Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 5582132, 10 pages
https://doi.org/10.1155/2021/5582132

Research Article

A Multilevel Single Stage Network for Face Detection

Kanghua Hui,[1] **Jin Wang**,[2] **Huaiqing He**,[1] **and W. H. Ip**[3]

[1] *Department of Computer Science and Technology, Civil Aviation University of China, 300300, China*
[2] *Department of Air Traffic Management, Civil Aviation University of China, 300300, China*
[3] *Department of Mechanical Engineering, University of Saskatchewan, S7K6P1, Canada*

Correspondence should be addressed to Jin Wang; jinwang_2019@aliyun.com

Received 13 January 2021; Revised 1 February 2021; Accepted 15 February 2021; Published 27 February 2021

Academic Editor: Chi-Hua Chen

Recently, tremendous strides have been made in generic object detection when used to detect faces, and there are still some remaining challenges. In this paper, a novel method is proposed named multilevel single stage network for face detection (MSNFD). Three breakthroughs are made in this research. Firstly, multilevel network is introduced into face detection to improve the efficiency of anchoring faces. Secondly, enhanced feature module is adopted to allow more feature information to be collected. Finally, two-stage weight loss function is employed to balance network of different levels. Experimental results on the WIDER FACE and FDDB datasets confirm that MSNFD has competitive accuracy to the mainstream methods, while keeping real-time performance.

1. Introduction

Face detection, the basis of face alignment [1, 2], face recognition [3, 4], facial expression analysis [5, 6], and other related facial problems, has always been a hot issue and widely applied in terms of computer vision. As the major breakthrough for face detection, Viola-Jones [7] used the handcrafted features for cascading detection. There have already been lots of researches devoted to exploring the methods for extracting features effectively [8] and designing more complex handcrafted features or higher-efficiency cascade structure based on Viola-Jones detector.

Recently, with the development of deep learning, convolutional neural networks (CNN), great advance has been achieved and practically applied in image classification [9] and semantic segmentation [10], which also inspired the research of face detectors that can be classified into two modes. One integrates CNN into traditional detection structure. For example, Cascade CNN [11] successfully applied CNN to traditional detection methods through a simple to complex cascade classification network. However, most of these methods use the constructing of Gaussian pyramid image and the sliding window to detect thus resulting in unbalanced samples with negative samples increased and causing RAM to be occupied severely and unable to process images with high resolution. The other focuses on the improvement of generic object detection based on the belief that face is merely another specific object to be detected. For example, Face R-CNN [12] exploited several new techniques including new multitask loss function design, online hard example mining, and multiscale training strategy to improve Faster R-CNN [13] in multiple aspects. S3FD [14] proposed a scale-equitable face detection framework to handle different scales of faces well based on SSD [15]. ISRN [16] presented an improved SRN face detector by combining some useful techniques together from generic object detection and so on.

Although all of these methods improved detection accuracy for ordinary faces, they are difficult to handle small faces fast and effectively. There are also some studies on small faces. HR [17] provided an in-depth analysis of image resolution, object scale, and spatial context for the purposes of finding small faces. Bai [18] proposed an algorithm to directly generate a clear high-resolution face from a blurry small one by adopting a generative adversarial network. PyramidBox++ [19] improved each part to further boost the performance, including Balanced-data-anchor-sampling, Dual-PyramidAnchors, and Dense Context Module. These three methods are recognized as achieving state-of-the-art performance on public dataset but suffer from time-consuming inference. The multilevel single stage network for face detection is inspired by the second mode.

The modern object detectors can be roughly divided into two groups: two-stage face detectors and one-stage detectors. Two-stage detector achieves the better performance but has low time efficiency, for example, SSFD$^+$ [20] focus on achieving comparable performance and simplifying the network architecture for detecting multiscale faces while it spends 582 ms on detecting a picture. Conversely, one-stage detector has faster speed. Face detection has the high demand of speed in real applications, and therefore, one-stage methods are chosen for the backbone. YOLO v3 [21], the version of the YOLO, has fast detection speed, simple structure, low false positive, and strong versatility. The idea of YOLO v3 is to divide an image into an $S \times S$ grid, and each grid cell is parameterized relative to some reference boxes through anchors regressing the bounding box coordinates, an objectness score, and a class probability according to anchors. Although SSD is also an excellent one-stage method and uses multiscale feature map, it is worse for small objects because semantic value for bottom layer is not high at least not higher than YOLO v3. Recently, there are lots of researchers focusing on detecting faces based on YOLO v3. For example, Gurkan [22] achieved 6% improvement on the detection rate of small sized faces, however, with the expense of a longer inference time. Thus, a multilevel single stage network for face detection based on YOLO v3 is designed and gets excellent results especially for small faces on the public datasets.

Although YOLO v3 performs well, it still causes some problems when directly used for face detecting without any adaptation especially for small faces. Lower features are used to detect smaller faces for YOLO v3. However, the lower feature with less semantic information will result in difficult detection for small faces. In addition, many factors, such as sample-unbalanced, severely challenge face detection. The initiative of the research is summarized as follows:

(1) Using multilevel network structure with more anchor scales to detect smaller faces

(2) Adopting enhanced feature module to add contextual and multiscale information to improve the ability of detecting small faces

(3) Balancing the outputs of networks of each level with two-stage weight loss function to optimize the network training

(4) Achieving excellent results on FDDB and WIDER FACE datasets with real-time detection speed

2. Multilevel Single Stage Network for Face Detection

This section will explain MSNFD through following aspects: the basic framework, enhanced feature module, and two-stage weight loss function.

2.1. Basic Framework. YOLO v3 is the preferred choice compared to other network architectures because the backbone of YOLO v3 is simple, and it is the faster one of the methods which have excellent performance for small faces. The basic framework of the method is shown in Figure 1. Some layers which are not critical are not drawn to maintain its visual clarity. First of all, the network is single stage, and the backbone network, Darknet-53, is the same as YOLO v3. Four residual networks used are named ResNet_2, ResNet_8, ResNet_8, and ResNet_4. The residual network is shown at the bottom left of Figure 1. The feature maps from ResNet_2, ResNet_8, ResNet_8, and ResNet_4 of the backbone are separately named Ori_0, Ori_1, Ori_2, and Ori_3 after a series of convolution and concatenating. The module "concat" in Figure 1 indicates multiscale fusion which is the same as YOLO v3. Up-sample small scales to the large scale, and then let them merge into a longer tensor.

Then, the enhanced feature module is added into the basic framework. The obtained feature maps are named Enh_1, Enh_2, and Enh_3, as shown in Figure 1. Ori_1, Ori_2, Ori_3, Enh_1, Enh_2, and Enh_3 constitute the first-level detection network. The first-level network has 6 outputs. The 6 outputs are named 13×13, 26×26, 52×52, 13×13 enhance, 26×26 enhance, and 52×52 enhance, respectively. Each output is encoded as an $S \times S \times (B \times 5 + C)$ tensor. The tensor contains $B = 3$ bounding boxes and $C = 1$ class probability for each one of the $S \times S$ grid. As shown in Figure 1, the 6 outputs have three different grids: 13×13, 26×26, and 52×52, respectively. For each one of the B bounding boxes, it generates coordinates and an objectness score. Each bounding box has one preset anchor. Different grids have bounding boxes with different sizes (width and length) of preset anchors. Each size of anchors is called as anchor scale. Increasing the number of preset anchor scales improves the matching ability between faces and anchors. That is to say, each bounding box is easier to be regressed to generate results.

The second-level network is formed by first-level network merging with low layer feature map, Ori_0. The input image is divided into a smaller and denser 104×104 grid for second-level network. The output of second-level network is named 104×104. The second-level network improves the detection capability of small faces particularly. Firstly, the smaller each grid is, the smaller the anchor scale is. The second-level network is attached with three smaller anchor scales to ensure matching small faces. Furthermore, to address the problem that the lower feature with less semantic information will result in difficult detection for small faces, the first-level network is integrated into the second-level network so that second-level network enriches semantic information greatly. In summary, it is a theoretical analysis that the proposed structure has detection capability particularly for small faces.

2.2. Enhanced Feature Module. The enhanced feature module borrows the idea of dilated convolution [23]. Small faces have little pixels at lower layer of the network. As the convolution goes on, the features left at high layers will be too few to get sufficient semantic information which will make it difficult

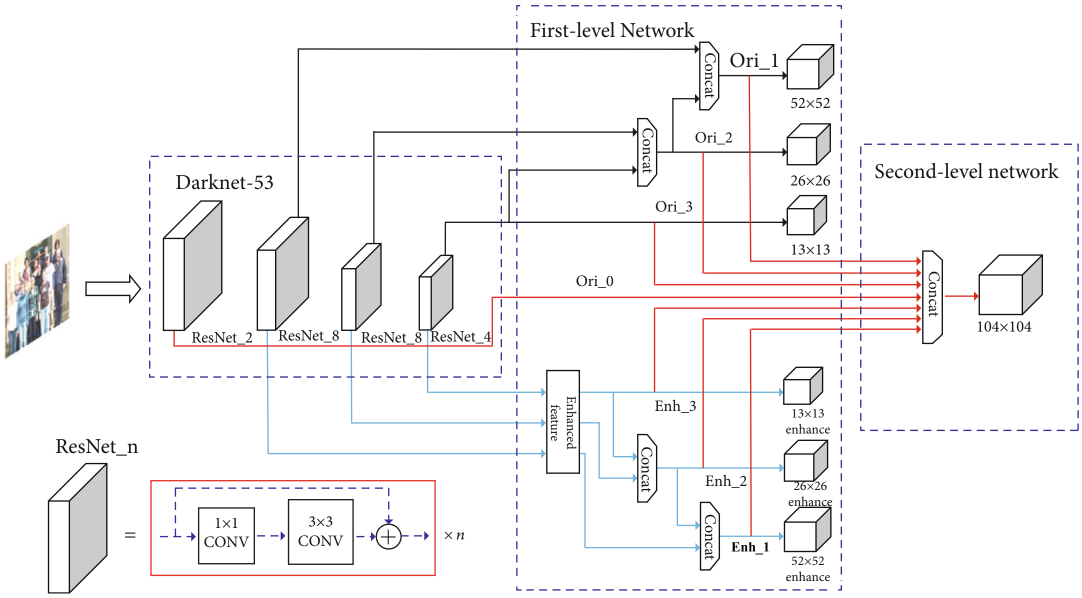


Figure 1: Basic framework of MSNFD. The basic framework consists of backbone network, first-level network, and second-level network. The schematic of ResNet_n is at the lower left corner.

to detect small faces. And adding context always helps for detecting small faces [17], and it is crucial to enlarge the receptive field to add context. As discussed above, enhanced feature module is proposed. The enhanced feature module is composed of three common convolutions (dilated rate = 1) and two dilated convolutions (dilated rate = 2), as shown in Figure 2. This module has three branches. The rectangular blocks are common convolutions, and the dotted blocks are dilated convolutions. Different dilated rates are shown at the bottom left of Figure 2. The first branch of the module is designed to retain information of origin feature map. The common convolution this branch only has is used to reduce the dimension. The second branch combines a dilated convolution with a common convolution. The dilated convolution is shaped by a 3×3 kernel and 2 dilated rate. N dilated rate is considered as inserting n-1 zeros between every two values in the convolution kernel. Downsampling is a common way to enlarge the receptive field, but it results in reducing the feature map and loss of some important information of small faces.

Dilated convolution can also enlarge the receptive field and fuse contextual information effectively [23] especially for small faces with the same number of parameters. It will not change the size of feature map. As shown in Figures 3(a) and 3(b), the context here means information of the face around, such as the neck and clothes, which benefits for detecting small faces. The third branch further enlarges the receptive field and contains more contextual information based on the second branch.

In addition, the enhanced feature module combines information of receptive fields of different scales through by three branches. That is to say, the module combines both texture and contextual information, respectively, from the common and dilated convolution. Information of features on multiple scales will be enriched so that the enhanced feature module can handle multiscale faces to some extent while detecting small faces performs better.

2.3. Two-Stage Weight Loss Function. Second-level network has different anchor scales from first-level network. Second-level network decreases the anchor scale with lots of simple and easily divided negative samples. Joint training for the first-level and second-level networks is adopted. As it is, the sample imbalance causes unbalanced training. If the loss function between first-level and second-level network has not been balanced to an even state, the whole training process will slide overwhelmingly into the second-level network because there are more samples in it than the first-level. In that case, the first-level network gets insufficient training, and the second-level network performs ill because the second-level is generated from the first-level. Thus, two-stage weight loss function is proposed. Considering the match between the anchor and the loss function, the location loss function remains the same as the distance equation used in clustering. The first stage loss function is defined as equations (1)–(3).

$$L_{\text{first\_output}}(p, p^*, t, t^*) = \sum_{i=1}^{6}\left[\frac{1}{N_{\text{cls}_i}}\sum L_{\text{cls}_i}(p, p^*) + \frac{1}{N_{\text{iou}_i}}\sum p^* L_{\overline{\text{iou}_i}}(t, t^*)\right], \tag{1}$$

$$L_{\text{second\_output}}(p, p^*, t, t^*) = \frac{1}{N_{\text{cls}}}\sum L_{\text{cls}}(p, p^*) + \frac{1}{N_{\overline{\text{iou}}}}\sum p^* L_{\overline{\text{iou}}}(t, t^*), \tag{2}$$

$$L_{\text{one\_stage}} = \frac{N_{\text{second}}}{N_{\text{first}} + N_{\text{second}}} L_{\text{first\_output}} + \frac{N_{\text{first}}}{N_{\text{first}} + N_{\text{second}}} L_{\text{second\_output}}. \tag{3}$$

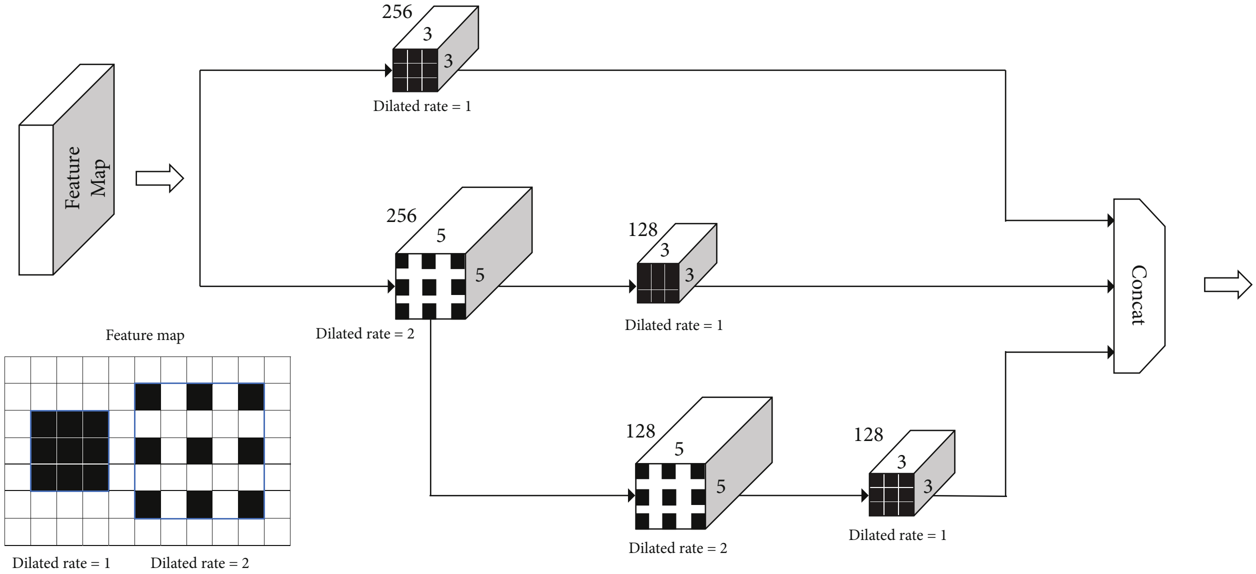


Figure 2: Enhanced feature module. The enhanced feature module is composed of two common convolutions and two dilated convolutions.

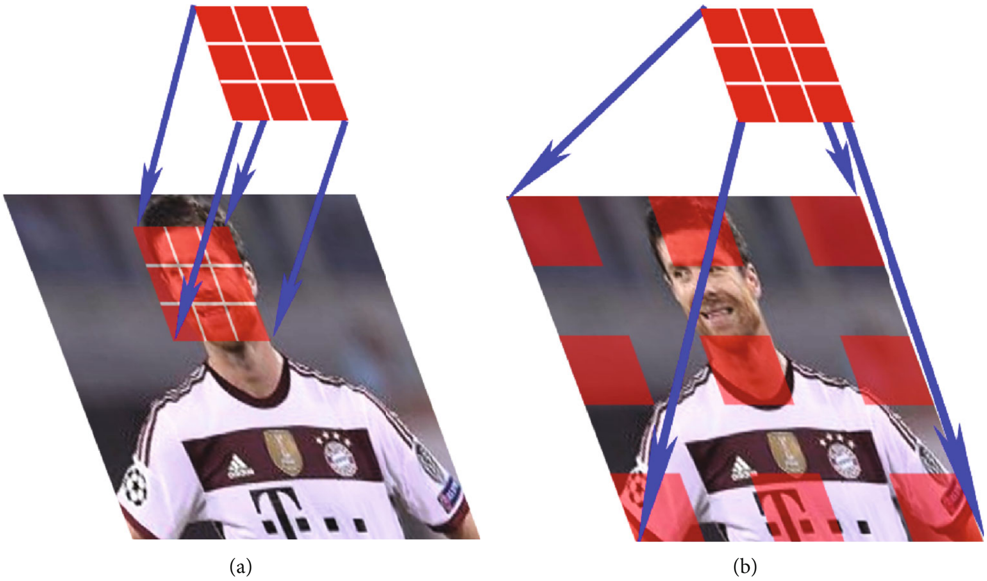


Figure 3: Comparison of different type convolutions: (a) common convolution; (b) dilated convolution.

$L_{\text{first\_output}}$ and $L_{\text{second\_output}}$ refer to the loss function of first-level and second-level network, p is the predicted probability of face, and $p^* = \{0, 1\}$ is the true probability. When the anchor is positive, the value of p^* is 1, otherwise 0. t means the predicted bounding boxes and t^* the ground truth. $L_{\text{cls\_i}}$ is the classification loss over two classes (face vs. background) of the i-th output, and $i = \{1, 2, \cdots, 6\}$ represents six outputs of first-level network. $\text{iou}(\text{t}, \text{t}^*)$ indicates intersection over union (IOU) between the predicted bounding box (t) and the ground truth (t^*). The larger the IOU is, the more accurately the method locates the face. $L_{\overline{\text{iou\_i}}}$ indicates location loss which adopts the equations $1 - \text{iou}(\text{t}, \text{t}^*)$ used in clustering. Moreover, $L_{\overline{\text{iou\_i}}}$ only has effect on positive anchors. L_{cls} and $L_{\overline{\text{iou}}}$ are classification loss and location loss of second-level network, respectively. Equation (3) shows the first-stage loss function, where N_{first} and N_{second} represent the number of anchors in the first- and second-level network, respectively. The second-level network has more anchors, so its proportion in the whole loss function needs to be reduced. Thus, the imbalance of loss function between first- and second-level networks can be eliminated via first-stage training.

After the loss function gets balanced and adjusted through the first-stage training, the first-level network will be fully trained. But it is still difficult to train small faces, so the training of second-level network in the second stage needs to be strengthened by means of increasing the

Table 1: Anchor scale of different methods.

Output	MSNFD	YOLO v3
13 × 13	(106, 140) (166, 220) (320, 400)	(90, 116) (156, 198) (326, 373)
26 × 26	(33, 38) (32, 49) (38, 45)	(30, 61) (45, 62) (59, 119)
52 × 52	(17, 23) (21, 25) (23, 30)	(10, 13) (16, 30) (23, 33)
13 × 13 enhance	(44, 56) (56, 72) (74, 98)	—
26 × 26 enhance	(28, 31) (25, 36) (29, 39)	—
52 × 52 enhance	(13, 13) (13, 17) (15, 18)	—
104 × 104	(6, 7) (9, 11) (11, 14)	—

proportion of the second-level network. The second-stage loss function is defined as equation (4).

$$L_{\text{two\_stage}} = \frac{N_{\text{first}}}{N_{\text{first}} + N_{\text{second}}} L_{\text{first\_output}} + \frac{N_{\text{second}}}{N_{\text{first}} + N_{\text{second}}} L_{\text{second\_output}}. \tag{4}$$

3. Experiments and Analysis

3.1. Training Details. The training dataset used in the research is the WIDER FACE [24] training set including 12,880 images, 62 daily scenes, and a total of 158,945 faces. These faces are also available on scales of a wide range, from tiny to giant. In order to further enhance the model of training robustness, data enhancement is used during training, such as vertical flipping, random cropping, and random panning. The classification loss adopts focal loss to balance the positive and negative samples, and the location loss uses cross-entropy loss. The optimization uses stochastic gradient descent, and the batch-size is set to 8. In the first stage of training, 3×10^5 steps about 100 epochs are iterated with initial learning rate of 10^{-4}. In the second stage, 9.7×10^4 steps about 30 epochs are iterated with initial learning rate of 10^{-6}, based on the first stage but finely tuned. Pretrained Darknet-53 which obtains 55.3%mAP on COCO [25] is used to initialize the parameters.

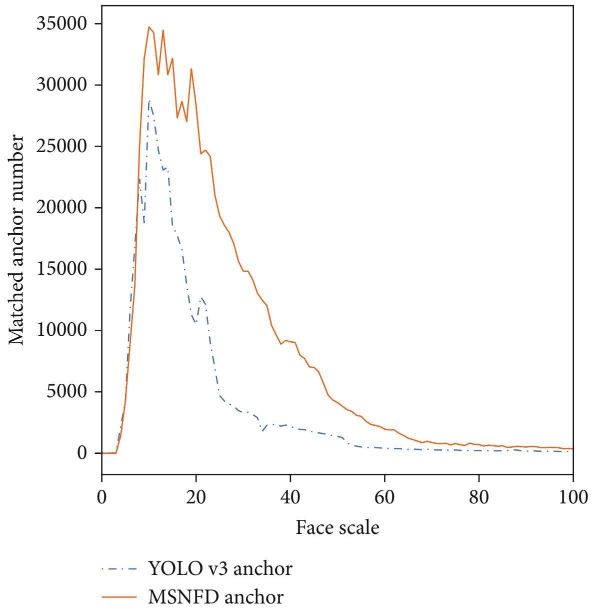


Figure 4: Comparisons on number of anchor matching faces between YOLO v3 and MSNFD.

3.2. Anchor Analysis. YOLO v3 uses k-means clustering to determine the anchor scale. It is shown in Table 1. K -means is also used to cluster but with more cluster centers to get more anchor scales on WIDER FACE training set in this research. As listed in Table 1, the anchor scales range from 6 pixels to 400 pixels and are divided into 7 types, each for a certain output of the multilevel networks. Within the testing framework based on anchors, the matching degree between anchors and faces determines the quality of the eventual result of the final training. Expanding number of the anchor scale improves the matching degree between anchors and faces. MSNFD does not increment B of the output tensor, $S \times S \times (B \times 5 + C)$ simply but adds the output tensors with the same $B = 3$ as YOLO v3 because the multilevel network has more outputs. In this research, anchor scale ranges from YOLO v3's 9 (3×3 outputs) to 21 (3×7 outputs). The 21 kinds of anchor scales added newly by this research, and the previously existing 9 kinds of anchor-scale of YOLO v3 are, respectively, matched with the face size of the training set. If IOU is greater than 0.4, the matching is successful. The statistics of successful matching on different scales is visualized in Figure 4. As shown in Figure 4, the multilevel network has more anchor scales, and the number of successful matching cases is far more than in YOLO v3, which improves the matching rate between the discrete anchor and the continuous face. In summary, the preset anchor is effective, which benefits the multilevel network being fully learned.

3.3. Model Analysis. In order to prove the effectiveness of the proposed methods, different methods are experimented on WIDER FACE validation dataset and Multiattribute Labelled Faces (MALF) dataset [26]. The WIDE FACE validation set is split into three subsets: easy, medium, and hard. "Hard" subset contains faces with much smaller scales, extreme pose, exaggerated expressions, and large portion of occlusion [24]. "Hard" subset is regarded as a particular set of small faces. MALF chooses the size of 60 and 90 to divide the scale range into small, medium, and large intervals corresponding to small, medium, and large subsets. Both datasets are very suitable for evaluating the method. Different parts are

Table 2: Results on MALF set and WIDER FACE validation set.

Method	WIDER FACE				MALF		
	Easy (%)	Medium (%)	Hard (%)	Time (ms)	Large (%)	Medium (%)	Small (%)
RFAB-f-R-FCN	91.3	89.6	79.4	109	—	—	—
PPN	79.5	77.4	57.7	17	—	—	—
XIE	88.5	85.4	73.8	190	—	—	—
HR	91.9	90.8	82.3	377	—	—	—
VJ	41.2	33.3	13.7	20	45.8	35.7	11.4
YOLO v3	87.5	80.9	47.3	29	75.0	73.1	58.5
MSNFD (enhance feature module)	86.6	79.9	52.3	30	77.5	82.6	65.5
MSNFD (first-level)	91.8	87.8	54.5	31	79.5	83.5	69.3
MSNFD (first-level + second-level)	91.2	87.5	67.1	33	79.7	84.1	77.8
MSNFD (first-level + second-level + two-stage train)	93.5	90.7	71.1	33	82.8	88.8	80.2

gradually added to existing YOLO v3 framework to build up MSNFD. The platform's hardware configuration involves CPU i7-9700 and GPU NVIDIA GTX 1080ti. Calculate average time detecting on WIDER FACE validation dataset and the average precision (AP) on WIDER FACE set and MALF set and the results are shown in Table 2. In this section, RFAB-f-R-FCN [27], PPN [28], XIE [29], HR [17], and VJ [7] are also used for being compared to MSNFD. They are the latest methods published in the past two years or the classical methods.

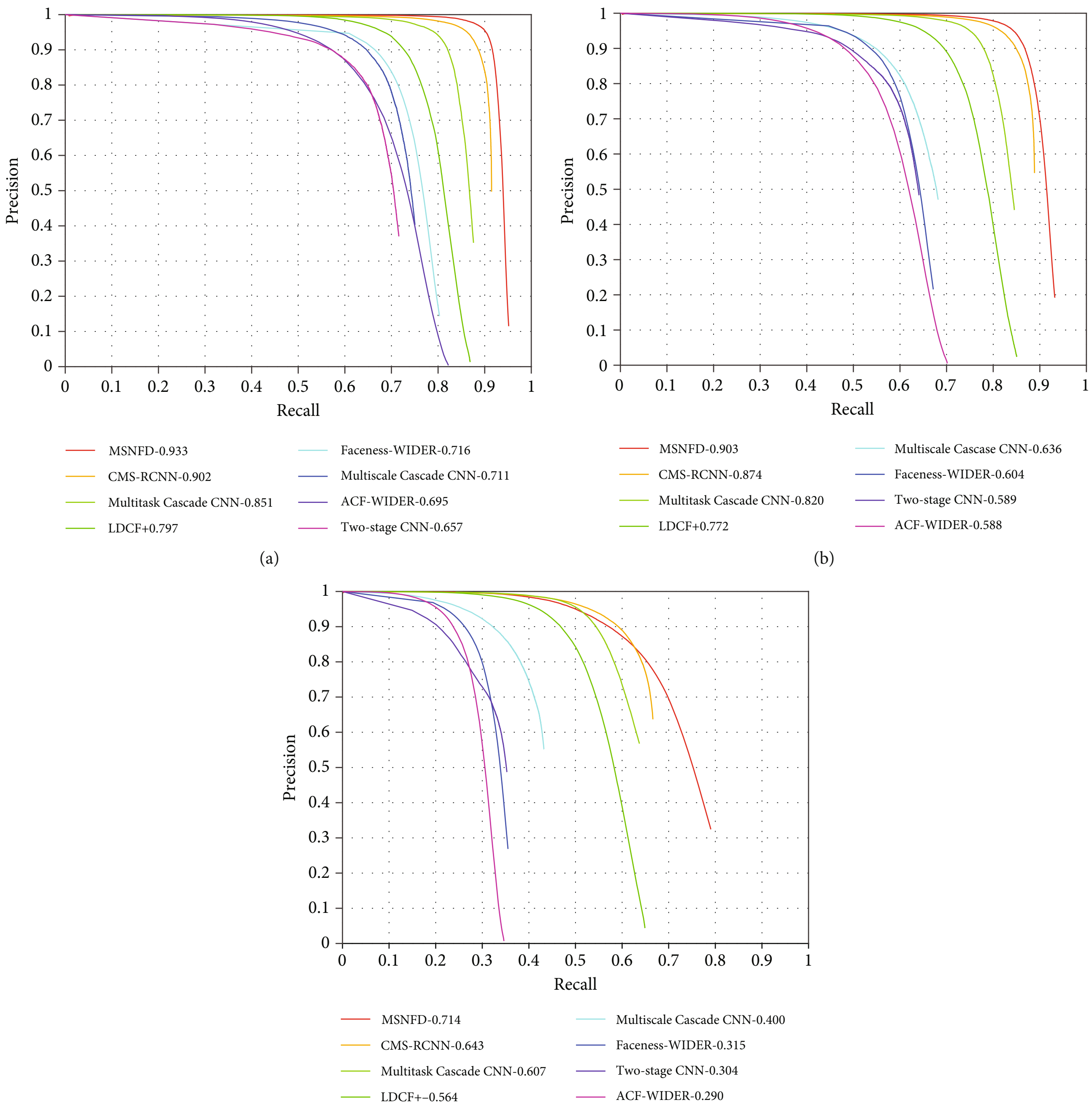
3.3.1. Enhanced Feature Module. Firstly, add enhanced feature module into YOLO v3 only, the result shows that AP increases from 58.5% to 65.5% on the small set of MALF and from 47.3% to 52.3% on the hard set of wider face, because the enhanced feature module can combine the contextual information of small faces to improve the representation ability of features. However, the detection effect drops on the easy and medium set of WIDER FACE. It is unsatisfactory to add enhanced feature module into YOLO v3 only.

3.3.2. First-Level Network. Secondly, in order to reverse the decline of detection capability of large and medium faces, enhanced feature module and the YOLO v3 together constitute the first-level network which raises AP to 79.5%, 83.5%, and 69.3% on MALF and 91.8%, 87.8%, and 54.5% on WIDER FACE. Attention should be paid that the performance of first-level network on the hard dataset is not outstanding enough compared to only adding the enhanced feature module with AP just raising by 3.8% from 65.5% to 69.3% and 2.2% from 52.3% to 54.5%. This is because that YOLO v3 is not ideal enough for small face detection and performs poorly when combined with the enhanced feature module to form the first-level network.

3.3.3. Second-Level Network. Thirdly, in order to further improve the ability of detecting small faces, the second-level network is introduced. The second-level network not only incorporates the contextual and multiscale information from the first-level network but also contains the texture information from low layers. Moreover, it reduces anchor scale to match smaller faces. The above adjustments help to improve the performance of small faces detection. As listed in Table 2, AP on small and hard set raised to 8.5% and 12.6%, which fully indicates that second-level network benefits small faces detecting. In addition, it can be seen that after adding second-level network, AP drops from 91.8% to 91.2% on easy set. As having analysed earlier, the adding of second-level network will cause imbalance of both samples and the loss functions between networks of different levels. The second-level network has more samples and will occupy the main part of learning process, resulting in the inappropriate deviation of normal distribution and the decline of classification ability of the first-level network.

3.3.4. Two-Stage Weight Loss Function. Finally, to further optimize the network, it is necessary to introduce two-stage weight loss function to balance network of different levels. The first-stage training can balance the loss between networks of different levels and solve the problem of performance declining in detecting large faces after adding the second-level network. The second-stage training intends to further optimize the second-level network, that is, to enhance the detection ability for small faces. As shown in Table 2, after using the two-stage weight loss function, whether the face is large or small and easy or hard, the overall performance is largely improved and achieves AP, 82.8%, 88.8%, and 80.2% and 93.5%, 90.7%, and 71.1%.

3.3.5. Comparison and Runtime Analysis. From Table 2, it is noticed that YOLO v3 does not perform as well as PPN on hard subset of WIDER FACE. However, MSNFD has more excellent results than PPN and YOLO v3. It indicates that the better performance is correlated with the method proposed. The AP of MSNFD is 23.8% on hard subset and 21.7% on small subset higher than YOLO v3, though MSNFD is just 4 ms slower than YOLO v3. The best three methods with higher AP on hard subset of WIDER FACE such as 82.3%, 79.4%, and 71.1% are HR, RFAB-f-R-FCN, and MSNFD. HR is a well-known method designed to detect tiny faces. HR provides much better detection performance about 11.2% higher than MSNFD, but HR runs much slower. HR takes 377 ms to detect an image on wider face while MSNFD only needs 33 ms. In addition, the results of MSNFD



(a)

(b)

(c)

Figure 5: Comparisons on WIDER FACE test set: (a) test: easy; (b) test: medium; (c) test: hard.

are rather similar to HR on easy and medium subsets. MSNFD even surpasses HR on easy subset. RFAB-f-R-FCN is designed for small-scale face published in 2020. The run time of MSNFD is 33 ms which is nearly 3 times smaller than RFAB-f-R-FCN. The two fastest methods are PPN which is published in 2019 and VJ. They only need 17 ms and 20 ms. However, their detection results are extremely disappointing especially for small faces. PPN achieves 57.7%, and VJ achieves 13.7% on hard subsets and 11.4% on small subsets. MSNFD runs slower than PPN and VJ a bit but has much higher AP. Moreover, XIE is a newly proposed method in 2019 which has very closed accuracy to MSNFD, but it runs nearly 6 times slower than MSNFD. In other words, MSNFD can balance accuracy and speed well compared to other methods with similar accuracy. In a word, MSNFD improves accuracy greatly while retaining YOLO V3's speed advantage. Both test code and models are available online at https://github.com/JackJinWang/Face-detection/tree/master.

4. Comparisons with Real-Time Methods

To compare MSNFD with other current popular methods which can detect faces at close to real-time speed, tests were taken on WIDER FACE test set and Face Detection Data

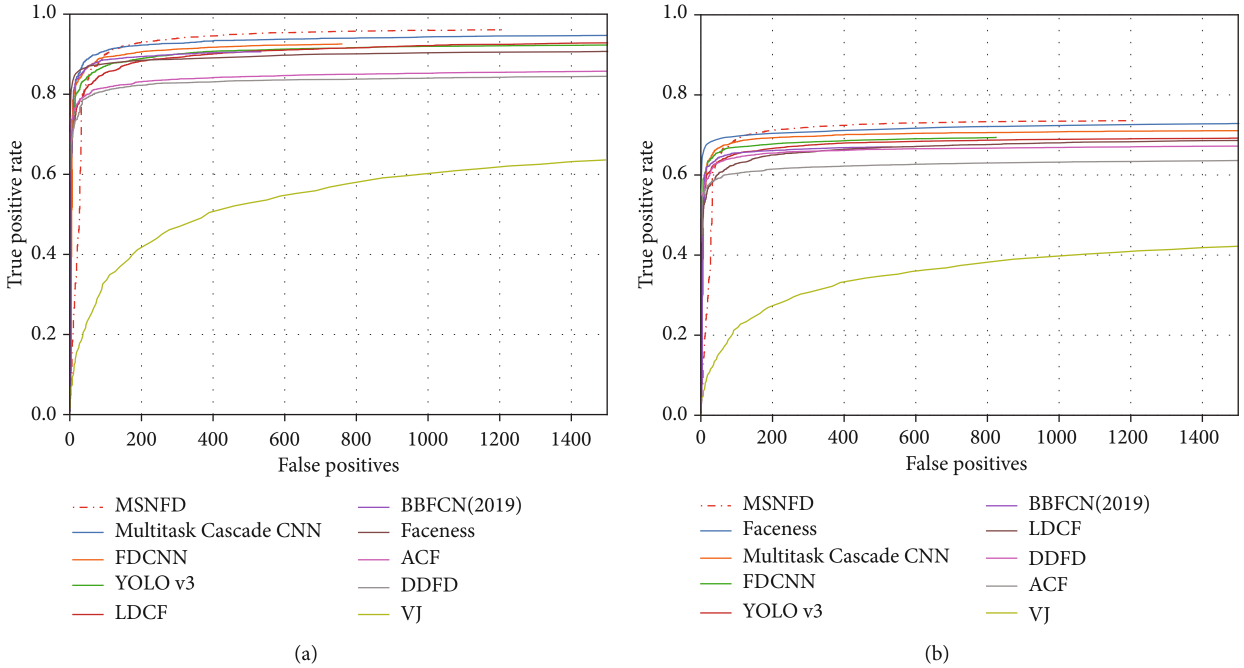


Figure 6: Comparisons on FDDB: (a) discontinuous ROC; (b) continuous ROC.

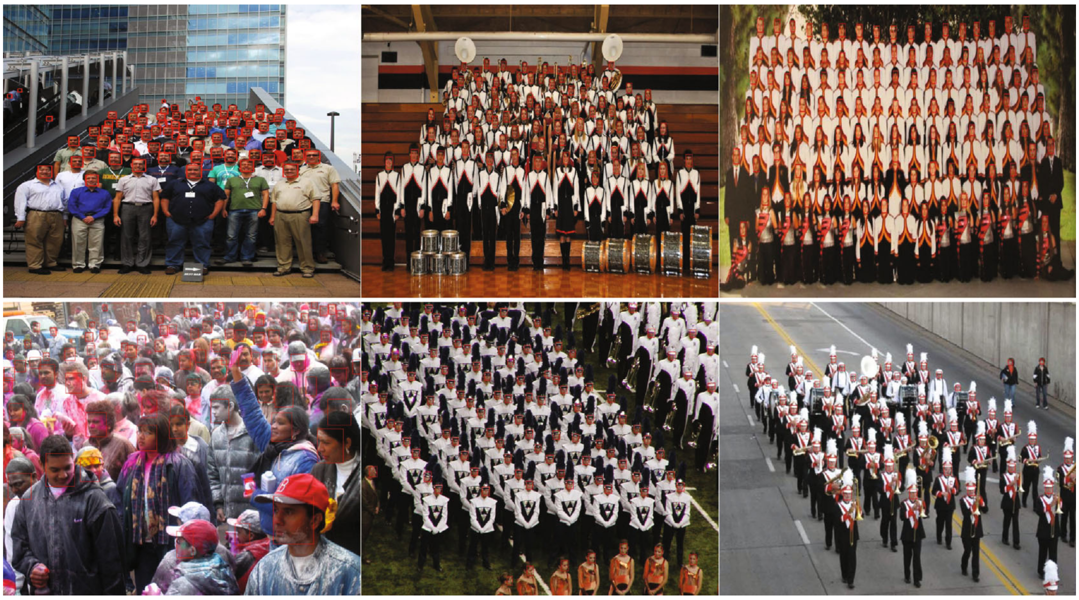


Figure 7: Detection examples.

Set and Benchmark (FDDB) [30]. Furthermore, the result is further shown via detecting a random selection of images with small faces from the public dataset.

4.1. Evaluate on Benchmark

4.1.1. WIDER FACE Dataset. The WIDER FACE test set has 16,097 images. It is a challenging test set. The results of comparison between MSNFD and real-time methods Multitask Cascade CNN [31], CMS-RCNN [32], LDCF [33], Multiscale Cascade CNN [24], Faceness [34], Two-stage CNN [24], and ACF [35] are shown in Figures 5(a)–5(c), where MSNFD achieves 93.3%, 90.3%, and 71.4%AP on easy, medium, and hard test sets of the WIDER FACE, respectively, much better than other methods.

4.1.2. FDDB Dataset. FDDB is also a pretty challenging face detection dataset with 2,845 images of 5,171 faces in total in a variety of states, such as occlusion, rare poses, low resolution, and out of focus. MSNFD is compared to the popular and real-time methods BBFCN [36], DDFD [37], Multitask Cascade CNN [31], FDCNN [38], LDCF [33], YOLO v3

[21], Faceness [34], VJ [7], and ACF [35] on FDDB. The ROC curves are shown in Figures 6(a) and 6(b), which the higher the ROC is, the better the method performs. The ROC curve of MSNFD is higher than other methods indicating that MSNFD performs much better.

4.2. Detection Examples. This section shows some selected examples of small faces detection from WIDER FACE test set (in Figure 7). Though, there are so small and many faces in these images, MSNFD is able to detect accurately. These successful results definitely prove that MSNFD is able to work perfectly especially for small faces.

5. Conclusion

A multilevel single stage network for face detection with more anchor scales is proposed to raise the matching rate between anchors and faces especially small faces. The enhanced feature module in the first-level network not only integrates the contextual information but also strengthens the ability to process multiscale data, enriching the overall information of small face features. The second-level network has smaller anchor scales. In addition, it is generated from the first-level one and is integrated with texture features of low layers, which enhances the process of semantic facial information, especially for small faces. During the training, the two-stage weight loss function is used to balance the network to optimize the classification effect of different levels. MSNFD improves accuracy greatly while retaining YOLO V3's speed advantage. Tests on the WIDER FACE datasets show that MSNFD achieves 93.3%, 90.3%, and 71.4% AP. In a word, MSNFD has greater reference significance for real-time face detection.

Data Availability

The data used to support the findings of this study are available from the authors upon request.

Conflicts of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Scientific Research Project of Tianjin Education Commission (2020KJ024).

References

[1] Y. Feng, F. Wu, X. Shao, Y. Wang, and X. Zhou, "Joint 3D face reconstruction and dense alignment with position map regression network," in *Computer Vision – ECCV 2018: 15th European Conference*, pp. 557–574, Munich, Germany, 2018.

[2] J. Wan, J. Li, Z. Lai, L. Zhang, and B. du, "Robust face alignment by cascaded regression and de-occlusion," *Neural Networks*, vol. 123, pp. 261–272, 2020.

[3] G. Hermosilla, J. L. Verdugo, G. Farias, E. Vera, F. Pizarro, and M. Machuca, "Face recognition and drunk classification using infrared face images," *Journal of Sensors*, vol. 2018, Article ID 5813514, 8 pages, 2018.

[4] Z. Lei, X. Zhang, S. Yang, Z. Ren, and O. F. Akindipe, "RFR-DLVT: a hybrid method for real-time face recognition using deep learning and visual tracking," *Enterprise Information Systems*, vol. 14, no. 9-10, pp. 1379–1393, 2020.

[5] K. Li, Y. Jin, M. W. Akram, R. Han, and J. Chen, "Facial expression recognition with convolutional neural networks via a new face cropping and rotation strategy," *The Visual Computer*, vol. 36, no. 2, pp. 391–404, 2020.

[6] B. Bozorgtabar, D. Mahapatra, and J. P. Thiran, "ExprADA: adversarial domain adaptation for facial expression analysis," *Pattern Recognition*, vol. 100, article 107111, 111 pages, 2020.

[7] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[8] K. Adnan and R. Akbar, "Limitations of information extraction methods and techniques for heterogeneous unstructured big data," *International Journal of Engineering Business Management*, vol. 11, article 1177, 2019.

[9] E. Real, A. Aggarwal, Y. Huang, and Q. V. Le, "Regularized evolution for image classifier architecture search," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 4780–4789, 2019.

[10] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 386–397, 2020.

[11] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5325–5334, Boston, MA, USA, 2015.

[12] H. J. Wang, Z. Li, X. Ji, and Y. Wang, "Face r-cnn," 2017, https://arxiv.org/abs/1706.01061.

[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[14] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang, and S. Li, "S^{3fd}: single shot scale-invariant face detector," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 192–201, Venice, 2017.

[15] W. Liu, D. Anguelov, D. Erhan et al., "Ssd: single shot multibox detector," in *Computer Vision – ECCV 2016: 14th European Conference*, pp. 21–37, Amsterdam, The Netherlands, 2016.

[16] S. Zhang, R. Zhu, X. Wang et al., "Improved selective refinement network for face detection," 2019, https://arxiv.org/abs/1901.06651.

[17] P. Hu and D. Ramanan, "Finding tiny faces," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. , 20171522–1530, 2017.

[18] Y. Bai, Y. Zhang, M. Ding, and B. Ghanem, "Finding tiny faces in the wild with generative adversarial network," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21–30, Salt Lake City, UT, 2018.

[19] Z. Li, X. Tang, J. Han, J. Liu, and R. He, "Pyramidbox++: high performance detector for finding tiny face," 2019, https://arxiv.org/abs/1904.00386.

[20] L. Shi, X. Xu, and I. A. Kakadiaris, "SSFD$^+$: a robust two-stage face detector," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 1, no. 3, pp. 181–191, 2019.

[21] J. Redmon and A. Farhadi, "Yolov3: an incremental improvement," 2018, https://arxiv.org/abs/1804.02767.

[22] F. Gurkan, B. Sagman, and B. Gnsel, "YOLOv3 as a deep face detector," in *2019 11th International Conference on Electrical and Electronics Engineering (ELECO)*, pp. 605–609, Bursa, Turkey, 2019.

[23] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, https://arxiv.org/abs/1511.07122.

[24] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "Wider face: a face detection benchmark," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5525–5533, Las Vegas, NV, USA, 2016.

[25] T.-Y. Lin, M. Maire, S. Belongie et al., "Microsoft coco: common objects in context," in *Computer Vision – ECCV 2014: 13th European Conference*, pp. 740–755, Zurich, Switzerland, 2014.

[26] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Fine-grained evaluation on face detection in the wild," in *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, pp. 1–7, Ljubljana, Slovenia, 2015.

[27] C. Tang, S. Chen, X. Zhou, S. Ruan, and H. Wen, "Small-scale face detection based on improved R-FCN," *Applied Sciences*, vol. 10, no. 12, pp. 4177–4193, 2020.

[28] D. Zeng, H. Liu, F. Zhao, S. Ge, W. Shen, and Z. Zhang, "Proposal pyramid networks for fast face detection," *Information Sciences*, vol. 495, no. 12, pp. 136–149, 2019.

[29] R. Xie, Q. Zhang, E. Yang, and Q. Zhu, "A method of small face detection based on CNN," in *2019 4th International Conference on Computational Intelligence and Applications (ICCIA)*, pp. 78–82, Nanchang, China, 2019.

[30] V. Jain and E. Learned-Miller, *Fddb: a benchmark for face detection in unconstrained settings*, University of Massachusetts, Amherst technical report, 2010.

[31] 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, 2016.

[32] C. Zhu, Y. Zheng, K. Luu, and M. Savvides, *CMS-RCNN: contextual multi-scale region-based CNN for unconstrained face detection*, Springer International Publishing, 2017.

[33] E. Ohn-Bar and M. M. Trivedi, "To boost or not to boost? On the limits of boosted trees for object detection," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 3350–3355, Cancun, 2016.

[34] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "From facial parts responses to face detection: a deep learning approach," in *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 3676–3684, Santiago, Chile, 2015.

[35] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Aggregate channel features for multi-view face detection," in *IEEE International Joint Conference on Biometrics*, pp. 1–8, Clearwater, FL, USA, 2014.

[36] L. Liu, G. Li, Y. Xie, Y. Yu, Q. Wang, and L. Lin, "Facial landmark machines: a backbone-branches architecture with progressive representation learning," *IEEE Transactions on Multimedia*, vol. 21, no. 9, pp. 2248–2262, 2019.

[37] S. S. Farfade, M. J. Saberian, and L.-J. Li, "Multi-view face detection using deep convolutional neural networks," in *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, pp. 643–650, Shanghai, China, 2015.

[38] D. Triantafyllidou, P. Nousi, and A. Tefas, "Fast deep convolutional face detection in the wild exploiting hard sample mining," *Big Data Research*, vol. 11, pp. 65–76, 2018.