ProgressFace: Scale-Aware Progressive Learning for Face Detection

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

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

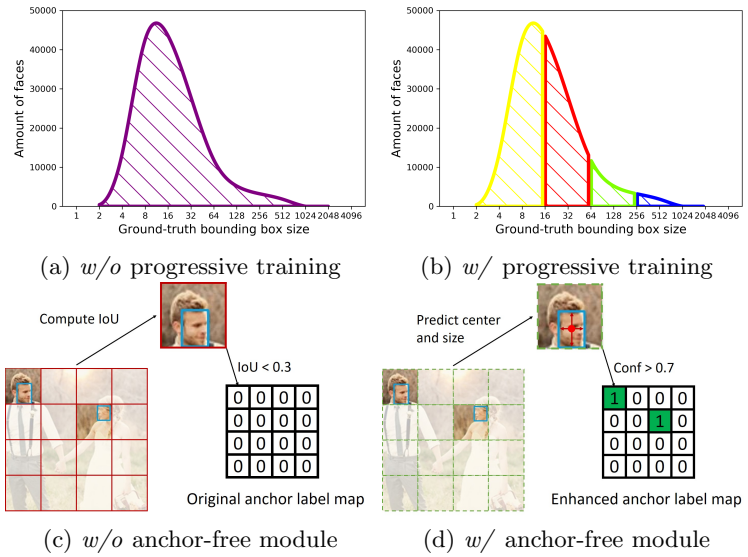
Abstract. Scale variation stands out as one of key challenges in face detection. Recent attempts have been made to cope with this issue by incorporating image / feature pyramids or adjusting anchor sampling / matching strategies. In this work, we propose a novel scale-aware progressive training mechanism to address large scale variations across faces. Inspired by curriculum learning, our method gradually learns large-to-small face instances. The preceding models learned with easier samples (i.e., large faces) can provide good initialization for succeeding learning with harder samples (i.e., small faces), ultimately deriving a better optimum of face detectors. Moreover, we propose an auxiliary anchor-free enhancement module to facilitate the learning of small faces by supplying positive anchors that may be not covered according to the criterion of IoU overlap. Such anchor-free module will be removed during inference and hence no extra computation cost is introduced. Extensive experimental results demonstrate the superiority of our method compared to the state-of-the-arts on the standard FDDB and WIDER FACE benchmarks. Especially, our ProgressFace-Light with MobileNet-0.25 backbone achieves 87.9% AP on the hard set of WIDER FACE, surpassing largely RetinaFace with the same backbone by 9.7%. Code and our trained face detection models are available at https://github.com/jiashu-zhu/ProgressFace.

Keywords: Face detection, progressive learning, anchor-free methods

1 Introduction

Face detection is an important task in computer vision with extensive subsequent research fields (e.g., face recognition and face tracking) and practical applications including intelligent surveillance for smart city and face unlock / beautification in smartphones. Owing to the great development of convolutional neural networks (CNNs), deep face detectors have achieved outstanding performance compared to the conventional hand-crafted features and classifiers. Typical methods include two-stage and one-stage anchor-based detectors. The predominant two-stage methods [37] first generate a set of candidate region proposals and then refine them for final detection. One-stage detectors [30] aim to directly classify and regress the pre-defined anchors without the extra proposal generation step.

Face detection, acting as a special case of object detection, has inherited effective techniques from generic detection methods but still suffers from large



(a) *w/o* progressive training (b) *w/* progressive training

(c) *w/o* anchor-free module (d) *w/* anchor-free module

Fig. 1. Illustration of our motivations. With progressive learning, we train faces with different scales in a large-to-small order instead of feeding them into network at the same time. In (b), the different colors mean the groups of face instances with different sizes. Blue represents the faces with largest sizes, green represents the second largest, and so on. With anchor-free enhancement module, small positive anchors are recovered for training.

scale variations across face instances. Previous attempts have been made to alleviate this issue. (1) Multi-scale image pyramids [17] or multi-level feature pyramids [29] are exploited to cope with large ranges of face scales. Image pyramids augment training samples for varying face scales, while feature pyramids offer multi-granularity feature representations for detecting faces with different scales. (2) Various anchor sampling and matching strategies are developed including designing suitable anchor stride [57], adjusting anchor layout [53] or balancing samples at different scales [34]. While these existing methods have shown promising results, they remain two main limitations as follows. First, even though multi-scale training or anchor sampling methods can balance face instances with a large scale range to an extent, those faces with different scales are fed into the network for training at the same time. It might be difficult to obtain a good optimum from learning such complex and varying samples. Second, discrete anchors are tiled on feature maps and are classified as positive and negative based on the metric of intersection-over-union (IoU) overlap. However, small faces may not be fully learned in this way as it is hard to assign precise positive training samples for them.

In this paper, we propose a novel scale-aware training approach to address large scale variations across faces in a different way. Motivated by curriculum

learning where a model is learned by gradually incorporating from easy to complex samples in training, we progressively learn face detection models by feeding grouped face instances into the network in a large-to-small order. The advantages of such progressive learning mechanism are two-fold. (1) Learning easier samples (i.e., large faces) first can provide good initialization for subsequent learning with harder samples (i.e., small faces), which helps improve the final optima of face detectors. (2) The intermediate models learned in the preceding stage can offer a larger effective receptive field for the succeeding learning stages [33]. Thus hard samples will be trained with stronger context information learned before. Fig. 1 (a) and (b) illustrate the motivation of our progressive learning mechanism compared to previous work.

Furthermore, to remedy the issue that small positive anchors may not be discovered based on the criterion of IoU overlap, we develop an auxiliary anchor-free enhancement module to facilitate the learning of small faces. Such anchor-free module will be removed during inference and hence no extra computation cost will be introduced. Fig. 1 (c) and (d) illustrate our motivations on how to remedy the miss of positive anchors for small faces. We also attempt to improve bounding box regression by estimating uncertainty caused by ambiguous annotations. To this end, we learn to predict localization variance for each predicted bounding box.

We extensively evaluate the proposed method, named ProgressFace, on the standard face detection benchmarks of FDDB and WIDER FACE. Our method achieves competitive performance with the state-of-the-art face detectors. Specifically, our ProgressFace with ResNet-152 obtains 98.7% TPR at 1,000 FPs on FDDB and 91.8% AP on the hard set of WIDER FACE, both performing favorably against the state-of-the-arts. Equipped with a light-weight MobileNet-0.25 backbone, we achieve 87.9% AP on the hard set of WIDER FACE, surpassing RetinaFace largely by 9.7%.

The main contributions of this paper are summarized as follows:

- We propose a novel scale-aware progressive learning method for face detection by gradually incorporating large-to-small face instances in training. Such mechanism effectively alleviates the issue of large scale variations and helps improve the quality of feature representations for detecting faces with different scales.
- We propose an anchor-free enhancement module to facilitate the learning of small faces. It serves the anchor-based detection branch with more small positive anchors. This anchor-free module will be removed during inference and does not introduce extra computation cost.
- Our empirical evaluations demonstrate the superiority of the proposed method compared to the state-of-the-arts on both FDDB and WIDER FACE benchmarks. Especially, with the same light-weight MobileNet-0.25 as backbone, our ProgressFace outperforms RetinaFace by a large margin.

2 Related Work

2.1 Generic Object Detection

In the deep learning era, generic object detection has achieved impressive performance due to the powerful representations learned by CNNs. The basic idea of detecting objects is casting this problem as classifying and regressing candidate bounding boxes in images. On the one hand, R-CNN [10] proposes to first generate candidate region proposals and then refine them in the deep network. This two-stage detection method has been improved by a broad range of following work, including reducing redundant calculation of RoI features with spatial pyramid pooling [12], RoIPooling [12] or RoIAlign [11], generating region proposals by RPN [37], improving efficiency by position-sensitive score maps [4], and improving performance by cascade procedure with increasing IoU thresholds [2]. On the other hand, one-stage methods [32] directly classify and refine the pre-defined anchors without region proposal generation. Attempts also have been made to further improve the performance by incorporating additional context information [7], tackling foreground-background class imbalance [30] and developing an anchor refinement module [51].

In contrast to anchor mechanism, an emerging line of recent work attempts to cast object detection as keypoint estimation [44,22,55,56,24,48], instead of enumerating possible locations, scales and aspect ratios by pre-defined anchor boxes. There are different designs in these anchor-free methods for object detection such as finding object centers and regressing to their sizes [18,55], detecting and grouping bounding box corners [24,56], modeling all points [44] or shrunk points [22] in boxes as positive. Different from [46], we integrate an auxiliary anchor-free enhancement module to boost the learning of small faces in this work.

2.2 Face Detection

Face detection has derived benefit from the development of generic object detection. Traditional Harr-AdaBoost [45] and DPM [6] algorithms have trailed deep face detectors. Most of recent face detectors are built upon the anchor-based detection paradigm [37]. Additional attempts have been made to further improve the performance of face detection including integration of context module [17,43,28], adjustment from anchor sampling or matching strategies [53] and utilization of multi-task learning with auxiliary supervision [50,5]. Scale variation is one of key challenges in face detection (e.g., the range of face sizes on WIDER FACE could be 2∼1289). Existing methods tackle the issue in the following aspects. (1) Multi-scale image pyramids are exploited to select specific scales or normalize different scales for training [17,36,40,41]. (2) Multi-level feature pyramids provide features with different spatial resolutions to help detect faces of different sizes [43,28,52]. The detection output can be drawn from multiple feature maps without [32] or with [29] feature fusion. (3) Various anchor sampling or matching strategies are employed for detecting small faces,

including data-anchor-sampling [43,28,27], high overlaps between anchors and ground-truth faces based on EMO score [57], scale compensation anchor matching strategy [53], two-stage anchor refinement [3] and balanced anchor sampling [34]. In this work, we propose a different mechanism to handle large scale variations in face detection by progressively training faces with different scales.

2.3 Curriculum Learning and Progressive Learning

Our work is related to curriculum learning [1] in which samples are not randomly presented but organized in a meaningful order for training. Bengio et al. [1] propose this learning paradigm and its intuition comes from the learning process of humans that gradually incorporates easy-to-hard samples. Self-paced learning further improves curriculum learning by joint optimization of original objective and curriculum design [23], which has been applied to many vision tasks such as visual tracking [42], image search [20] and object discovery [25]. Progressive methods also share similar inspirations with curriculum learning in other problem contexts [31,26] by decomposing complex problems into simpler ones. Our work resembles these learning regimes but we apply free curriculum (i.e., object sizes) to address the issue of large scale variations in the face detection task.

3 Approach

3.1 Anchor-Based Face Detection Baseline

Backbone. We build our backbone of face detection network based on feature pyramid network (FPN) [29], which can incorporate low-level details and high-level semantics. We denote $\{C_i\}|_{i=1}^{n}$ as the last feature map before reducing the spatial resolution in a typical network. Naturally, C_i has the $\frac{1}{2^i}$ resolution of input image. Feature pyramids $\{P_i\}|_{i=l}^{h}$ are extracted by top-down pathways and lateral connections between the l-th and h-th layers. P_i has the same spatial size with the corresponding feature map C_i. Following [43], we build the FPN structure starting from an intermediate layer instead of top layers ($h < n$). Besides, in order to reduce the complexity of FPN structure, we do not incorporate feature maps with too large resolutions ($l > 1$). Feature pyramids $\{P_i\}$ are used as detection outputs and each has an output stride $R = 2^i$.

Anchor Design. We takes anchors with IoU > 0.5 to at least one ground-truth face as positive and those with IoU < 0.3 to all ground-truth faces as negative (i.e., background). Unlike RPN in generic object detection, we restrict the aspect ratios of anchors as one since faces have relatively rigid shape. We set the base anchor size $s_b = 16$, which means the minimum area of anchor boxes is $s_b^2 = 256$. We tile anchors on all the feature pyramids $\{P_i\}|_{i=l}^{h}$. Specifically, suppose we have feature pyramids $\{P_3, P_4, P_5\}$ and each level P_i has two anchor scales, we will use anchor scales $\{1, 2\}$ in P_3, $\{4, 8\}$ in P_4 and $\{16, 32\}$ in P_5. This results in 6 sizes of anchor boxes ($s \times s_b, s \in \{1, 2, 4, 8, 16, 32\}, s_b = 16$) in the 640×640 input image.

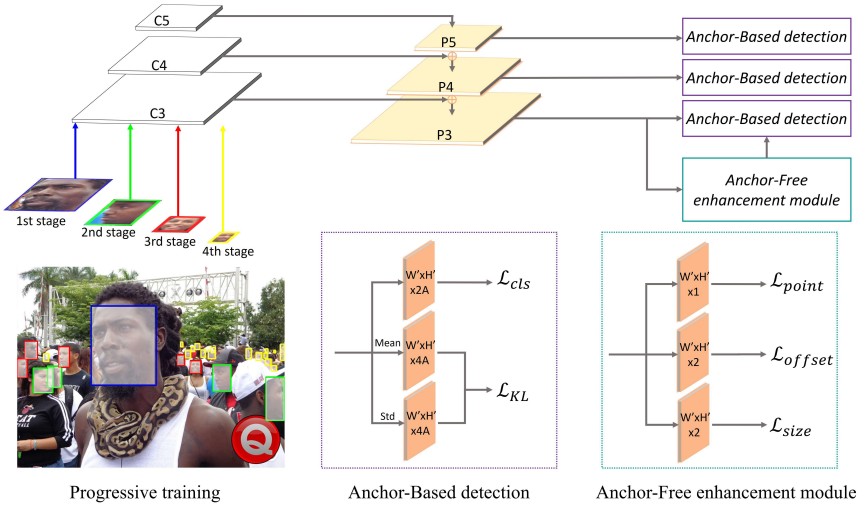


Fig. 2. Overall architecture of the proposed method. See Section 3 for details.

Multi-Task Loss. Following previous anchor-based detectors [30,53,43], we optimize the objective of detection by simultaneously classifying and regressing anchor boxes. Such multi-task loss will be minimized for each anchor i:

$$\mathcal{L} = \mathcal{L}_{\text{cls}}(p_i, p_i^*) + \lambda \cdot p_i^* \mathcal{L}_{\text{reg}}(t_i, t_i^*) \tag{1}$$

The classification loss $\mathcal{L}_{\text{cls}}(p_i, p_i^*)$ is a binary cross-entropy loss to classify positive and negative samples (i.e., faces and background), where p_i is the predicted probability of anchor i being a face and p_i^* represents its ground-truth label (1 for positive and 0 for negative). The localization loss $\mathcal{L}_{\text{reg}}(t_i, t_i^*)$ is a smooth-L1 loss [9], where t_i represents the 4-D coordinate parameters of a predicted box and t_i^* is the ground-truth bounding box. λ is used to balance these two losses and is set to 0.25 in our experiments.

3.2 Progressive Training Framework

Fig. 2 illustrates the overall architecture of our method. Inspired by curriculum learning [1], we propose a progressive training mechanism for face detection by gradually incorporating large-to-small samples. We use the free curriculum, i.e. size of face instances, to guide the entire learning process. Specifically, we first group faces with different scales based on the valid scale range on each level of feature pyramids P_i. Then these grouped faces are gradually fed into the network for training in a large-to-small order. For example, in the first stage, we use the smaller anchor scale of P_5 (i.e., 16) to determine the minimum area of ground-truth faces to be addressed, i.e., $(16 \times s_b)^2$. Thus, face instances with the area of $[(16 \times s_b)^2, +\infty]$ will be valid for training in this stage. In the next stage, the

smaller anchor scale of P_4 is 4 and thus faces with the area of $[(4\times s_b)^2, (16\times s_b)^2]$ will be newly added for training. Such scheme is performed stage by stage until all training samples are included.

Suppose we have K levels of feature pyramids for detection outputs, the training samples will be divided into $K+1$ groups according to the aforementioned progressive learning scheme. In the k-th training stage, we exploit the same optimization objective as Eq. 1 and retrain network parameters which are initialized by the last stage:

$$\begin{gathered}\mathcal{L}^{(k)} = \mathcal{L}(p_i, p_i^*, t_i, t_i^* | \Theta^{(k-1)}),\ t = 1, 2, \ldots, K+1. \\ \Theta^{(k-1)} = \arg\min_{\Theta} \mathcal{L}^{(k-1)}\end{gathered} \tag{2}$$

where Θ indicates the network parameters to be optimized. To avoid getting stuck in local optima induced by subsets of partial samples, we raise the initial learning rate for each training stage.

3.3 Anchor-Free Enhancement Module

In the anchor-based face detection baseline, the anchor scale affects face sizes which can be handled. A metric of IoU overlap is often used to define positive and negative samples. For example, anchors with IoU > 0.5 to ground-truth faces are taken as positive. Such procedure may lead to two main limitations for matching small faces. First, in order to cover more small faces, we need more anchors with smaller size or denser layouts, which will incur extensive computation cost and more imbalanced distributions of positive and negative samples. Second, it is difficult to cover small ground-truth faces and prone to miss the corresponding positive anchors based on this metric. Typically, if the base anchor size is set to 16 and IoU threshold is set to 0.5, faces with area $< 16^2 \times 0.5 = 128$ will be ignored for training [1] if no other scale-aware augmentation strategies are used. Although multi-scale training can be applied to mitigate this issue, it is not efficient especially when the scale range of faces is extremely large.

To remedy the problem of missing small positive anchors in the anchor-based paradigm, we propose an anchor-free enhancement module to facilitate the training of small faces. Specifically, we append an auxiliary anchor-free branch to the feature map P_l with the highest spatial resolution in FPN. The anchor-based branch will generate a label map of $W' \times H' \times A$ to classify anchors, where W' and H' mean the spatial shape of P_l and A represents the amount of anchors for each location. The anchor-free branch will provide more positive anchors by predicting the face centers and regressing their sizes, which leads to an enhanced anchor label map for better training the anchor-based branch.

We train the anchor-free branch by modeling faces as points inspired by CenterNet [55] in generic object detection. Specifically, denote $Y \in [0,1]^{\frac{W}{R}\times\frac{H}{R}}$ as a predicted heatmap where R is the output stride of the feature map, W and H are the size of input image. $Y_{xy} = 1$ means the detected point (x, y) is a face

[1] Faces with area < 128 accounts for $\sim$29% in WIDER FACE.

center and $Y_{xy} = 0$ is background. The training objective of classifying points is pixel-wise logistic regression with focal loss [30]:

$$\mathcal{L}_{\text{point}} = \frac{1}{N} \sum_{x=1}^{\frac{W}{R}} \sum_{y=1}^{\frac{H}{R}} \begin{cases} -(1 - Y_{xy})^{\alpha} log(Y_{xy}) & \text{if } Y^*_{xy} = 1 \\ -(1 - Y^*_{xy})^{\beta} (Y_{xy})^{\alpha} log(1 - Y_{xy}) & \text{otherwise} \end{cases} \tag{3}$$

where Y^*_{xy} is a Gaussian kernel softly representing the ground-truth face center, α and β are hyper-parameters of focal loss, and N is the number of face centers. We use $\alpha = 2$ and $\beta = 4$ in our experiments. To restore the error of discretizing each face center point (x_k, y_k) by the output stride, we use L1 loss to train the offset o_k:

$$\mathcal{L}_{\text{offset}} = \frac{1}{N} \sum_{k=1}^{N} \left| o_k - o^*_k \right|, \text{where } o^*_k = \left(\frac{x_k}{R} - \left\lfloor \frac{x_k}{R} \right\rfloor, \frac{y_k}{R} - \left\lfloor \frac{y_k}{R} \right\rfloor \right) \tag{4}$$

For each ground-truth bounding box $(x^k_1, y^k_1, x^k_2, y^k_2)$, we also regress to the size by L1 loss:

$$\mathcal{L}_{\text{size}} = \frac{1}{N} \sum_{k=1}^{N} \left| s_k - s^*_k \right|, \text{where } s_k = \left(\frac{x^k_2 - x^k_1}{R}, \frac{y^k_2 - y^k_1}{R} \right) \tag{5}$$

We use the following multi-task loss as the training objective to optimize our anchor-free branch:

$$\mathcal{L} = \mathcal{L}_{\text{point}} + \lambda_1 \cdot \mathcal{L}_{\text{offset}} + \lambda_2 \cdot \mathcal{L}_{\text{size}} \tag{6}$$

where $\lambda_1 = 1$ and $\lambda_2 = 0.1$ are used in our experiments.

This anchor-free enhancement module is activated in the last stage of progressive training when small faces are incorporated. At each iteration, points with predicted probabilities $Y_{xy} > T$ will be set as complementary positive anchors. We use $T = 0.7$ in our experiments. For inference, this anchor-free module will be removed and no extra computation cost will be introduced

3.4 Uncertainty Estimation in Face Localization

To improve the robustness and interpretability of deep neural networks, uncertainty estimation has been investigated in Bayesian deep learning by learning a distribution over network weights [21]. Recently, it has also been applied in vision tasks such as face recognition [38] and generic object detection [14]. In this work, we find that ambiguities exist in ground-truth bounding boxes as shown in Fig. 3 (a) and attempt to further improve the quality of face localization by estimating uncertainty.

To address the problem, we estimate the variance of a predicted location for each ground-truth bounding box. In detail, we formulate each possible bounding box location as a Gaussian distribution:

$$P(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\hat{x})^2}{2\sigma^2}} \tag{7}$$

(a) (b)

Fig. 3. (a) Examples of ambiguous ground-truth bounding boxes including occlusion and inaccurate annotations across different face scales in the WIDER FACE dataset. (b) Each predicted bounding box can be modeled with a Gaussian distribution. More accurate location has the smaller variance.

where the mean of gaussian $\hat{x}$ represents the predicted bounding box and the stardard deviation σ represents the estimated uncertainty. Each ground-truth bounding box x^* can be formulated as a Dirac delta function (i.e., Gaussian distribution with $\sigma \to 0$).

$$P_G(x) = \delta(x - x^*) \tag{8}$$

Then the objective is minimizing the KL divergence between the predicted and ground-truth bounding boxes:

$$\mathcal{L}_{\text{KL}} = D_{\text{KL}}(P_G(x) \parallel P(x)) \propto \frac{(x^* - \hat{x})^2}{2\sigma^2} + \frac{\log(\sigma^2)}{2} \tag{9}$$

Following [14], we predict $\alpha = \log \sigma^2$ instead of σ to avoid gradient explosion and exploit a similar smooth-L1 loss for training:

$$\mathcal{L}_{\text{KL}} = \begin{cases} \frac{e^{-\alpha}}{2}(x^* - \hat{x})^2 + \frac{1}{2}\alpha & |x^* - \hat{x}| \leq 1 \\ e^{-\alpha}(|x^* - \hat{x}| - \frac{1}{2}) + \frac{1}{2}\alpha & |x^* - \hat{x}| > 1 \end{cases} \tag{10}$$

The improved bounding box regression loss (Eq. 10) is applied to each progressive training stage and each feature map in FPN. Unlike [14], we only rely on the standard bounding box voting [8] to vote for a more accurate location without using the predicted location variance.

4 Experiments

4.1 Datasets and Evaluation Metrics

WIDER FACE Dataset. The WIDER FACE dataset [47] consists of 32,203 images and 393,703 annotated faces, 158,989 of which are in the *train* set, 39,496 in the *validation* set, and the rest are held out in the *test* set. Each subset has three levels of detection difficulty: *Easy*, *Medium* and *Hard*. It is one of the most challenging face benchmarks with large variations in scale, pose, expression, occlusion and illumination. We use the *train* set of WIDER FACE to train our face detector and perform evaluations on the *validation* and *test* sets.

FDDB Dataset. The FDDB dataset [19] contains 2,845 images and 5,171 annotated faces with different image resolutions, occlusions and poses. We use this dataset for test only.

Evaluation Metrics. We use the standard average precision (AP) metric to evaluate the performance of face detectors on the WIDER FACE dataset. For FDDB, we draw the receiver operating characteristic (ROC) curves and compute the true positive rate (TPR) when the amount of false positives (FP) is equal to 1,000. For both AP and TPR metrics, a predicted bounding box is considered as correct if it has an IoU > 0.5 with a ground-truth face annotation.

4.2 Implementation Details

We summarize other techniques used in our method as follows. We use the five facial landmarks on WIDER FACE provided by [5] to train a auxiliary landmark prediction task with smooth-L1 loss. Thus the multi-task loss in Eq. 1 is improved with an extra term for landmark prediction and its loss weight is set to 0.1 in our experiments. We use online hard example mining (OHEM) [39] and constrain the ratio of positive and negative anchors to $1:3$. We employ context modules [35] on each level of feature pyramid to incorporate more context information and increase the receptive field. We also apply deformable convolution [58] in the feature pyramids as well as context modules.

For data augmentation, we randomly resize an original image from a predefined scale set and randomly crop a fixed size of 640×640 with random flipping as input for training.

We evaluate our method with both ResNet-152 [13] and MobileNet-0.25 [16] backbones. We constrcut 5 levels of feature pyramids for ResNet-152 (P2-6) and 3 levels of feature pyramids for MobileNet-0.25 (P3-5). Both backbones are pretrained on the ImageNet classification task. We use the MobileNet-0.25 backbone to conduct ablation studies.

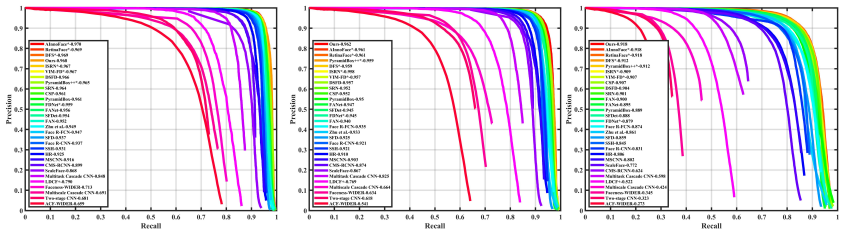
We train the face detection networks with a batch size of 32 on 4 NVIDIA Tesla P100 GPUs. We use Adam to optimize the last stage of progressive training in which the anchor-free module is activated. The initial learning rate is set to 5e-4 and decreased $10\times$ twice during training. We use SGD to optimize the other training stages with momentum of 0.9 and weight decay of 5×10^{-4}. In each stage (except the last one), an initial learning rate of 1e-2 is used and decreased $10\times$ twice. We train for 380 epochs and cost 3 days to obtain the final face detector with the MobileNet-0.25 backbone. For inference, we apply the multi-scale testing strategy [53,5,35] in which the short side of image is resized to $\{500, 800, 1100, 1400, 1700\}$. All of our experiments are conducted on MXNet. Code and our trained face detection models are available at https://github.com/jiashu-zhu/ProgressFace.

4.3 Comparisons to the State-of-the-Arts

Table 1. Performance comparisons on the WIDER FACE *validation* set. * indicates the work which is under review or not formally published. For fair comparisons, FLOPs are computed with the same 640 × 480 input size for all the methods.

Methods	Backbone	*Easy*	*Medium*	*Hard*	Params	FLOPs
MTCNN [50]	Customized	0.851	0.820	0.607	0.50M	4.65G
Faceboxes-3.2x [52]	Customized	0.798	0.802	0.715	1.01M	2.84G
LFFD v2* [15]	Customized	0.837	0.835	0.729	1.45M	6.87G
LFFD v1* [15]	Customized	0.910	0.881	0.780	2.15M	9.25G
RetinaFace* [5]	MobileNet-0.25	0.914	0.901	0.782	0.31M	0.57G
RetinaFace* [5] + DCNv2 [58]	MobileNet-0.25	0.922	0.910	0.795	0.60M	1.23G
ProgressFace-Light	MobileNet0.25	**0.949**	**0.935**	**0.879**	0.66M	1.35G
S^3FD [53]	VGG-16	0.928	0.913	0.840	22.46M	96.60G
SSH [35]	VGG-16	0.927	0.915	0.844	19.75M	99.98G
PyramidBox [43]	VGG-16	0.956	0.946	0.887	57.18M	236.58G
FA-RPN [36]	ResNet-50	0.950	0.942	0.889	-	-
DSFD [27]	VGG-16	0.960	0.953	0.900	141.38M	140.19G
SRN [3]	ResNet-50	0.964	0.953	0.902	-	-
VIM-FD* [54]	DenseNet-121	0.967	0.957	0.907	-	-
PyramidBox++* [28]	VGG-16	0.965	0.959	0.912	-	-
AInnoFace* [49]	ResNet-152	**0.970**	0.961	**0.918**	-	-
RetinaFace* [5]	ResNet-152	0.969	0.961	**0.918**	-	-
ProgressFace	ResNet-152	0.968	**0.962**	**0.918**	68.63M	123.91G

Results on WIDER FACE. Table 1 compares our method with the state-of-the-art approaches on the WIDER FACE *validation* set. Taking the light-weight MobileNet-0.25 as backbone, our ProgressFace-Light only requires 1.35G FLOPs and achieves 87.9% AP on the hard set, significantly surpassing the previous methods. Especially, we outperform RetinaFace with the same backbone by a large margin of 9.7%. For fair comparisons, we also reimplement RetinaFace with DCNv2 [58], which has similar FLOPs with ours. Compared to the improved RetinaFace, we also achieve superior performance (87.9% vs. 79.5%). On the easy and medium sets, our method consistently outperforms the other light-weight face detectors. Taking ResNet-152 as backbone, our ProgressFace achieves detection AP of 96.8%, 96.2%, 91.8% with respect to the easy, medium and hard sets, which is competitive with the state-of-the-art methods. Detailed precision-recall curves on the *validation* set are shown in Fig. 4. On the *test* set, we obtain similar results of 95.9% (easy), 95.7% (medium) and 91.5% (hard). Detailed precision-recall curves on the *test* set are presented in the supplementary material. We also show some detection results on the WIDER FACE *validation* set in Fig. 5. Our method can detect faces in a wide variety of scales, illuminations, poses, scenes and occlusion.



(a) Val:Easy (b) Val:Medium (c) Val:Hard

Fig. 4. Precision-recall curves on the WIDER FACE *validation* set. * indicates the work which is under review or not formally published.

Fig. 5. Sample detection results by our method on the WIDER FACE *validation* set.

Results on FDDB. For evaluations on the FDDB benchmark, we use the trained model on the *train* set of WIDER FACE with the ResNet-152 backbone. Our ProgressFace achieves 98.7% TPR when the amount of false positives is equal to 1,000, which is comparable with existing methods. Detailed ROC curves are presented in the supplementary material.

4.4 Ablation Study

Contributions from Algorithmic Components. We first conduct ablation experiments to show the relative contributions of each algorithmic component in the proposed method. Table 2 compares the baseline with our method in different settings on the WIDER FACE *validation* set. Based on the MobileNet-0.25 backbone, we implement a strong baseline with 85.1% AP on the hard set. With the proposed progressive training mechanism, the performance can be improved by 0.7∼0.9% on the three sets. The results demonstrate that training with samples in the large-to-small order helps learn better face detectors. By applying KL loss for uncertainty estimation in the bounding box regression step, we can obtain a 0.5% gain on the hard set (86.3% vs. 85.8%). After integrating our anchor-free enhancement module, the performance can be further improved, especially on the hard set (87.9% vs. 86.3%). Such results validate the effectiveness of this auxiliary anchor-free module.

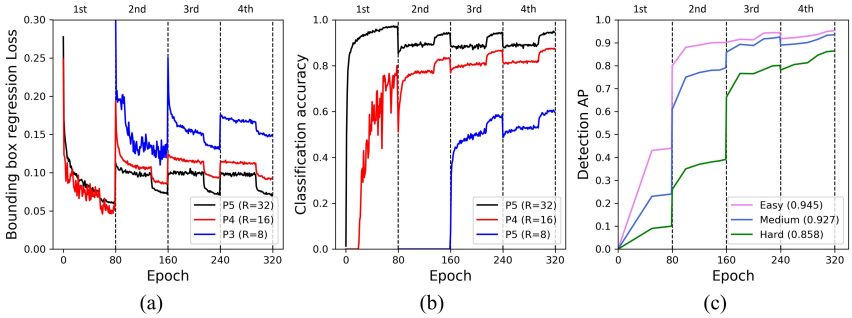


Fig. 6. (a) Loss curve for bounding box regression loss during training. (b) Classification accuracy during training. (c) Detection AP performance during validation.

Table 2. Ablation experiments of our methods on the WIDER FACE *validation* set. PT: Progressive training scheme. UE: Uncertainty estimation by KL loss. AF: Anchor-free enhancement module.

Baseline	PT	UE	AF	*Easy*	*Medium*	*Hard*
✓				0.937	0.918	0.851
✓	✓			0.945	0.927	0.858
✓	✓	✓		0.946	0.929	0.863
✓	✓		✓	0.949	0.933	0.876
✓	✓	✓	✓	**0.949**	**0.935**	**0.879**

Discussions on Progressive Training. To further examine the effect of progressive training on the performance, we also train the same epochs for the baseline method. The results show that training longer only introduces a slight performance boost on the hard set (85.3% vs. 85.1%). With the same training epochs, the progressive learning scheme still can obtain another 0.5% improvement (85.8% vs. 85.3%). In addition, we show the bounding box regression loss, classification accuracy during training and detection performance during validation in Fig. 6. We observe that the validation performance increases with gradually incorporating easy-to-hard samples stage by stage. Even though easy samples encounter the potential risk of overfitting in the early stage, incorporation of more complex samples in the subsequent stage will mitigate this issue. Moreover, in order to avoid getting stuck in the intermediate sub-optimal solutions, we increase the initial learning rate of each stage when new samples are added into training.

Anchor-Based vs. Anchor-Free. To better understand the effect of our anchor-free enhancement module, we conduct three sets of ablation experiments in Table 3 to investigate the effects of different optimization methods, different levels of feature pyramids and different test schemes. (1) In the first group of

Table 3. Ablation experiments of anchor-based and anchor-free methods.

	Methods	*Easy*	*Medium*	*Hard*
Optimization methods	Anchor-based only	0.937	0.918	0.851
	Anchor-free only	0.879	0.870	0.813
	Anchor-based + Anchor-free	0.939	0.920	0.860
Feature pyramids	Anchor-based + Anchor-free (P_3)	0.949	0.935	0.879
	Anchor-based + Anchor-free (P_4)	0.946	0.930	0.867
	Anchor-based + Anchor-free (P_5)	0.944	0.930	0.864
Test schemes	Anchor-based only	0.949	0.935	0.879
	Anchor-free only	0.889	0.882	0.828
	Anchor-based + Anchor-free	0.947	0.932	0.876

Table 3, the results show training with anchor-based branches only outperforms training with anchor-free only. We accordingly choose the anchor-based method as our strong baseline. After combining these two optimization methods, the performance is better than either of them, which validates the motivation of our anchor-free enhancement module. (2) We add the anchor-free module to different levels of feature pyramids and compare their performance. Implementing such module on the lowest feature map P_3 in FPN obtains the best performance. The results validate our observations that small positive anchors tend to be missed on the low feature map. We also try adding anchor-free modules to each anchor-based branch and no more gains are obtained. (3) After training the anchor-based face detector with the anchor-free enhancement module together, we compare different test schemes. We found that only using the output of anchor-based branches is responsible for good results. Simply combining the output of anchor-based and anchor-free branches will not be a good choice because their generated scores tend to have different distributions.

5 Conclusion

In this paper, we propose a novel scale-aware progressive training mechanism to address large scale variations for face detection. Inspired by curriculum learning, our method gradually learns large-to-small face instances during training. We propose an auxiliary anchor-free enhancement module to facilitate the learning of small faces. We also apply KL loss to further improve bounding box regression by estimating uncertainty caused by ambiguous annotations. Extensive experimental results demonstrate the superiority of our method on the standard FDDB and WIDER FACE benchmarks. Especially, our ProgressFace with the MobileNet-0.25 backbone achieves 87.9% AP on the hard set of WIDER FACE, surpassing RetinaFace largely with the same backbone by 9.7%.

References

1. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: ICML (2009) 5, 6
2. Cai, Z., Vasconcelos, N.: Cascade r-cnn: Delving into high quality object detection. In: CVPR (2018) 4
3. Chi, C., Zhang, S., Xing, J., Lei, Z., Li, S.Z., Zou, X.: Selective refinement network for high performance face detection. In: AAAI (2019) 5, 11
4. Dai, J., Li, Y., He, K., Sun, J.: R-fcn: Object detection via region-based fully convolutional networks. In: NeurIPS (2016) 4
5. 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, 10, 11
6. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. TPAMI **32**(9), 1627–1645 (2009) 4
7. Fu, C.Y., Liu, W., Ranga, A., Tyagi, A., Berg, A.C.: Dssd: Deconvolutional single shot detector. arXiv preprint arXiv:1701.06659 (2017) 4
8. Gidaris, S., Komodakis, N.: Object detection via a multi-region and semantic segmentation-aware cnn model. In: ICCV (2015) 9
9. Girshick, R.: Fast r-cnn. In: ICCV (2015) 6
10. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014) 4
11. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: ICCV (2017) 4
12. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. TPAMI **37**(9), 1904–1916 (2015) 4
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) 10
14. He, Y., Zhu, C., Wang, J., Savvides, M., Zhang, X.: Bounding box regression with uncertainty for accurate object detection. In: CVPR (2019) 8, 9
15. 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) 11
16. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017) 10
17. Hu, P., Ramanan, D.: Finding tiny faces. In: CVPR (2017) 2, 4
18. Huang, L., Yang, Y., Deng, Y., Yu, Y.: Densebox: Unifying landmark localization with end to end object detection. arXiv preprint arXiv:1509.04874 (2015) 4
19. Jain, V., Learned-Miller, E.: Fddb: A benchmark for face detection in unconstrained settings. Tech. rep., UMass Amherst technical report (2010) 10
20. Jiang, L., Meng, D., Mitamura, T., Hauptmann, A.G.: Easy samples first: Self-paced reranking for zero-example multimedia search. In: ACM MM (2014) 5
21. Kendall, A., Gal, Y.: What uncertainties do we need in bayesian deep learning for computer vision? In: NeurIPS (2017) 8
22. Kong, T., Sun, F., Liu, H., Jiang, Y., Shi, J.: Foveabox: Beyond anchor-based object detector. arXiv preprint arXiv:1904.03797 (2019) 4
23. Kumar, M.P., Packer, B., Koller, D.: Self-paced learning for latent variable models. In: NeurIPS (2010) 5
24. Law, H., Deng, J.: Cornernet: Detecting objects as paired keypoints. In: ECCV (2018) 4

25. Lee, Y.J., Grauman, K.: Learning the easy things first: Self-paced visual category discovery. In: CVPR (2011) 5
26. Li, D., Huang, J.B., Li, Y., Wang, S., Yang, M.H.: Weakly supervised object localization with progressive domain adaptation. In: CVPR (2016) 5
27. Li, J., Wang, Y., Wang, C., Tai, Y., Qian, J., Yang, J., Wang, C., Li, J., Huang, F.: Dsfd: dual shot face detector. In: CVPR (2019) 5, 11
28. Li, Z., Tang, X., Han, J., Liu, J., He, R.: Pyramidbox++: High performance detector for finding tiny face. arXiv preprint arXiv:1904.00386 (2019) 4, 5, 11
29. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: CVPR (2017) 2, 4, 5
30. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: ICCV (2017) 1, 4, 6, 8
31. Liu, C., Zoph, B., Neumann, M., Shlens, J., Hua, W., Li, L.J., Fei-Fei, L., Yuille, A., Huang, J., Murphy, K.: Progressive neural architecture search. In: ECCV (2018) 5
32. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: ECCV (2016) 4
33. Luo, W., Li, Y., Urtasun, R., Zemel, R.: Understanding the effective receptive field in deep convolutional neural networks. In: NeurIPS (2016) 3
34. Ming, X., Wei, F., Zhang, T., Chen, D., Wen, F.: Group sampling for scale invariant face detection. In: CVPR (2019) 2, 5
35. Najibi, M., Samangouei, P., Chellappa, R., Davis, L.S.: Ssh: Single stage headless face detector. In: ICCV (2017) 10, 11
36. Najibi, M., Singh, B., Davis, L.S.: Fa-rpn: Floating region proposals for face detection. In: CVPR (2019) 4, 11
37. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. TPAMI **39**(6), 1137–1149 (2015) 1, 4
38. Shi, Y., Jain, A.K.: Probabilistic face embeddings. In: ICCV (2019) 8
39. Shrivastava, A., Gupta, A., Girshick, R.: Training region-based object detectors with online hard example mining. In: CVPR (2016) 10
40. Singh, B., Davis, L.S.: An analysis of scale invariance in object detection snip. In: CVPR (2018) 4
41. Singh, B., Najibi, M., Davis, L.S.: Sniper: Efficient multi-scale training. In: NeurIPS (2018) 4
42. Supancic, J.S., Ramanan, D.: Self-paced learning for long-term tracking. In: CVPR (2013) 5
43. Tang, X., Du, D.K., He, Z., Liu, J.: Pyramidbox: A context-assisted single shot face detector. In: ECCV (2018) 4, 5, 6, 11
44. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection. In: ICCV (2019) 4
45. Viola, P., Jones, M.J.: Robust real-time face detection. IJCV **57**(2), 137–154 (2004) 4
46. 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) 4
47. Yang, S., Luo, P., Loy, C.C., Tang, X.: Wider face: A face detection benchmark. In: CVPR (2016) 9
48. Yu, J., Jiang, Y., Wang, Z., Cao, Z., Huang, T.: Unitbox: An advanced object detection network. In: ACMMM (2016) 4
49. Zhang, F., Fan, X., Ai, G., Song, J., Qin, Y., Wu, J.: Accurate face detection for high performance. arXiv preprint arXiv:1905.01585 (2019) 11

50. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters **23**(10), 1499–1503 (2016) 4, 11
51. Zhang, S., Wen, L., Bian, X., Lei, Z., Li, S.Z.: Single-shot refinement neural network for object detection. In: CVPR (2018) 4
52. Zhang, S., Zhu, X., Lei, Z., Shi, H., Wang, X., Li, S.Z.: Faceboxes: A cpu real-time face detector with high accuracy. In: IJCB (2017) 4, 11
53. Zhang, S., Zhu, X., Lei, Z., Shi, H., Wang, X., Li, S.Z.: S3fd: Single shot scale-invariant face detector. In: ICCV (2017) 2, 4, 5, 6, 10, 11
54. Zhang, Y., Xu, X., Liu, X.: Robust and high performance face detector. arXiv preprint arXiv:1901.02350 (2019) 11
55. Zhou, X., Wang, D., Krähenbühl, P.: Objects as points. arXiv preprint arXiv:1904.07850 (2019) 4, 7
56. Zhou, X., Zhuo, J., Krahenbuhl, P.: Bottom-up object detection by grouping extreme and center points. In: CVPR (2019) 4
57. Zhu, C., Tao, R., Luu, K., Savvides, M.: Seeing small faces from robust anchor's perspective. In: CVPR (2018) 2, 5
58. Zhu, X., Hu, H., Lin, S., Dai, J.: Deformable convnets v2: More deformable, better results. In: CVPR (2019) 10, 11