# Hourglass Face Detector for Hard Face

Zijun Yu[1], Jian Yin[1], Qiang Zhang[2], *Wenming Yang[1], Jing-Hao Xue[3], Qingmin Liao[1]
[1] *Shenzhen International Graduate School/Department of Electronic Engineering, Tsinghua University, China*
[2] *Beijing Institute of Environmental Feature, Beijing, China*
[3] *Department of Statistical Science, University College London, London, UK*
{1763224962, 83409159, 532926930}@qq.com, yangelwm@163.com, jinghao.xue@ucl.ac.uk, liaoqm@tsinghua.edu.cn

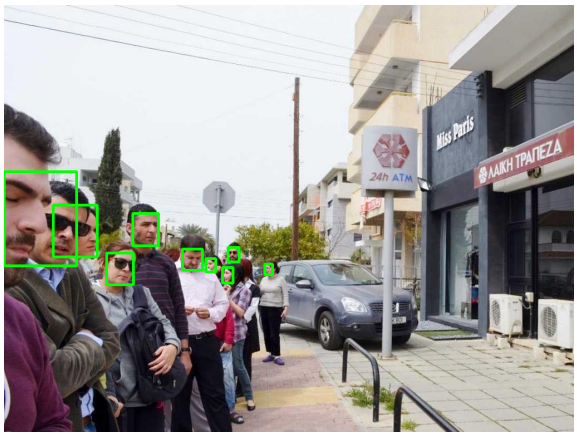
***Abstract*—Face detection is an upstream task of facial image analysis. In many real-world scenarios, we need to detect small, occluded or dense faces that are hard to detect, but hard face detection is a challenging task in particular considering the balance between accuracy and inference speed for real-world applications. This paper proposes an Hourglass Face Detector (HFD) for hard face by developing a deep one-stage fully-convolutional hourglass network, which achieves an excellent balance between accuracy and inference speed. To this end, the HFD firstly shrinks a feature map by a series of stridden convolutional layers rather than pooling layers, so that useful subtle information is preserved better. Secondly, it exploits context information by merging fine-grained shallow feature maps with deep ones full of semantic information, making a better fusion of detailed information and semantic information to achieve a better detection of small faces. Moreover, the HFD exploits prior and multiscale information from the training data to enhance its scale-invariance and adaptability of anchor scales. Compared with the SSH and S$^3$FD methods, the HFD can achieve a better performance in average precision on detecting hard faces as well as a quicker inference. Experiments on the WIDER FACE and FDDB datasets demonstrate the superior performance of our proposed method.**

## I. Introduction

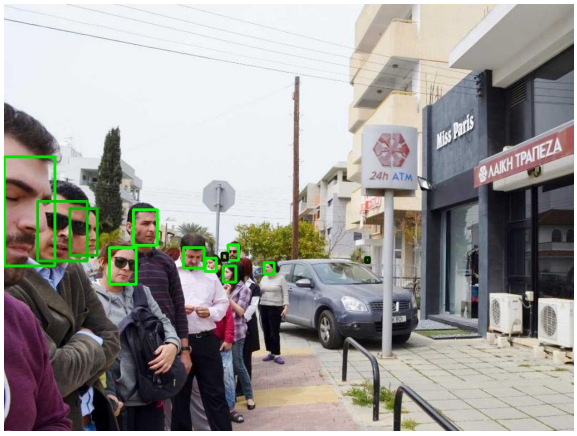
Face detection is fundamental to face recognition [3] [12] [14] [19] [22], face alignment [6] [29] [38] and face clustering [20] [28]. The pioneering work on face detection like Viola-Jones [25] and Deformable Part Model [4] use hand-crafted features and classifiers to detect faces. Recently, by embracing the ideas that led to the remarkable success of CNN-based object detectors, such as Faster-RCNN [18], SSD [11], FPN [9] and YOLO [15] [16] [17], face detection has also achieved impressive further improvement.

However, detecting small, blurred or occluded faces is still a challenging task. The above-mentioned powerful anchor-based general object detection frameworks are now explored to detect hard faces in uncontrolled environment, e.g. on the WIDER FACE benchmark [31] and FDDB benchmark [8]. SSH [13] develops scale-invariant networks based on SSD [11] to detect faces from various layers in a single network. S$^3$FD [35] proposes a new anchor matching strategy to improve the recall rates of small faces. FAN [26] embeds attention mechanisms into the anchor setting in order to gathering context information which is beneficial to occluded faces detection. PyramidBox [23] also exploits context information for hard faces. Among these methods, SSH and S$^3$FD are relatively simple and efficient while others are more complicated and time consuming.

In order to achieve a better balance between efficiency and accuracy, based on SSH and S$^3$FD, we identify several




(a) SSH




(b) HFD

Fig. 1: Detection examples of faces. HFD gets a better result on detecting crowded faces.

*Corresponding Author: Wenming Yang.

aspects of them for potential improvement. Firstly, the pooling-shrinkage strategy may overlook subtle but crucial human-facial information, making it less robust (see Fig. 1 for a small face missed by SSH when it is close to another bigger face). Secondly, as indicated by FPN [9], both bottom-up and top-down structures could benefit object detection, while the backbone of SSH and $S^3$FD is bottom-up only. Thirdly, SSH's large feature map and $S^3$FD's too many detection layers lead to inefficiency of the inference.

Therefore, to address these issues, we develop an hourglass network-based face detector in this paper, termed hourglass face detector (HFD),mainly focusing on the balance of hard face detection accuracy and efficiency. The HFD firstly shrinks a feature map by a series of stridden convolutional layers instead of pooling layers, thus preserves more useful subtle information to make it possible to detect hard faces on a deeper feature map. Then two upsampling layers are used to increase the resolution of feature maps. The scales of feature maps change from large to small and then small to large like an hourglass. The HFD introduces the YOLOv3 [17] darknet backbone to face detection, which is similar to but proved to be more effective than that of FPN. Secondly, to exploit both detailed and semantic information beneficial to small face detection, we propose a simple and effective feature-fusion context module to merge fine-grained shallow feature maps with deep ones full of semantic information. Moreover, to adapt to different scales of input, the HFD is trained with random scales and tested pyramidically.

The main contributions of this paper are three-fold:

- Designing a simple and effective context module merging the fine-grained shallow features with deep ones, learning more discriminative features beneficial to hard face detection.
- Exploiting prior and multi-scale information from the training data to enhance the model's scale-invariance and adaptability of anchor shapes.
- Developing a new hourglass network based face detector for hard face detection, achieving good performance on accuracy and efficiency on the WIDER FACE [31] and FDDB [8] datasets.

## II. Related Work

### A. General Object Detectors Based on CNNs

Deep learning has made tremendous achievements in many computer vision tasks including object detection. A great many of object detectors based on Convolutional Neural Networks (CNNs) have been designed and achieved great improvements in both accuracy and speed. These CNN-based methods can be devided into one-stage methods and two-stage methods. For a single-stage detector, the detection head only makes prediction for one time, and the classification and regression results are directly obtained by doing convolution on the feature maps. For example, YOLO [15] [16] [17] divides the image into a series of grids, and directly predicts the category and bounding box of the object in each grid. RetinaNet [10] uses a light-weight single-stage detection head. The classification and regression branches use four cascaded convolutional layers for feature extraction, and then predicts the categories and offsets based on the anchors at each point to obtain the final detection results. The detection head structure of FCOS [39] is similar to that of RetinaNet. The only difference is that FCOS has no anchors and adopts a new rule for selecting positive and negative samples. The detection head of the bottom-up detection algorithms represented by CornerNet [40] and CenterNet [41] output the heatmap and embeddings of some representative points, and then combine several points into a bounding box.

The detection head of a multi-stage detector is more complicated. The detection head of the classic anchor-based two-stage detector represented by Faster R-CNN [18] and Mask R-CNN [42] includes a region proposal generation step and a proposal refinement step. In the first stage, the detection head performs binary classification and regression on all anchors, and selects out anchors that are more likely to contain objects and refines their positions. In the second stage, the detection head outputs classification and regression results of the candidate box selected in the first stage to obtain the category and final position of the object. Recently, many methods adding a refinement stage based on detection results predicted by a single-stage anchor-free detection framework have been proposed. For example, RepPoints [43] uses 9 points to describe a bounding box, and its detection head uses deformable convolution to refine the positions of 9 points twice. The detection heads of BorderDet [44] and VarifocalNet [45] both add some additional processing steps after the FCOS detection head, and make a further feature extraction based on the coarse detection results of FCOS, then make use of the enhanced features for refine prediction. Compared with single-stage detection methods, two-stage detection methods achieve better detection accuracy, but the processing time will also increase. Now more attention has been paid to designing object detectors which can achieve a better speed-accuracy balance.In practical applications, it is necessary to select a suitable detector according to the specific scene.

### B. Face Detectors Based on CNNs

The early CNN-based face detectors usually adopt a coarse-to-fine strategy [33] and form a CNN in a cascade way. In recent years, general object detection has achieved significant progress.These new frameworks including one-stage and two-stage have inspired many face detection methods.

FaceBoxes [34] proposes a CPU real-time face detector based on SSD. SSH [13] and $S^3$FD [35] are scale-invariant face detectors. SSH uses feature maps of shallow layers to detect small faces while $S^3$FD introduces some specific strategies to improve the recall rate of small faces. FAN [26] embeds attention mechanisms into the anchor setting and it is based on RetinaNet [10]. PyramidBox [23] proposes a novel context-assisted single shot face detector to handle hard face detection. DFS [24] introduces a more effective feature fusion pyramid and a more efficient segmentation branch for hard

face detection. All these methods mentioned above are single stage face detectors.

In contrast, CMS-RCNN [37], Face R-FCN [27] and FDNet [32] are two-stage face detectors based on Faster-RCNN [18] and R-FCN [1]. CMS-RCNN [37] improves the performance of hard face detection by making use of context information. Face R-FCN [27] replaces global average pooling by position-sensitive average pooling to re-weight the different regions of the face. FDNet [32] designs a light-head Faster RCNN and introduces some specific training and testing strategies into face detection.

Generally speaking, two stage face detectors achieve higher detection accuracy but are not as efficient and fast as single-stage ones. How to achieve a relatively better balance between speed and accuracy still remains an open problem.

## III. Methods

In this section,we first illustrate the hourglass backbone of HFD which contains both bottom-up and top-down pathways. It is superior to SSH and S$^3$FD whose backbones are only bottom-up. Then, a simple and effective context module is added before the third detection layer, for better detecting hard faces.

### A. Network architecture

Fig. 2 shows the sketched general architecture of the proposed HFD. It is a Res-based fully convolutional network which localizes faces on three detection layers. Feature maps for detection are extracted by continuous conv-bn-leaky ReLU (CBL) modules, residual blocks and upsampling layers, which form the bottom-up and top-down structures looking like an hourglass. The bottom-up path contains a series of feature maps of five different scales with a scale step of 1/2. After the last layer of each stage, we use a CBL layer with kernel size of 3 and stride of 2 instead of a pooling layer to reduce the size of the feature maps by half, which can preserve more detailed information that is conducive to small face detection [21]. We denote the last feature map of each stage from shallow to deep as $\{C_1, C_2, C_3, C_4, C_5\}$. $C_5$ is the deepest feature map whose size is reduced to 1/32 of the input size while the number of channels increases to 1024, which will be input into the first detection layer as $E_1$ after being processed by five CBL layers.

The following two detection layers are formed in a similar way, and a top-down pathway is proposed to combine the feature maps of different scales, in order to take full advantages of the detailed context information in the shallow layers along with the deep robust features full of semantic information for better detection. We merge $E_1$ and $C_4$ by our proposed FFC module to get a more representative feature for the second detection layer. For the last detection layer which is mainly responsible for small face detection, more discriminative feature which contains both detailed and semantic information is needed. Therefore, we fuse two low-level feature maps $C_2$ and $C_3$ by the FFC module firstly to get feature map $F_1$ which is full of detailed context information. Then we merge $F_1$ with $E_2$ which contains robust semantic information. After a series of CBL layers, we finally obtain $E_3$ which contains both detailed and semantic information as the input of the last detection layer. This hourglass architecture is more powerful than that of SSH and S$^3$FD for detecting small faces, while the latter two methods only adopt pooling-extracted features which are not sufficiently facial representative.

The details of our proposed feature-fusion context module (FFC, section III-B) will be introduced in the next section. In addition, to further enhance the detection performance, two other components are added into this general architecture: 1) a spacial pyramid pooling structure borrowed from SPP-Net [5], to help distinguish faces from non-faces; and 2) a multi-scale training as suggested by YOLO and a multi-scale testing with a pyramid, to enhance scale-invariance of the proposed HFD.

### B. Feature Fusion Context (FFC) Module

A shallow layer contains more detailed context information but is not discriminative, while a deep layer is classifiable and full of semantic information but lacks context information. Merging deep layers with shallow ones is beneficial for detecting small objects, as FPN and YOLOv3 already demonstrate. However, questions remain about how to merge effectively and efficiently, particularly as different feature maps from the network always differ in scales and channels.

To overcome this problem, we propose a simple feature-fusion context (FFC) module (Fig. 3). The FFC module can effectively fuse two feature maps of different scales and numbers. The high-level feature maps to be fused are always smaller in scale; we assume that their scale are $1/a$ of the scale of low-level feature maps. We use a $1 \times 1$ convolution layer to transform the channels of deep features to be consistent with the channels of shallow ones, and then we upsample the transformed deep features by factor of $a$ to make the resolution of two fused feature maps to be consistent. After that, we concatenate these two feature maps by channel and use another $1 \times 1$ convolution layer to balance the weights of the two feature maps.

Our FFC module differs from the YOLOv3 backbone in two ways: 1) shallower layers with better fine-grained information are used, and richer features from three different scales are combined; and 2) in each FFC module, a $1 \times 1$ convolutional layer is applied to balance the weights from the shallow map and the deep one. Our proposed feature fusion context module obtains a 1.4% AP improvement than immediately merging the medium one with the deep one, this proves its effectiveness.

## IV. Experiments

### A. Datasets

The WIDER FACE dataset contains 32,203 images with 393,703 annotated faces with a high degree of variability in scale pose and occlusion. 40%, 10% and 50% of the data are randomly selected as training, validation and test sets. The validation and test sets are divided into “easy”, “medium” and “hard” subsets. We train all models on the training set of the WIDER FACE dataset and evaluate on the WIDER

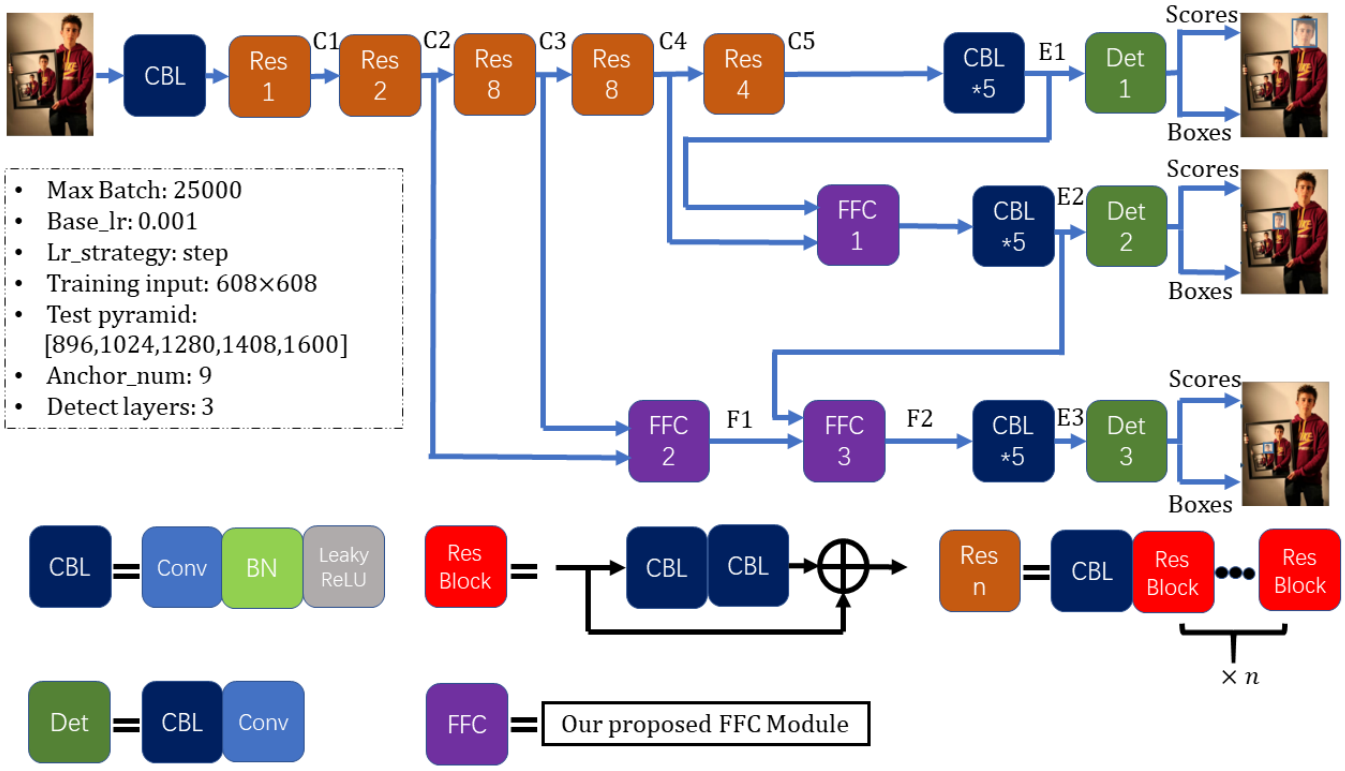



Fig. 2: Network architecture of the proposed hourglass face detector (HFD).

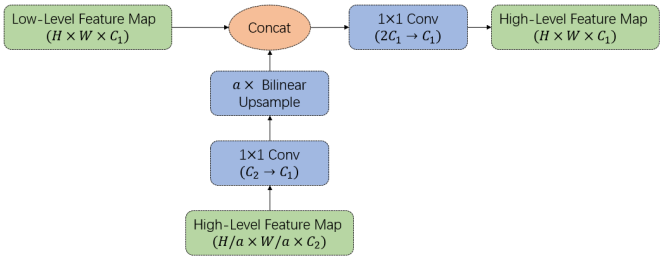



Fig. 3: The proposed feature-fusion context module.

FACE validation set. We mainly focus on "hard" subset which contains all validation images including faces hard to detect. For evaluation, the standard average precision (AP) is used.

The FDDB dataset is a dataset of face regions designed for studying the problem of unconstrained face detection. It has 2,845 images with 5,171 annotated faces. Faces in FDDB are represented by ellipses instead of rectangle boxes. We use our model trained on the WIDER FACE dataset to directly test on the FDDB dataset. On the FDDB dataset, we use the True Positive Rate (TPR) at the false positives (FP) equals to 2000 to evaluate the performance of the methods.

### B. Experimental Setup

Each input image is resized such that its longer side is 608. Our networks are fine-tuned for 25K iterations starting from a pre-trained ImageNet classification network. We use mini-batch of 4 and batch-size of 64 and synchronized SGD is used to train the model on 4GPUs. The learning rate is initially set to 0.001 and drops by a factor of 10 after 18K iterations, and in the early 4K iterations we use the burn-in strategy. We set momentum to 0.9 and weight decay to 0.0005.

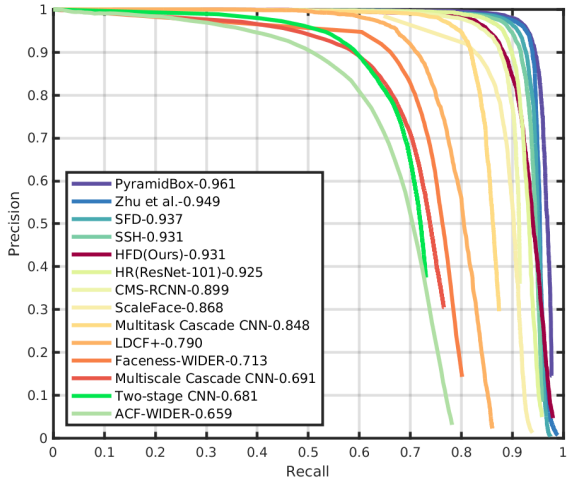
Anchors matching the ground truth best are assigned to positive, with one ground truth one anchor. Anchors with IoU>0.7 but not the best overlapped with any ground-truth faces are ignored, and the remaining anchors are assigned to negative. For anchor generation, we adapt anchor scales to the distribution of face scales. Anchor aspect ratios are set to 1 as faces are nearly square.

During inference, each detection layer outputs all the transformed anchors with scores larger than 0.01 as detection, and a non-maximum suppression (NMS) with a threshold of 0.4 is performed on the outputs of all detection layers.

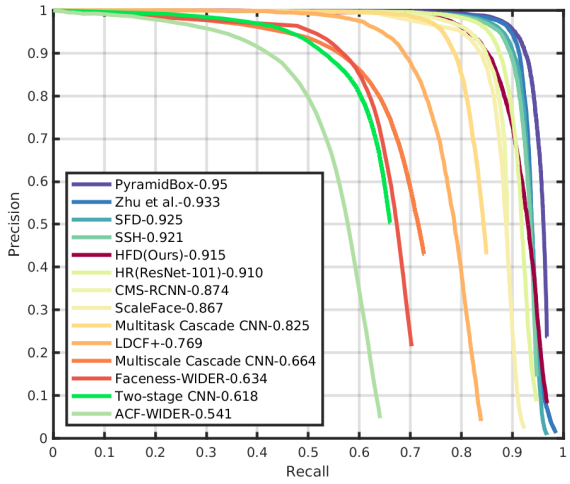
### C. Results on WIDER FACE

On the WIDER FACE validation set, we compare the proposed HFD with some popular face detectors, including HR [7], CMS-RCNN [37], Multitask Cascade CNN (MTCNN) [33], Faceness [30], SSH [13], $S^3$FD [35], PyramidBox [23] and Zhu et al. [36].

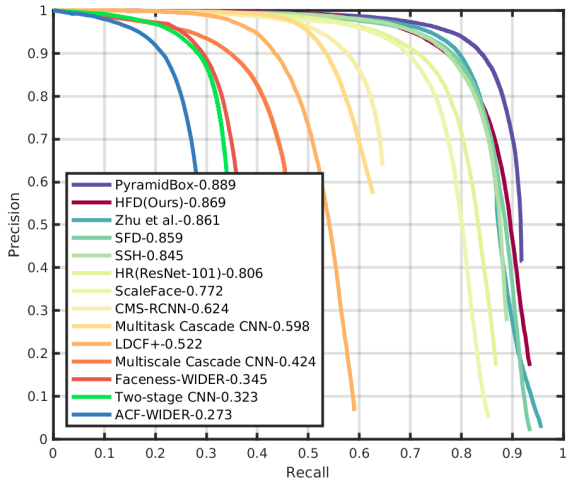
As shown in Fig. 4, our proposed HFD (plotted with dark red line) achieves good performance on the validation set, *i.e.* 93.1% (Easy), 91.5% (Medium) and 86.9% (Hard). Experimental results demonstrate that our HFD outperforms many SOTA face detectors on the "hard" set, outperforming SSH by about 2.4% and HR by more than 6% in terms of AP




(a) Easy




(b) Medium




(c) Hard

Fig. 4: Precision-recall curves and average precision (AP) of the compared methods on the whole WIDER validation dataset. HFD is plotted with dark red line.

while trained with smaller size than them. HFD also achieves a comparable result with SSH and S$^3$FD on the "easy" and "medium" sets. HFD has a lower accuracy than PyramidBox but is more efficient than the latter as shown in Table III. Therefore, HFD not only shows excellent performance in hard face detection, but also is deemed to having high efficiency and application potential. More discussion about the performance of the HFD on the WIDER FACE dataset will be shown in the next section (section V).

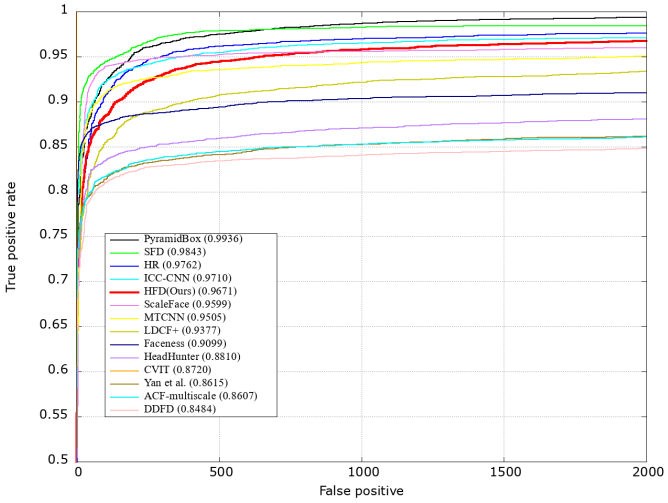



Fig. 5: Receiver operating characteristic (ROC) curves with discrete scores on the FDDB dataset. The number in the legend is the true positive rate (TPR) at the false positives (FP) equals to 2000. HFD is plotted with red line.

### D. Results on FDDB

We use our model trained on the WIDER FACE dataset to directly test on the FDDB dataset and compare our HFD with other SOTA methods including S$^3$FD [35], PyramidBox [23] and Zhu et al. [36] as shown in Fig. 5. From Fig. 5 we can observe that our HFD has achieved 96.71% TPR (FP=2000) on the FDDB dataset. Because the faces in the FDDB dataset are similar to those in the "easy" and "medium" subsets of the WIDER FACE dataset, that is, the size of the faces is relatively large, and there are relatively few faces in each image. Therefore, we can come to the conclusion that our method achieves comparable results with other SOTA methods on simple face detection.

## V. Discussion

### A. Ablation study

Results of our ablation study are listed in Table I. The model that all five modules or strategies are not used is set as the baseline. We study effects of five strategies, one by one as detailed below:

- "Prior" indicates the strategy to take the properties of the training data to adapt anchor scales. It produces an increase of 1.5% in AP from the baseline;

(a) HFD (b) SSH

Fig. 6: Detection results of the HFD and the SSH in different scenarios.

TABLE I: Ablation Study Results on WIDER FACE Validation "Hard" Set.

| Modules and Strategies | | | | | AP |
|---|---|---|---|---|---|
| **Prior** | **SPP** | **#Anchor** | **Context** | **Pyramid** | |
| | | | | | 0.807 |
| ✓ | | | | | 0.822 |
| ✓ | ✓ | | | | 0.825 |
| ✓ | ✓ | ✓ | | | 0.825 |
| ✓ | ✓ | ✓ | ✓ | | 0.839 |
| ✓ | ✓ | ✓ | ✓ | ✓ | **0.869** |

- "SPP" is the trick to adopt a spacial pyramid pooling for feature extraction before the three detection modules. A pooling field of [1×1, 5×5, 9×9, 13×13] is used to form four feature maps, which are then concatenated to form the SPP feature. It provides a slight increase of 0.3% in AP;
- "#Anchor" is the trick of trialing denser or coarser anchors. We find no difference between 2, 3 or 4 anchors per detection layer. Perhaps with more anchors, it will match better with the ground truth, but it may also increase the negatives;

- "Context" corresponds to the context module proposed in section III-B, which improves the AP scores by 1.4%;
- "Pyramid" means that we test the WIDER val dataset with multiscale input. This boosts AP by 3.0%.

### *B. Time complexity*

TABLE II: Inference time and AP of HFD and SSH on WIDER FACE Validation "Hard" Set.

| Models | Max Size | Time (s) | AP |
|---|---|---|---|
| **HFD** | 896 | **172.4** | 0.839 |
| | 1024 | 208.8 | 0.846 |
| | 1216 | 285.2 | **0.853** |
| **SSH** | 1000 | 267.8 | 0.790 |
| | 1200 | 374.2 | 0.806 |
| | 1600 | 571.0 | 0.814 |

TABLE III: Inference Time Comparison on WIDER FACE Validation "Hard" Set.

| Method | HFD | SSH | $S^3$FD | PyramidBox |
|---|---|---|---|---|
| Time (s) | **172.4** | 201.3 | 253.9 | 369.2 |

Our proposed HFD is more efficient than SSH: inference time of HFD and SSH on the WIDER FACE validation dataset by using an NVIDIA 1080 Ti GPU is listed in Table II. For a fair comparison, we evaluate both methods on PyTorch 0.4.1. The time is accumulated for all 3,226 pictures of the WIDER FACE validation dataset. HFD achieves the reported detection performance (83.9% in AP) in 172s with the longest input side of 896, while SSH achieves the best performance (81.4% in AP) in 571s with that of 1600. That is, HFD outperforms SSH on the WIDER FACE validation "hard" subset by 2.5% in AP with the inference time reduced by about 70%.

To compare the inference speed with other methods, we test the inference time of different methods. All experiments are implemented on an NVIDIA 1080 Ti GPU with CUDA 9.0 and cuDNN v7.0. The maximum sizes of test pictures are resized to 896. The total time spent on inferring all 3,226 pictures of the WIDER FACE validation dataset is used to compare the speed. As indicated in Table III, our HFD has superiority in inference speed compared with other algorithms.

### *C. Qualitative Comparison*

Fig. 6 illustrates the detection results using the HFD and SSH methods in different scenarios. In the first crowded demonstration scenario, there are a lot of small dense faces. We can find that HFD has a very obvious advantage in detecting small faces in the back rows. In the second marching scenario, profile and occlusion are the main obstacles to face detection. Our HFD basically makes a correct detection for all the faces while there are many false positives in the detection result of the SSH. In the last conference scenario, the image has a low quality and the faces are blurred. The HFD is not interfered by the low image quality while the SSH makes some wrong predicitions. In conclusion, compared with the SSH method, our method has significant advantages in detecting dense, small, profile, occluded and blurred faces, which proves that our method has superiority on hard face detection.

## VI. Conclusion

In this paper, we proposed a one-stage hourglass network-based hard face detector HFD. The HFD's excellent performance is attributable to three main technical innovations adopted in this work: the HFD uses stridden convolutional layers rather than pooling layers in order to preserve useful subtle information for hard faces; to supply more detailed and semantic information for better detecting small faces, the HFD merges fine-grained shallow and deep feature maps by the FFC module; to enhance the scale-invariance and adaptability, the HFD exploits both prior and multiscale information from the training data. Compared with the SSH face detector, our method improves the AP on WIDER FACE "hard" set by about 2.4% and reduces the inference time by 70%. Compared with other SOTA methods, HFD also achieves good performance on the challenging WIDER FACE and FDDB datasets. The advantages of HFD in inference speed and hard face detection bring great potential for practical application.

## VII. Acknowledgment

This work was supported by the Natural Science Foundation of Guangdong Province(No.2020A1515010711) and the Special Foundation for the Development of Strategic Emerging Industries of Shenzhen(Nos. JCYJ20170818161845824, JCYJ20200109143010272 and JCYJ20200109143035495).

## References

[1] J. Dai, Y. Li, K. He and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," Advances in neural information processing systems, pp. 379–387, 2016.

[2] A. Dapogny and K. Bailly, "Face alignment with cascaded semi-parametric deep greedy neural forests," Pattern Recognition Letters, vol. 102, pp. 75–81, 2018.

[3] J. Deng, J. Guo, N. Xue and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in CVPR, 2019, pp. 4690–4699.

[4] P. Felzenszwalb, R. Girshick, D. McAllester and D. Ramanan, "Object detection with discriminatively trained part-based models," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, pp. 1627–1645, 2010.

[5] K. He, X. Zhang, S. Ren and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, pp. 1904–1916, 2015.

[6] Q. Hou, J. Wang, R. Bai, S. Zhou and Y. Gong, "Face alignment recurrent network," Pattern Recognition, vol. 74, pp. 448–458, 2018.

[7] P. Hu and D. Ramanan, "Finding tiny faces," in CVPR, 2017, pp. 1522–1530.

[8] V. Jain and E. Learned-Miller, "Fddb: A benchmark for face detection in unconstrained settings," University of Massachusetts, Amherst, Tech. Rep., 2010.

[9] T. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan and S. Belongie, "Feature pyramid networks for object detection," in CVPR, 2017, pp. 2117–2125.

[10] T. Lin, P. Goyal, R. Girshick, K. He and P. Dollar, "Focal loss for dense object detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, pp. 318–327, 2020.

[11] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu and A. Berg, "SSD: Single shot multibox detector," in ECCV, 2016, pp. 21–37.

[12] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj and L. Song, "SphereFace: Deep hypersphere embedding for face recognition," in CVPR, 2017, pp. 6738–6746.

[13] M. Najibi, P. Samangouei, R. Chellappa and L. Davis, "SSH: Single stage headless face detector," in ICCV, 2017, pp. 4875–4884.

[14] W. Ou, X. Luan, J. Gou, Q. Zhou, W. Xiao, X. Xiong and W. Zeng, "Robust discriminative nonnegative dictionary learning for occluded face recognition," Pattern Recognition Letters, vol. 107, pp. 41–49, 2018.

[15] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: Unified, real-time object detection," in CVPR, 2016, pp. 779–788.

[16] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in CVPR, 2017, pp. 7263–7271.

[17] J. Redmon and A. Farhadi, 2018. "YOLOv3: An incremental improvement," 2018, abs/1804.02767.

[18] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in NIPS, 2015, pp. 91–99.

[19] F. Schroff, D. Kalenichenko and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in CVPR, 2015, pp. 815–823.

[20] X. Shi, Z. Guo, F. Xing, J. Cai and L. Yang, "Self-learning for face clustering," Pattern Recognition, vol. 79, pp. 279–289, 2018.

[21] J. Springenberg, A. Dosovitskiy, T. Brox and M. Riedmiller, "Striving for simplicity: The all convolutional net," 2014, abs/1412.6806.

[22] Y. Sun, X. Wang and X. Tang, "Deep learning face representation from predicting 10,000 classes," in CVPR, 2014, pp. 1891–1898.

[23] X. Tang, D. Du, Z. He and J. Liu, "PyramidBox: A context-assisted single shot face detector," in ECCV, 2018, pp. 797–813.

[24] W. Tian, Z. Wang, H. Shen, W. Deng, Y. Meng, B. Chen, X. Zhang, Y. Zhao and X. Huang, "Learning better features for face detection with feature fusion and segmentation supervision," 2018, abs/1811.08557.

[25] P. Viola and M. Jones, "Robust real-time face detection," International Journal of Computer Vision, vol. 57, pp. 137–154, 2004.

[26] J. Wang, Y. Yuan and G. Yu, "Face attention network: An effective face detector for the occluded faces," 2017, abs/1711.07246.

[27] Y. Wang, X. Ji, Z. Zhou, H. Wang and Z. Li, "Detecting faces using region-based fully convolutional networks," 2017, abs/1709.05256.

[28] B. Wu, Y. Zhang, B. Hu and Q. Ji, "Constrained clustering and its application to face clustering in videos," in CVPR, 2013, pp. 3507–3514.

[29] X. Xiong and F. De la Torre, "Supervised descent method and its applications to face alignment," in CVPR, 2013, pp. 532–539.

[30] S. Yang, P. Luo, C. Loy and X. Tang, "From facial parts responses to face detection: A deep learning approach," in ICCV, 2015, pp. 3676–3684.

[31] S. Yang, P. Luo, C. Loy and X. Tang, "WIDER FACE: A face detection benchmark," in CVPR, 2016, pp. 5525–5533.

[32] C. Zhang, X. Xu and D. Tu, "Face detection using improved Faster RCNN," 2018, abs/1802.02142.

[33] K. Zhang, Z. Zhang, Z. Li and Q. Yu, "Joint face detection and alignment using multitask cascaded convolutional networks," IEEE Signal Processing Letters, vol. 23, pp. 1499–1503, 2016.

[34] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang and S. Li, "FaceBoxes: A CPU real-time face detector with high accuracy," IEEE International Joint Conference on Biometrics, pp. 1–9, 2017.

[35] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang and S. Li "S3FD: Single shot scale-invariant face detector," in ICCV, 2017, pp. 192–201.

[36] C. Zhu, R. Tao, K. Luu and M. Savvides, "Seeing small faces from robust anchor's perspective," in CVPR, 2018, pp. 5127–5136.

[37] C. Zhu, Y. Zheng, K. Luu and M. Savvides, "CMSRCNN: contextual multi-scale region-based CNN for unconstrained face detection," Deep Learning for Biometrics, pp. 57–79, 2017.

[38] X. Zhu, Z. Lei, X. Liu, H. Shi and S. Li, "Face alignment across large poses: A 3D solution," in CVPR, 2016, pp. 146–155.

[39] Z. Tian, C. Shen, H. Chen and T. He, " Fcos: Fully convolutional one-stage object detection," in CVPR, 2019, pp. 9627-9636.

[40] H. Law and J. Deng, "Cornernet: Detecting objects as paired keypoints," in ECCV, 2018, pp. 734-750.

[41] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang and Q.Tian, "Centernet: Keypoint triplets for object detection," in ICCV, 2019, pp.6569-6578.

[42] K. He, G. Gkioxari, P. Dollár and R.Girshick. "Mask r-cnn," in ICCV, 2017, pp.2961-2969.

[43] Z. Yang, S. Liu, H. Hu, L. Wang and S. Lin, "Reppoints: Point set representation for object detection," in ICCV, 2019, pp. 9657-9666.

[44] H. Qiu, Y. Ma, Z. Li, S. Liu and J. Sun, "Borderdet: Border feature for dense object detection," in ECCV, pp. 549-564.

[45] H. Zhang, Y. Wang, F. Dayoub and N.Sünderhauf, "Varifocalnet: An iou-aware dense object detector," 2020, abs/2008.13367.