# ProgressFace: Scale-Aware Progressive Learning for Face Detection
# Supplementary Material

Jiashu Zhu, Dong Li, Tiantian Han, Lu Tian, and Yi Shan

Xilinx Inc., Beijing, China
{jiashuz, dongl, hantian, lutian, yishan}@xilinx.com

## 1 Overview

In this supplementary material, we present six sets of additional results.

1. We analyze the effect of hyper-parameter $T$ in the anchor-free enhancement module.
2. We discuss the effect of uncertainty estimation in the different stages of progressive training.
3. We compare the inference time between our method with the state-of-the-art approaches.
4. We describe the differences between our anchor-free module and SFace [3].
5. We show discrete and continuous ROC curves on the FDDB dataset and detailed precision-recall curves on the WIDER FACE *test* sets.
6. We show additional qualitative results of our method on the FDDB and WIDER FACE datasets.

## 2 Parameter Analysis for Anchor-Free Module

We evaluate the effect of hyper-parameter $T$ in the anchor-free enhancement module. Table 1 shows the results on the WIDER FACE *validation* set with different thresholds. We find that $T = 0.7$ works best and hence we choose this value for all of our experiments.

## 3 Effect of Uncertainty Estimation

We analyze the effect of uncertainty estimation with KL loss during different stages of progressive training. As shown in Table 2, applying KL loss in the entire training process can achieve the best AP performance. The results show uncertainty estimation works for faces with different scales. Such scheme improves bounding box regression and helps learn more accurate face detectors in our method.

## 4 Inference Time

We measure the inference time of our ProgressFace and the state-of-the-art face detectors in Table 3. Compared to the RetinaFace with DCNv2, our ProgressFace-Light costs similar inference time (6.92 ms vs. 6.80 ms) but achieves better performance on WIDER FACE *validation* set (e.g., 87.9% vs. 79.5% for the hard set).

## 5 Differences between our anchor-free module and SFace

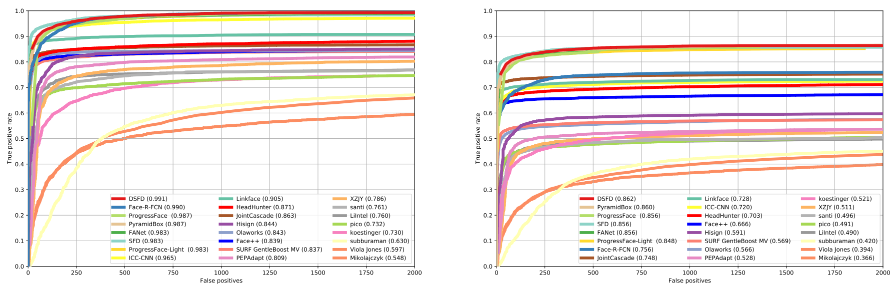
The main differences between our anchor-free branch and SFace [3] are three-fold. (1) The strategies of integrating anchor-free branches are different. SFace needs a re-scoring strategy to merge the anchor-based and anchor-free branches to unify the confidence scores of these two branches. Differently, we take the output of the anchor-free branch as complementary positive anchors for the anchor-based branch and thus bypass the discrepancy of confidence scores generated by these two branches. (2) The inference procedures are different. SFace needs the anchor-free branch for inference as it is coupled with the anchor-based branch. In contrast, our anchor-free enhancement module will be removed for inference and no extra computation cost will be introduced. (3) The optimization targets are different. SFace relies on UnitBox [4] to infer a pixel that falls in a target face or not with cross-entropy loss and regress four bounding box offsets by IoU loss for each positive pixel. Our anchor-free branch is inspired by CenterNet [5], which predicts centers of faces with focal loss and regresses size and offset with L1 Loss for each center.

## 6 Detailed Performance Curves

We present the detailed ROC curves on the FDDB benchmark with discrete and continuous metrics in Fig. 1. The results of previous methods are provided by the official website of the FDDB benchmark. Our ProgressFace achieves 98.7% and 85.6% TPR with respect to discrete and continuous metrics, which is competitive with state-of-the-art methods. It is notable that our ProgressFace-Light still can achieve comparable performance with much less computation cost. We also present detailed precision-recall curves on the WIDER FACE *test* sets in Fig. 2

## 7 Additional Qualitative Results

For qualitative evaluations, we show in Fig. 3 that our method can detect 1037 faces in the image containing dense faces with extremely small scales. We also present additional face detection results on the FDDB and WIDER FACE datasets in Fig. 4, 5, 6. These results show that our method can detect faces accurately in a wide variety of scale, pose, blur, expression, occlusion and illumination.




(a) Discrete ROC curves (b) Continuous ROC curves

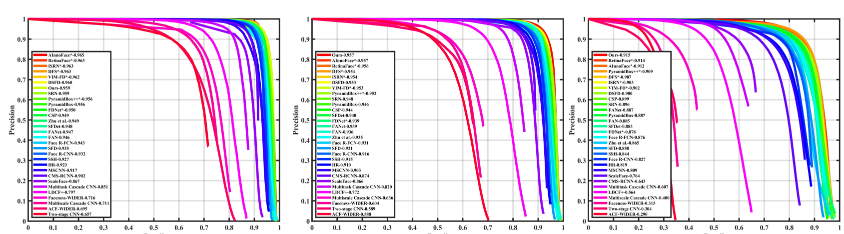
**Fig. 1.** Detailed ROC curves of previous methods and ours on the FDDB benchmark with discrete and continuous metrics. The numbers indicate the true positive rate when the amount of false positives is equal to 1,000.

**Table 1.** Effect of different thresholds $T$ in the anchor-free enhancement module on the WIDER FACE *validation* set.

| $T$ | *Easy* | *Medium* | *Hard* |
|---|---|---|---|
| 0.5 | 0.946 | 0.931 | 0.871 |
| 0.6 | **0.949** | 0.932 | 0.874 |
| 0.7 | **0.949** | **0.935** | **0.879** |
| 0.8 | 0.949 | 0.933 | 0.877 |

**Table 2.** Effect of uncertainty estimation with KL loss. We compare the performance when the KL loss is applied on the different training stages.

| When to apply KL loss | *Easy* | *Medium* | *Hard* |
|---|---|---|---|
| 1st stage | **0.949** | **0.935** | **0.879** |
| 2nd stage | **0.949** | 0.934 | 0.878 |
| 3rd stage | 0.948 | 0.934 | 0.877 |
| 4th stage | 0.947 | 0.933 | 0.877 |




(d) Test:Easy (e) Test:Medium (f) Test:Hard

**Fig. 2.** Precision-recall curves on the WIDER FACE *test* sets. * indicates the work which is under review or not formally published.

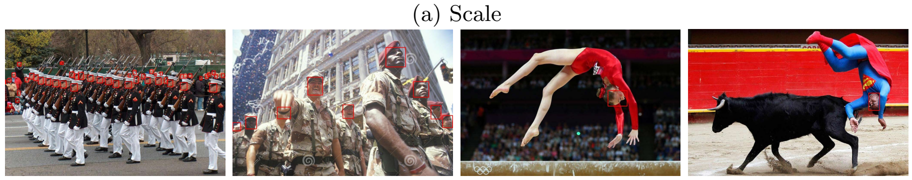
**Table 3.** Inference time of our ProgressFace and the state-of-the-art face detectors. For fair comparisons, inference time are computed with the same $640 \times 480$ input size for all methods. Platform info: NVIDIA Tesla P100, CUDA 9.0, CUDNN 7.0.5, MXNet 1.3.0.

| Methods | *Easy* | *Medium* | *Hard* | Inference time (ms) |
|---|---|---|---|---|
| LFFD v2 [2] | 0.837 | 0.835 | 0.729 | 7.11 |
| LFFD v1 [2] | 0.910 | 0.881 | 0.780 | 9.64 |
| RetinaFace (MobileNet-0.25) [1] | 0.914 | 0.901 | 0.782 | 5.21 |
| RetinaFace (MobileNet-0.25) [1] + DCNv2 [6] | 0.922 | 0.910 | 0.795 | 6.80 |
| ProgressFace-Light (MobileNet-0.25) | 0.949 | 0.935 | 0.879 | 6.92 |
| RetinaFace (ResNet-152) [1] | 0.969 | 0.961 | 0.918 | 72.65 |
| ProgressFace (ResNet-152) | 0.968 | 0.962 | 0.918 | 72.75 |

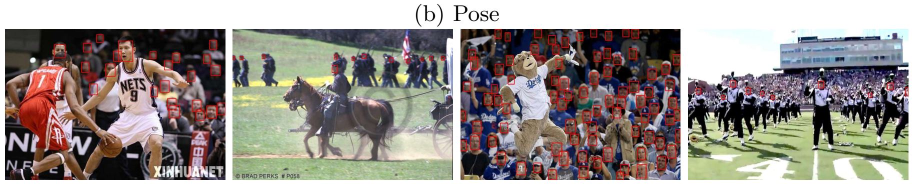

**Fig. 3.** Our method can detect 1037 faces out of the reported 1151 faces in the image containing dense faces with extremely small scales. We also list the amount of detected faces by previous methods.


**Fig. 4.** Sample detection results of our mehtod on the FDDB dataset.

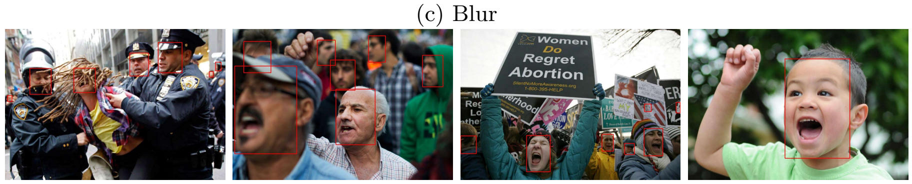

(a) Scale



(b) Pose




(c) Blur



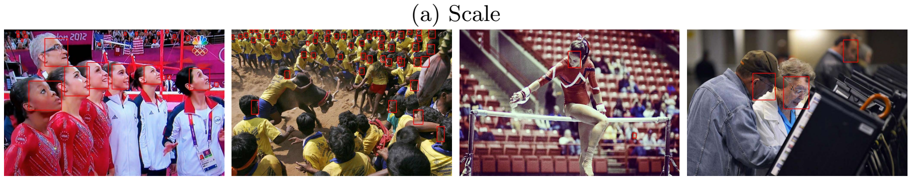

(d) Expression


(e) Occlusion

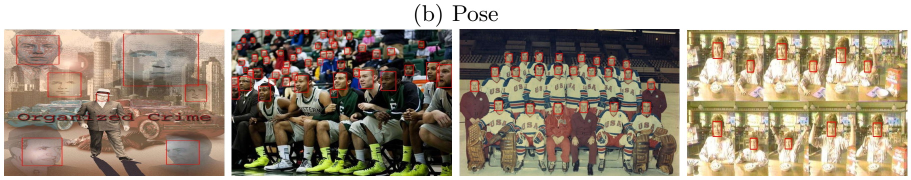
(f) Illumination

**Fig. 5.** Sample detection results on the WIDER FACE *validation* set. Our method can detect faces accurately with large variations in scale, pose, blur, expression, occlusion and illumination.


(a) Scale

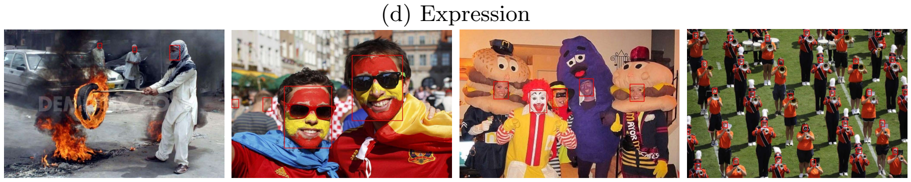


(b) Pose




(c) Blur


(d) Expression



(e) Occlusion

(f) Illumination

**Fig. 6.** Sample detection results on the WIDER FACE *validation* set. Our method can detect faces accurately with large variations in scale, pose, blur, expression, occlusion and illumination.

## References

1. Deng, J., Guo, J., Zhou, Y., Yu, J., Kotsia, I., Zafeiriou, S.: Retinaface: Single-stage dense face localisation in the wild. arXiv preprint arXiv:1905.00641 (2019) 4
2. He, Y., Xu, D., Wu, L., Jian, M., Xiang, S., Pan, C.: Lffd: A light and fast face detector for edge devices. arXiv preprint arXiv:1904.10633 (2019) 4
3. Wang, J., Yuan, Y., Li, B., Yu, G., Jian, S.: Sface: An efficient network for face detection in large scale variations. arXiv preprint arXiv:1804.06559 (2018) 1, 2
4. Yu, J., Jiang, Y., Wang, Z., Cao, Z., Huang, T.: Unitbox: An advanced object detection network. In: ACMMM (2016) 2
5. Zhou, X., Wang, D., Krähenbühl, P.: Objects as points. arXiv preprint arXiv:1904.07850 (2019) 2
6. Zhu, X., Hu, H., Lin, S., Dai, J.: Deformable convnets v2: More deformable, better results. In: CVPR (2019) 4