.2% when the number of false positives equals to 1,000. After adding additional annotations to those unlabeled faces [39], the false positives of our model can be further reduced and outperform all other methods.

5. Conclusions

This paper introduces a novel face detector named Dual Shot Face Detector (DSFD). In this work, we propose a novel Feature Enhance Module that utilizes different level information and thus obtains more discriminability and robustness features. Auxiliary supervisions introduced in early layers by using smaller anchors are adopted to effectively facilitate the features. Moreover, an improved anchor matching method is introduced to match anchors and ground truth faces as far as possible to provide better initialization for the regressor. Comprehensive experiments are conducted on popular face detection benchmarks, FDDB and WIDER FACE, to demonstrate the superiority of our proposed DSFD compared with the state-of-the-art face detectors, *e.g.*, SRN and PyramidBox.

References

[1] Yotam Abramson, Bruno Steux, and Hicham Ghorayeb. Yet even faster (yef) real-time object detection. *International Journal of Intelligent Systems Technologies and Applications*, 2(2-3):102–112, 2007. 2

[2] Zhaowei Cai, Quanfu Fan, Rogerio S Feris, and Nuno Vasconcelos. A unified multi-scale deep convolutional neural network for fast object detection. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2016. 2

[3] Yu Chen, Ying Tai, Xiaoming Liu, Chunhua Shen, and Jian Yang. Fsrnet: End-to-end learning face super-resolution with facial priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1

[4] Cheng Chi, Shifeng Zhang, Junliang Xing, Zhen Lei, Stan Z Li, and Xudong Zou. Selective refinement network for high performance face detection. In *Proceedings of Association for the Advancement of Artificial Intelligence (AAAI)*, 2019. 2, 7

[5] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2016. 2

[6] Jiankang Deng, Jia Guo, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. *arXiv:1801.07698v1*, 2018. 1

[7] Ross Girshick. Fast r-cnn. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2015. 2, 3

[8] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 580–587, 2014. 2

[9] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Hypercolumns for object segmentation and fine-grained localization. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[11] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, and Kevin Murphy. Speed/accuracy trade-offs for modern convolutional object detectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 7

[12] Vidit Jain and Erik Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. Technical report, Technical Report UM-CS-2010-009, University of Massachusetts, Amherst, 2010. 7

[13] Kobi Levi and Yair Weiss. Learning object detection from a small number of examples: the importance of good features. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004. 2

[14] Haoxiang Li, Zhe Lin, Xiaohui Shen, Jonathan Brandt, and Gang Hua. A convolutional neural network cascade for face detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[15] Jian Li, Jianjun Qian, and Jian Yang. Object detection via feature fusion based single network. In *IEEE International Conference on Image Processing*, 2017. 2

[16] Zeming Li, Chao Peng, Gang Yu, Xiangyu Zhang, Yangdong Deng, and Jian Sun. Detnet: A backbone network for object detection. In *Proceedings of European Conference on Computer Vision*, 2018. 7

[17] Tsung-Yi Lin, Piotr Dollár, Ross B Girshick, Kaiming He, Bharath Hariharan, and Serge J Belongie. Feature pyramid networks for object detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 4

[18] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2017. 2, 3

[19] Songtao Liu, Di Huang, and Yunhong Wang. Receptive field block net for accurate and fast object detection. In *Proceedings of European Conference on Computer Vision*, 2018. 2, 4

[20] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Proceedings of European conference on computer vision (ECCV)*, 2016. 2, 3, 5

[21] Wei Liu, Andrew Rabinovich, and Alexander Berg. Parsenet: Looking wider to see better. In *Proceedings of International Conference on Learning Representations Workshop*, 2016. 2

[22] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[23] Mahyar Najibi, Pouya Samangouei, Rama Chellappa, and Larry S Davis. Ssh: Single stage headless face detector. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2017. 2

[24] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2015. 2, 4

[25] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2014. 2

[26] Ying Tai, Yicong Liang, Xiaoming Liu, Lei Duan, Jilin Li, Chengjie Wang, Feiyue Huang, and Yu Chen. Towards highly accurate and stable face alignment for high-resolution videos. In *The AAAI Conference on Artificial Intelligence (AAAI)*, 2019. 1

[27] Xu Tang, Daniel K Du, Zeqiang He, and Jingtuo Liu. Pyramidbox: A context-assisted single shot face detector. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2018. 2, 3, 5

[28] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104(2):154–171, 2013. 2

[29] Paul Viola and Michael J Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, 2004. 1, 2

[30] Hao Wang, Zhifeng Li, Xing Ji, and Yitong Wang. Face r-cnn. *arXiv preprint arXiv:1706.01061*, 2017. 2

[31] Jianfeng Wang, Ye Yuan, and Gang Yu. Face attention network: An effective face detector for the occluded faces. *arXiv preprint arXiv:1711.07246*, 2017. 3, 7

[32] Xinlong Wang, Tete Xiao, Yuning Jiang, Shuai Shao, Jian Sun, and Chunhua Shen. Repulsion loss: Detecting pedestrians in a crowd. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[33] Yitong Wang, Xing Ji, Zheng Zhou, Hao Wang, and Zhifeng Li. Detecting faces using region-based fully convolutional networks. *arXiv preprint arXiv:1709.05256*, 2017. 2

[34] Jian Yang, Lei Luo, Jianjun Qian, Ying Tai, Fanlong Zhang, and Yong Xu. Nuclear norm based matrix regression with applications to face recognition with occlusion and illumination changes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 39(1):156–171, 2017. 1

[35] Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang. Wider face: A face detection benchmark. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 3, 7

[36] Changzheng Zhang, Xiang Xu, and Dandan Tu. Face detection using improved faster rcnn. *arXiv preprint arXiv:1802.02142*, 2018. 2

[37] Jialiang Zhang, Xiongwei Wu, Jianke Zhu, and Steven CH Hoi. Feature agglomeration networks for single stage face detection. *arXiv preprint arXiv:1712.00721*, 2017. 2, 3

[38] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016. 2

[39] Shifeng Zhang, Xiangyu Zhu, Zhen Lei, Hailin Shi, Xiaobo Wang, and Stan Z Li. Sˆ 3fd: Single shot scale-invariant face detector. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2017. 2, 3, 5, 8