

# CRFace: Confidence Ranker for Model-Agnostic Face Detection Refinement

Noranart Vesdapunt  
Microsoft Cloud&AI  
noves@microsoft.com

Baoyuan Wang  
Xiaobing.AI  
zjuwby@gmail.com

## Abstract

*Face detection is a fundamental problem for many downstream face applications, and there is a rising demand for faster, more accurate yet support for higher resolution face detectors. Recent smartphones can record a video in 8K resolution, but many of the existing face detectors still fail due to the anchor size and training data. We analyze the failure cases and observe a large number of correct predicted boxes with incorrect confidences. To calibrate these confidences, we propose a confidence ranking network with a pairwise ranking loss to re-rank the predicted confidences locally within the same image. Our confidence ranker is model-agnostic, so we can augment the data by choosing the pairs from multiple face detectors during the training, and generalize to a wide range of face detectors during the testing. On WiderFace, we achieve the highest AP on the single-scale, and our AP is competitive with the previous multi-scale methods while being significantly faster. On 8K resolution, our method solves the GPU memory issue and allows us to indirectly train on 8K. We collect 8K resolution test set to show the improvement, and we will release our test set as a new benchmark for future research.*

## 1. Introduction

Face detection is a long-standing research topic in computer vision. Many important downstream applications need to build on top of it, including face reconstruction, face tracking and face recognition, etc. Thanks to Convolution Neural Network, face detection has been improved significantly in the past few years. However, as the camera hardware is evolving, the demand for an even faster, more accurate, and support for higher resolution in face detector is rising. Recent smartphones (Samsung Galaxy S20 Ultra, Xiaomi MI 10 Pro) can capture a 108MP image and record a video in 8K resolution. We believe that 8K resolution will be practical in the near future, but a lot of existing works [40, 13, 33, 72] still fail on many 8K resolution inputs.

The main reasons for failure are anchor size and training data. Most of the anchors are designed for the most popular face detection dataset, WiderFace [63] which only has

an image width of 1024. The common largest anchor size is 512 [40, 13, 33, 72], so the detector has to predict up to 7K residuals. While predicting such large residuals is hard, but possible, the classifier has never seen such large resolution in the training set and it will almost always suppress all these boxes. Collecting 8K training data is expensive because a single image could contain hundreds of faces (Fig. 1). Designing a large anchor requires the network to be deep enough to output multiple scales up to 8K, but BFBBox [39] shows that the average precision (AP) decreases when the network becomes deeper. While down-sampling the image allows the detector to find a large face, the small face could be reduced to just 1 pixel (Fig. 1). We also found that 48GB GPU memory is not enough to train RetinaFace [13] on 8K resolution. As there are many challenges in 8K face detection, multi-scale [26] is perhaps the most reasonable workaround by predicting the small faces in 8K, and the large faces in the smaller resolution, then fuse them together with box voting [16]. However, this approach still relies on the correct confidence as the majority of the high confidence boxes will determine the result. Multi-scale also decreases the 8K prediction speed (which is already severely slow) to become much slower than a single-scale.

We analyze the fail cases and found that most of the correct box locations have already existed because many detectors predict a large number of boxes, but the confidences are very low. To systematically test these boxes, we replace the prediction confidence with the intersect-over-union (IoU) between the predicted box and the closest ground truth box and call it: oracle confidence. Fig. 2 shows that the oracle prediction has a consistently high AP on WiderFace validation set across multiple resolutions, and by testing only on a single-scale, it can outperform the state-of-the-art [40] by a large margin (AP Hard: 93.3% vs 98.4%). On our collected FFHQ [30] dataset, AP could even be increased from 16.2% to 96.5%. We saw the possibility of closing the gap between the predicted and oracle confidence. Since we can use the oracle confidence as ground truth, we formulate this confidence refinement problem as supervised learning.

We initially try to learn a regressor that takes an image and face detector's output as input to predict the new con-



Figure 1. (Best viewed electronically) Examples of our 8K test set. The first two images contain both large and small faces. Detecting on the large resolution will fail the large face due to the small anchor size and the training data. Down-sampling will cause the small faces to be too small to detect. Last two images contain a large amount of small faces that are expensive to label.

fidence, then pass them into NMS. This, however, does not improve AP because regressing to the exact floating point is challenging. The confidences of the boxes around the face boundary could be only  $10^{-4}$  different from each other, and a very small regression error can significantly change the confidence order, and thus changing the outcome of the non-maximal suppression (NMS). Since NMS is a greedy technique that is only affected by the order of the confidence, we propose to relax the problem from regression to local ordinal ranking where we only need to rank the refined confidences within the same image and we can ignore the magnitude of the confidence. We use a pairwise ranking loss to enforce such constraint, and we show that our ranker learns to preserve the order of confidence and improve AP.

We propose a confidence ranking network by taking the bounding box and confidence prediction from a face detector to output new refined confidences. On top of Feature Pyramid Network (FPN [36]), we add a Box Processing Network (BPN) to extract features from the face detector’s output and interpolate them to concatenate with the image feature, then pass them into our confidence module. Fig. 3 demonstrates our pipeline. We design our network to be model-agnostic, so in theory, it can be used with any object detector, as long as the outputs are bounding boxes and their corresponding confidence values, and we show the generalization on three face detectors including HAMBox [40], RetinaFace [13], and HRNet [61]. Model-agnostic design is important for a fair comparison with multi-scale which is the de facto post-processing for modern face detectors in order to get high AP, with the cost of slow speed. Since our network only needs a single-scale to run, we are a few times faster than those state-of-the-arts [40, 13] with multi-scale, while still retain a competitive AP on WiderFace. On a single-scale, our method is the new state-of-the-art. Furthermore, our method allows us to solve the GPU memory issue and indirectly train on 8K (it would otherwise not be possible even with 48GB GPU memory) by backpropagating up until the 8K prediction from the face detector. We then collect 8K test set to demonstrate the effectiveness of our method and set up a new benchmark for future camera hardware. To summarize, our main contributions are:

1. We propose to refine confidence in a local relative setting with our ranking loss, in contrast to existing works that regress to the absolute value in a global manner, inspired by the failure case analysis on WiderFace validation set and our collected FFHQ dataset.
2. We propose a confidence ranking network to achieve the new state-of-the-art on the single-scale face detector. Our AP is competitive with the previous multi-scale state-of-the-art [40] while remaining a few times faster. Our network is model-agnostic and we show the generalization on HAMBox [40], RetinaFace [13], and HRNet [61] respectively.
3. Our method solves the GPU memory issue and allows us to indirectly train on 8K resolution to further increase AP. We collect 8K resolution test set to show the improvement, and we will release our test sets as a new benchmark for future camera hardware.

## 2. Related Work

**Object Detection** has been advanced significantly by deep learning and there are many types of deep learning based detectors. Two-stage detectors [18, 17, 53, 21, 5, 12] generate region proposals in the first stage, then refine them in the second stage. One-stage detectors [51, 52, 38, 37, 71] remove the first stage and run directly on the predefined anchors. Anchor-free detectors [31, 32, 59, 65, 74, 75] remove anchor to reduce design parameters. Although each type is different, we design our network to only use the box and confidence from the input detector, so in theory, our work is compatible with these wide ranges of detectors.

**Face Detection** is a special case of object detection and they can leverage each other. On top of a generic object detector, past works propose to change the anchor matching [72, 9, 43, 76, 33, 40] or change the anchor sampling [46] to detect small faces, add context module [26, 57, 35] to learn contextual features, add face alignment task [69, 8, 13] to leverage different dataset, search for face-specific backbone [39] to increase network capacity, or train progressively [77] to handle multi-scale. While training strategies are

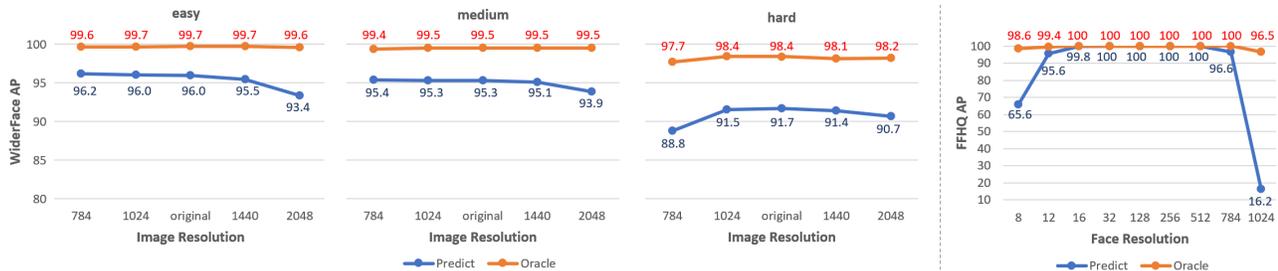


Figure 2. AP of predicted vs oracle box from HRNet [61, 55] face detector. We resize the image height of WiderFace validation set (left figure) to multiple resolutions (x-axis), test on the single-scale, and observe a large consistent gap across all resolutions. We double verify the gap in a more controlled test set by labeling 1,875 images from FFHQ dataset (right figure). We tested on resized face resolution and the conclusion remains the same. This shows that most of the corrected boxes are already predicted, but the confidences are incorrect.

varied, recent face detectors focus on the one-stage design [72, 45, 8, 13, 77, 57, 35, 40] due to the benefit of dense anchor sampling and scale variation from feature pyramid. Our network is inspired by these works and we add extra layers to process the box and output the new confidence. Many state-of-the-arts [13, 77, 40] consistently see improvement when applying multi-scale [26]. We found that the detector can predict most of the corrected box, but the confidences are incorrect, so we propose to speed up by refining the confidence and only run on a single-scale.

**Detection Refinement** is commonly used in object detection [71, 6, 44, 19, 1, 54]. The idea is to detect the object multiple times to refine the prediction. In face detection, previous works [15, 42, 69, 41] propose to iteratively regress the box, or use cascade networks [48, 50, 49, 66] to refine the prediction. Our approach is a form of refinement where we propose a second network to only refine the confidence with extra supervision from the oracle confidence. While IoU-Net [28] has proposed the oracle confidence for object detection before, we are the first to introduce the oracle confidence to face detection through the observation from Fig. 2. For face detection, confidence is the bottleneck for the prediction error, so we can bypass the box refinement step in IoU-Net, which is a heavy optimization process. We also show the importance of changing from Smooth L1 loss in IoU-Net to our ranking loss. Lastly, our design is model-agnostic, in contrast to IoU-Net’s dependency on the PrRoiPooling [28] layer.

**Ranking Loss** is a well-studied loss function, especially in information retrieval, and is grouped into Pointwise ([34, 11, 23], Pairwise ([3, 2, 4]), and Listwise ([7, 62, 60]). In this paper, we show the issue with the existing regression losses, and with such a simple pairwise ranking loss, we were able to achieve significant improvement.

### 3. Methodology

We built our baseline by re-implementing HAMBox [40], and to simplify the baseline, we do not use Pyramid

Anchor [57], Deep Head [37], or regression-aware focal loss [40]. We instead change the backbone of HAMBox from ResNet-50 [22] to HRNet-W48 [61, 55] to further close the AP gap. With the new baseline, we analyze the failure case, propose a ranking loss, and design a model-agnostic confidence ranking network to further improve AP.

#### 3.1. Oracle Confidence

We first look at the failure cases from our baseline, and we found multiple reasons such as occlusion, small face, large pose, etc. While we could tackle one failure case at a time, we found a common behavior that the correct bounding boxes are predicted, but the predicted confidences are incorrect. Out of the 4,062 false negative faces on WiderFace validation set, only 531 faces can not be predicted by the detector. If we can correct these confidences, 87% of the false negatives will be fixed, and all of the false positives will be suppressed by setting their confidences to zero.

To achieve better confidences, we first consider optimizing new confidences based on the predicted box as the input, and AP as the cost function. This optimization is too heavy as our baseline detector can predict >5k boxes and we need to rerun NMS and recompute AP on every iteration. By definition, the new confidence should be a distance metric between the predicted and ground truth box, so although not the most optimal solution, we replace the predicted confidence with the IoU between a predicted box and the closest ground truth box, and call it: oracle confidence. We tested our oracle prediction across multiple scales of WiderFace validation set and we see AP>99% on the Easy and Medium set. On Hard set, if we resize the resolution down to 784 (almost half), we start to see the AP drop to 97.7% because some of the faces are being resized to <5 pixels (could be 1 pixel). The missing boxes are the limitation of our method, but there is no alternative solution that can detect a 1-pixel face. By testing on WiderFace, one could argue that the challenging image condition causes the confidence prediction to be less accurate, and the oracle prediction improvement came from using the ground truth

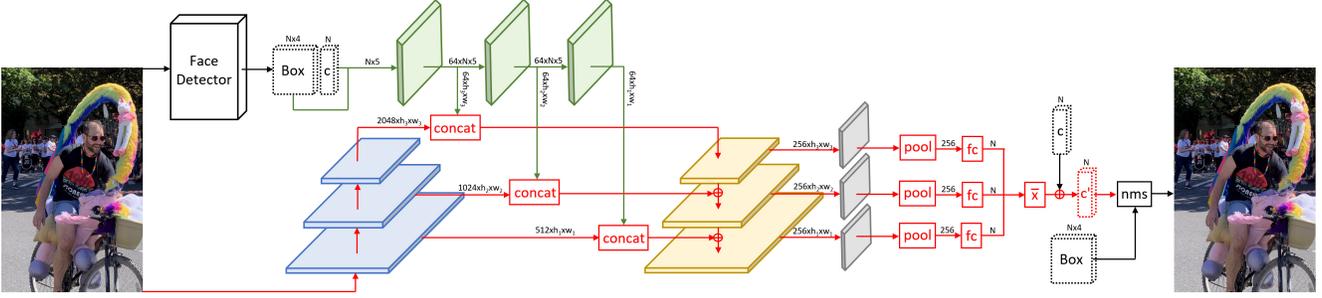


Figure 3. An overview of our pipeline. We use FPN with 3 scales to extract features from the image. We pass the face detector’s output into our Box Processing Network (BPN, denoted in green) and interpolate BPN’s feature dimension to concatenate with the image features. Our confidence module applies global average pooling and fully-connected layer to the features from each scale, then averages them to output the refined confidence ( $c'$ ). The final prediction is the bounding box from the face detector and the new confidence from our network.

to overcome these issues. Therefore, we labeled 1,875 images from FFHQ, which only has high quality, single face with minimal background, to double verify our oracle confidence. Fig. 2 shows the AP comparisons.

### 3.2. Ranking Loss

After obtaining the oracle confidence, we can train to refine confidence in a supervised learning fashion. We first formulate the problem as regression and test multiple loss functions. While the regressor is powerful enough to move the new confidence towards the oracle confidence distribution, it is not accurate enough to preserve the order, especially for boxes around the face. This is due to the small IoU difference between them and it results in a very small loss. To relax this problem, we observe a few priors from NMS. First, the confidence magnitude does not change the NMS output, as long as the order remains the same within an image. Second, the order across different images does not change the outcome of NMS, which means these confidences do not have to be globally ranked across the entire dataset. We just need to focus on the order of the confidence within the same image, so we propose to use a pairwise ranking loss. We first define the relationship between the ground truth pair as a binary classification problem.

$$Y(c_{gt,1}, c_{gt,2}) = \begin{cases} 0 & \text{if } c_{gt,1} > c_{gt,2} \\ 1 & \text{if } c_{gt,1} \leq c_{gt,2} \end{cases} \quad (1)$$

where  $c_{gt}$  is the oracle confidence or the IoU between the predicted and ground truth box.  $Y$  is the class label from a pair of oracle confidences. We then subtract a pair of predicted confidences to obtain a linear difference between them and pass it into a Sigmoid cross entropy loss.

$$L(c', c_{gt}) = -(c'_1 - c'_2) * \log(Y(c_{gt,1}, c_{gt,2})) - (1 - (c'_1 - c'_2)) * \log(1 - Y(c_{gt,1}, c_{gt,2})) \quad (2)$$

where  $c'_1$  and  $c'_2$  are the predicted confidence pair. We consider using the margin rescaling technique [29, 58] and this loss can push the confidences away from each other (Fig. 4), but the AP is about the same as our simpler ranking loss. We select the pair by sorting the predicted confidences in descending order and pick the neighbor pair. This pair selecting strategy exposes the pair that the detector struggles with the most, but we also would like to include an easy pair as well, not just the pair with a subtle difference. We add n-pair to the original neighbor pair by skipping n confidences and divide the loss by n, for example, 2-pairs are the neighbor pair and the odd and even pair. We observe that the number of pairs can affect AP. Fig. 4 shows the comparison of the confidence distributions from each loss.

### 3.3. Model-Agnostic Design

We consider multiple types of inputs for our confidence ranking network. Our design principle is to make our network be compatible with as many face detectors as possible so that we can increase the number of training data by combining the predictions from all these detectors. Our model-agnostic design can be viewed as a plug-in module for an existing face detector. In this paper, we use this refinement to tackle the 8K resolution input, but the same idea can also be applied for other scenarios, and since we do not change the predicted box, we are preserving the original face detector’s behavior as much as possible. This could be a good design for a personalized face detector where we can train a specialized confidence refinement module for each scenario, but it is out of scope for this paper.

To simplify the design, we take the predicted box and confidence from a face detector as an  $N \times 5$  matrix and feed them together with an image into our network. It is possible to further simplify by only refining the confidence without the image input, but this is overfitting to a specific detector. We do not take the feature from the detector’s backbone due to the design complication. Choosing the feature is not

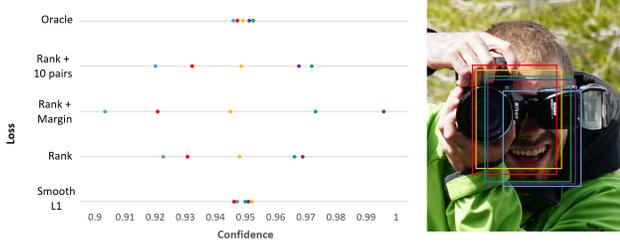


Figure 4. Confidence distribution visualization for each loss function. Smooth L1 loss has the closest distribution to the oracle (ground truth), but the order is not preserved. Ranking loss does not respect the confidence magnitude, but performs better at preserving the order. Margin rescaling technique [29, 58] can push the distribution further but does not change the order much. Selecting the number of pairs (10 pairs) can also change the order.

trivial because of the diversity of network architectures, for example, there are multiple feature scales to be chosen from HRNet, or a large expand layer in MobileNet V3 [24] or EfficientDet [56] will subsequently increase the refine network size due to the large number of channels in the input. We would like our design to be easily plugged into as many detectors as possible.

Our design avoids the insufficient GPU memory issue in order to train on 8K resolution images. We attempt to resize WiderFace training set to 8K resolution, and train RetinaFace [13] on Nvidia Quadro 8000 (48GB memory), but this is still not enough for 8K images. Forwarding an 8K image, on the other hand, does not require as much memory, and it is much faster. We forward all the 8K images to obtain 8K boxes and their confidences, then use them to train our confidence ranking network with the original WiderFace resolution. This allows our network to indirectly learn from the 8K images through the 8K prediction of the face detector, and we do not back-propagate to the face detector, so we do not require the large GPU memory.

### 3.4. Confidence Ranking Network

Confidence refinement problem is a sub-task of the detection problem where the region proposal is fixed, and the network learns to predict new confidences. We follow a popular detector design, feature pyramid network (FPN), to inherit the scale variation feature. We then add a box processing network (BPN) by expanding the channel size to 64 (expand  $N \times 5$  to  $64 \times N \times 5$ ) and pass the box feature through multiple convolutions and leaky ReLU layers. We use the convolution with  $5 \times 3$  kernel to guarantee enough field of view to cover the face detector output (5 numbers:  $x$ ,  $y$ ,  $w$ ,  $h$ , confidence). We add paddings to keep the  $64 \times N \times 5$  dimension throughout BPN and we add skip connections between them to preserve the original face detector information. We only interpolate these features with the nearest neighbor before concatenating with the image feature from

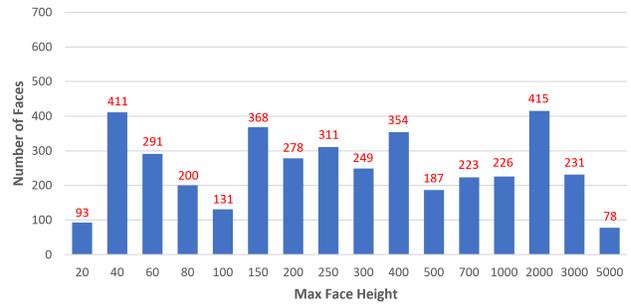


Figure 5. Face height distribution of our 8K resolution test set. The X-axis denotes the size range, for example, 40 is the face with  $20 < \text{height} \leq 40$ . We cover a wide range of face resolution and has a good balance of the number of faces in each resolution bin.

FPN. The concatenated features are then passed into the remaining FPN layers, and we modify SSH [45] from a detection module to a confidence module. We first use global average pooling to force the feature dimension to 256 then apply fully connected layers to change the output dimension to the number of confidence from the face detector. We then average all the features across all the scales from FPN to obtain the confidence residual and we sum it back with the predicted confidence from the face detector. We found that learning the new confidence directly (without using the residual) is possible, but the training is less stable, and it takes longer to train. The newly refined confidences are then passed into the ranking loss in Eq.2 during the training. During the testing, we pass the refined confidences and the boxes from the face detector into NMS to obtain the result. Fig. 3 demonstrates our pipeline.

## 4. Experimental Setup

### 4.1. Datasets

We train our network on WiderFace [63] training set and test on WiderFace validation set. WiderFace has 32,203 images with an image width of 1024. This dataset focuses on small faces, where half of all faces are  $\leq 50$  pixels, and there are 393,703 labeled faces (on average:  $>10$  faces/image). We collect our 8K resolution test set from Div8K [20] and YouTube. We remove the duplicate frames by computing image similarity with the threshold greater 50%, which ends up with 1,428 images and they cover the full spectrum of face heights from 9 pixels to 4,130 pixels. Fig. 1 shows example images, and Fig. 5 shows the face size distribution of our 8K resolution test set.

### 4.2. Training Configurations

Our network is implemented in PyTorch [47] and trained using Adam optimizer with batch size 32 for 100K iterations. The initial learning rate is  $10^{-3}$  and it decays exponentially until  $10^{-6}$ . Training takes about a day on an

Loss	AP (%)		
	Easy	Med	Hard
Baseline	95.8	95.2	91.5
Cross Entropy	95.0	94.0	89.7
L1	95.1	94.4	90.0
L2	95.1	94.3	90.4
Smooth L1	95.2	94.5	90.7
<b>Rank</b>	<b>96.9</b>	<b>96.2</b>	<b>92.7</b>

Table 1. Ablation study on each loss function. The baseline is HRNet on a single-scale. Regression losses are worse than the baseline, but our ranking loss can improve AP Hard by +1.2%.

Pair	AP (%)		
	Easy	Med	Hard
Baseline	95.8	95.2	91.5
1	96.7	95.8	91.9
2	96.8	95.9	92.2
3	96.7	95.9	92.1
5	96.8	96.0	92.3
<b>10</b>	<b>96.9</b>	<b>96.2</b>	<b>92.7</b>
20	96.8	96.0	92.4
50	96.8	95.9	92.1
100	96.7	95.8	91.9

Table 2. Ablation study on the number of pairs for pairwise ranking loss. A single pair only includes hard pairs, and 100-pairs focuses on easy pairs. Ten pairs yield the best trade-off.

Nvidia RTX 2080. For data augmentation, we randomly crop, scale, mirror, and change brightness, contrast, hue, and saturation. When we transform the image, we use the same transformation on the predicted box from the face detector directly, without rerunning the augmented image. This is not perfectly accurate, but it is very close and much faster. We normalize the predicted box by the image width and height to keep the input range between 0 to 1. We also clip the refined confidence ( $c'$ ) to be within 0 to 1 to keep the same output range as the standard confidence probability. We use 5K face detection outputs ( $N=5000$ ). We sort the boxes descendingly by face detector’s confidences and remove the boxes if the number exceeds 5K, or we fill the matrix with 0 if there are less than 5K boxes. We combine face detector outputs from RetinaFace, HAMBox, and HRNet, then randomly pick the output from a single face detector on each iteration. The same confidence ranker with NMS (IOU 0.4) is used across all detectors during testing.

### 4.3. Network Architecture

We follow FPN design to extract the image feature with ResNet-50 backbone. BPN consists of 9 convolutions with 64 channels and kernel size 3x5 with padding 1x2 to preserve the same Nx5 dimensions throughout the entire network. We add skip connections on every layer except the first layer. Every 3 layers, we interpolate BPN’s features

Model	Scale	AP (%)			Speed (ms)
		Easy	Med	Hard	
RetinaFace	single	95.2	94.5	84.3	52
	multi	96.4	95.5	90.3	1048
	rank	96.1	95.3	88.7	107
HAMBox	single	95.1	94.2	89.1	70
	multi	96.2	95.3	90.9	1383
	rank	96.1	95.4	91.7	126
HRNet (ours)	single	95.8	95.2	91.5	91
	multi	96.9	96.1	92.1	1886
	<b>rank</b>	<b>96.9</b>	<b>96.2</b>	<b>92.7</b>	<b>149</b>

Table 3. Results on WiderFace validation set. Our confidence ranker (rank) improves AP on top of the single-scale (single) across all backbones. On HRNet, confidence ranker’s AP Hard outperforms multi-scale (multi) by +0.6%, and runs 12.7x faster.

Model	Scale	AP (%)	Speed (ms)
RetinaFace	single	24.1	428
	multi	38.8	1579
	rank	40.7	988
HAMBox	single	21.8	495
	multi	34.9	1875
	rank	37.3	1060
HRNet (ours)	single	22.5	705
	multi	34.8	2690
	rank	37.0	1261
	<b>rank 8k</b>	<b>42.5</b>	<b>1263</b>

Table 4. Results on 8K test set. Our confidence ranker (rank) on the single-scale (scale) outperforms multi-scale (multi) on all backbones, and runs 1.7-2.1x faster. Training confidence ranker on 8K face detection outputs further increases AP by +5.5%.

into the image feature dimension and concatenate with the intermediate features from the backbone. The concatenated features are then passed into FPN layers which is a 3x3 convolution with 256 channels and perform the merge on each scale. Each of the FPN features is then passed into a 3x3 convolution with 256 channels, global average pooling, and fully-connected layers (FC) to change into 5000 numbers. FC outputs are then summed up with the face detector’s confidence ( $c$ ) to output the refined confidence ( $c'$ ).

## 5. Results

### 5.1. Ablation Studies

**Importance of Ranking Loss.** We attempt to compare our method with existing detection refinement works, but it is unclear how to change their designs to be model-agnostic, for example, MS R-CNN [27] relies on mask head output and RoiAlign features, IoU-Net [28] relies on PrRoiPooling features. All these layers do not exist in most face detectors and ignoring these layers will break the refinement because

the refiner was trained for these specific features. Loss functions, on the other hand, can be swapped to compare fairly. We use HRNet as the baseline to study the loss functions, and test the AP on WiderFace validation set. Tab. 1 shows a significant improvement between the ranking loss and all of the other commonly used losses (Cross Entropy is used in [41, 42, 49, 50, 48, 69], L2 [27], Smooth L1 [28]). The improvement is because the regression losses focus on bringing the confidence magnitude towards the oracle confidence, but the small regression error can easily change the confidence order (as observed in Fig. 4).

**Importance of Pair Selection.** We then tested the number of pairs for our pairwise ranking loss. Since it is too computationally intensive to use all pairs (5K boxes = 12M pairs), we select the pair by sorting the prediction confidence from the face detector and use the neighbor pair. Tab. 2 shows that when the number of pairs is too small, our network will only focus on the hard pairs. On the other hand, when the number of pairs is too large, the network will only focus on the easy pairs. Empirically, using 10 pairs yields a good trade-off between the hard and easy pairs, and thus achieving the highest AP. We use 10 pairs throughout the paper. We show performance curves in the supplementary.

## 5.2. Internal Comparison

We show the comparisons of a single-scale, multi-scale, and our confidence ranker on the single-scale on WiderFace validation set. For single-scale testing, we use the image as-is without any resizing (image width is 1024). For multi-scale testing, we follow RetinaFace [13] by resizing the short edge of the image to [500, 800, 1100, 1400, 1700], as well as flipping the image on each scale, then fuse them together with box voting [16] with an IoU threshold of 0.4. Tab. 3 shows that on HRNet, our ranking network, despite running on a single-scale, outperforms the AP hard of the multi-scale prediction, and achieves 12.7x speed up. We show the generalization on the open-source RetinaFace [14] and our re-implement HAMBox [40] in Tab. 3. On RetinaFace, the confidence ranker increases AP Hard on the single-scale by 4.4%, and it is 9.8x faster than multi-scale. On HAMBox, the confidence ranker allows the single-scale to outperform multi-scale on AP hard by 0.8%, along with 11x speed up. Performance curves are in Fig. 6.

## 5.3. External Comparison

We externally compare our method on WiderFace validation set. Tab. 5 shows that we outperform RetinaFace’s AP Hard by 0.5%, and we can estimate HAMBox’s AP. We demonstrate in sec.5.2 that our method is compatible with HAMBox, but the AP Hard of our re-implementation is 2.4% behind HAMBox, so our final model is 0.6% lower. We show the performance curves for external comparisons

Model	AP (%)			Multi-scale
	Easy	Med	Hard	
S <sup>3</sup> FD [72]	92.8	91.3	84.0	Yes
SSH [45]	92.7	91.5	84.4	No
PyramidBox [57]	95.6	94.6	88.7	Yes
FA-RPN [46]	95.0	94.2	88.9	Yes
DSFD [33]	96.0	95.3	90.0	Yes
SRN [10]	96.4	95.3	90.2	Yes
VIM-FD* [73]	96.7	95.7	90.7	-
PyramidBox++* [35]	96.5	95.9	91.2	Yes
MaskFace* [64]	97.2	96.5	91.5	Yes
BFBBox [39]	96.5	95.7	91.7	-
AInnoFace* [68]	97.0	96.1	91.8	Yes
ProgressFace [77]	96.8	96.2	91.8	Yes
RefineFace [70]	97.1	96.2	91.8	Yes
ASFD* [67]	97.2	96.2	92.0	Yes
RetinaFace [13]	96.9	96.3	92.2	Yes
HAMBox [40]	97.0	96.4	93.3	Yes
<b>Ours</b>	<b>96.9</b>	<b>96.2</b>	<b>92.7</b>	<b>No</b>

Table 5. External comparison on WiderFace validation set. Works with \* are not formally published. Our method is competitive with multi-scale, and achieve the highest AP on single-scale.

in Fig. 7. Most of the previous works adopt multi-scale which is significantly slower than the single-scale. RetinaFace reported the unofficial single-scale AP [14] as Easy: 96.5%, Medium: 95.6%, Hard: 90.4%. We outperform RetinaFace by Easy: +0.4%, Medium: +0.6%, Hard: +2.3%, and achieve the highest AP on the single-scale.

## 5.4. 8K Resolution

We study the impact of single-scale, multi-scale, and our method on 8K resolution test set. For multi-scale, we follow RetinaFace by down-sampling the smallest scale to approximately half of the original image dimension. We omit the up-sampling scales because the 48GB GPU memory is not enough to run them. We resize the long edge to [7680, 5760, 4096] and flip the image on each scale, then fuse them together with box voting. Tab. 4 shows that our confidence ranker on the single-scale outperforms the multi-scale on all three detectors. The AP improvements are +1.9%, +2.4%, +2.2%, and the speedups are 1.7x, 1.8x, 2.1x for RetinaFace, HAMBox and HRNet respectively. This shows the generalization of our method. By resizing the WiderFace training set to 8K resolution and feed them into all three detectors, we can train our confidence ranker on 8K face detector outputs (which is not possible to train on the face detector directly due to GPU memory limitation). Our confidence ranker, trained on WiderFace in 8K resolution, can further increase AP on HRNet by 5.5% and the speed remains unchanged. We show the visual comparison in Fig. 8 and the performance curve in the supplementary.

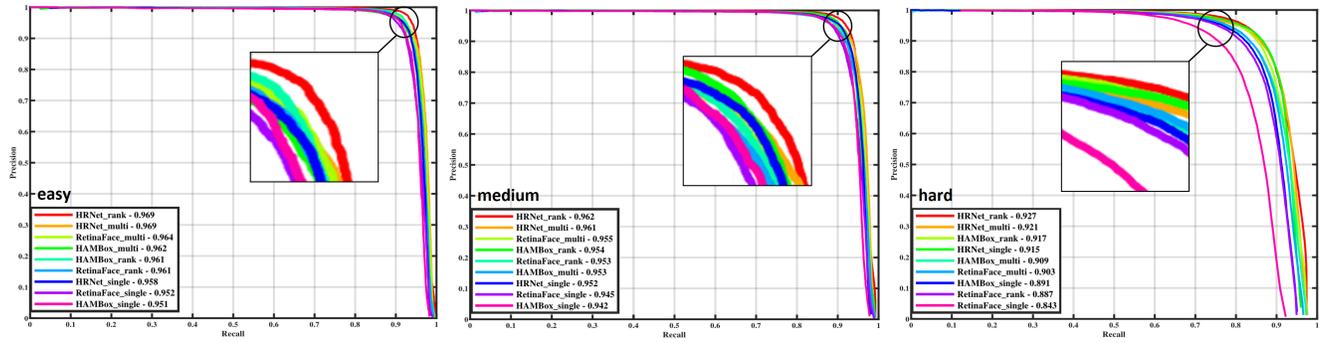


Figure 6. Performance curves for internal comparison. We plot the single-scale, multi-scale (multi), and our confidence ranker (rank) across multiple face detectors on WiderFace validation set. HRNet with confidence ranker achieves the highest AP across all settings.

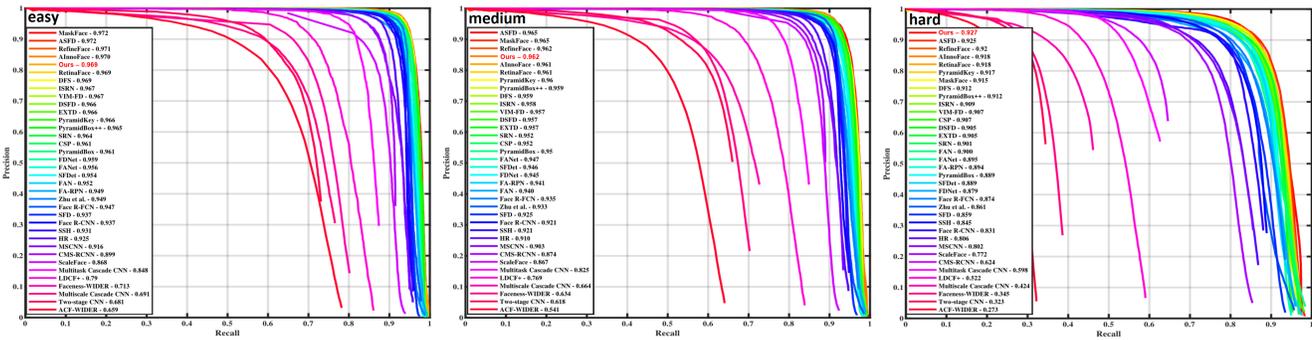


Figure 7. Performance curves for external comparison on WiderFace validation set. All of the top 5 detectors (except ours) across all settings use multi-scale. We can approximate multi-scale results with our confidence ranker on a single-scale.

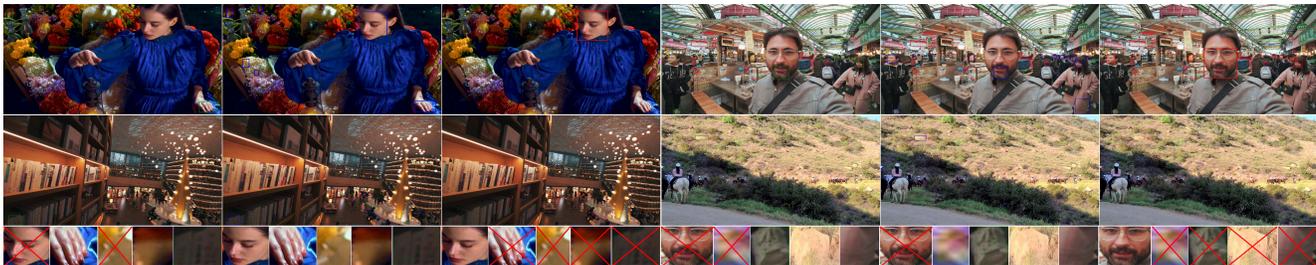


Figure 8. (Best viewed electronically) Prediction examples on 8K test set. We show the predictions from HRNet with confidences  $>0.5$  on the single-scale (left), multi-scale (mid), and confidence ranker trained on 8K (right). Single-scale misses multiple large faces, and multi-scale is more prone to false positives. Our confidence ranker can mitigate both failure cases.

## 5.5. Computational Complexity

Our confidence ranker is 128MB and the speed on an Nvidia RTX 2080, is 45ms/frame and 550ms/frame for 1024 and 8K respectively. Our confidence ranker is in this size because face detectors are the bottleneck where HRNet and HAMBox run at 81ms/frame and 65ms/frame on 1024. To achieve a smaller footprint, we change the backbone from ResNet-50 to MobileNet v1 [25] and reduce the channel size of the confidence module from 256 to 64 (this will reduce the FC layer from  $256 \Rightarrow 5000$  to  $64 \Rightarrow 5000$ ). Our small model is 8MB and it achieves AP Easy: 96.6%, Medium: 95.7%, Hard: 91.7% on WiderFace validation set with 6ms/frame on a 1024 and 59ms/frame on 8K.

## 6. Conclusion and Future Works

We propose a confidence ranking network with a pairwise ranking loss to re-rank the incorrect confidences. On WiderFace, we achieve the highest AP on the single-scale, and we can approximate the AP of the previous multi-scale methods with a few times speed up. Our confidence ranker is model-agnostic and supports training indirectly on 8K resolution. For future work, it is possible to combine ranking loss with other techniques, such as adversarial loss, for improving challenging cases such as blurry, underexposed, and heavily occluded faces. In theory, our approach can also be applied for generic object detection and instance segmentation. We leave them as future explorations.

## References

- [1] Vijay Aroulanandam, Thamarai Latchoumi, Battula Bhavya, and Shaik Sultana. Object detection in convolution neural networks using iterative refinements. *Revue d'Intelligence Artificielle*, 33(5):367–372, Nov. 2019. 3
- [2] Christopher Burges, Robert Ragno, and Quoc Le. Learning to rank with nonsmooth cost functions. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2007. 3
- [3] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning, ICML '05*, pages 89–96, New York, NY, USA, 2005. ACM. 3
- [4] Christopher J. C. Burges. From RankNet to LambdaRank to LambdaMART: An overview. Technical report, Microsoft Research, 2010. 3
- [5] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. *computer vision and pattern recognition*, 2018. 2
- [6] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3
- [7] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *ICML*, pages 129–136, 2007. 3
- [8] Bindita Chaudhuri, Noranart Vespapunt, and Baoyuan Wang. Joint face detection and facial motion retargeting for multiple faces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2, 3
- [9] Cheng Chi, Shifeng Zhang, Junliang Xing, Zhen Lei, Stan Z. Li, and Xudong Zou. Selective refinement network for high performance face detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:8231–8238, July 2019. 2
- [10] Cheng Chi, Shifeng Zhang, Junliang Xing, Zhen Lei, Stan Z. Li, and Xudong Zou. Selective refinement network for high performance face detection. In *AAAI*, volume 33, pages 8231–8238, 2019. 7
- [11] Koby Crammer and Yoram Singer. Pranking with ranking. In *Advances in Neural Information Processing Systems 14*, pages 641–647. MIT Press, 2001. 3
- [12] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29, pages 379–387. Curran Associates, Inc., 2016. 2
- [13] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2, 3, 5, 7
- [14] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. <https://github.com/deepinsight/insightface/tree/master/detection/RetinaFace>, 2020. 7
- [15] Zhen-Hua Feng, Josef Kittler, Muhammad Awais, Patrik Huber, and Xiao-Jun Wu. Face detection, bounding box aggregation and pose estimation for robust facial landmark localisation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017. 3
- [16] Spyros Gidaris and Nikos Komodakis. Object detection via a multi-region and semantic segmentation-aware cnn model. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015. 1, 7
- [17] Ross Girshick. Fast r-cnn. In *ICCV*, pages 1440–1448, 2015. 2
- [18] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014. 2
- [19] Jicheng Gong, Z. Zhao, and Nic Li. Improving multi-stage object detection via iterative proposal refinement. In *BMVC*, 2019. 3
- [20] S. Gu, A. Lugmayr, M. Danelljan, M. Fritsche, J. Lamour, and R. Timofte. Div8k: Diverse 8k resolution image dataset. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3512–3516, 2019. 5
- [21] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017. 2
- [22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. 3
- [23] Ralf Herbrich, Thore Graepel, and Klaus Obermayer. Large margin rank boundaries for ordinal regression. In P. J. Bartlett, B. Schölkopf, D. Schuurmans, and A. J. Smola, editors, *Advances in Large Margin Classifiers*, pages 115–132. MIT Press, 2000. 3
- [24] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, Quoc V. Le, and Hartwig Adam. Searching for mobilenetv3. *CoRR*, abs/1905.02244, 2019. 5
- [25] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications, 2017. 8
- [26] Peiyun Hu and Deva Ramanan. Finding tiny faces. In *CVPR*, pages 951–959, 2017. 1, 2, 3
- [27] Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask Scoring R-CNN. In *CVPR*, 2019. 6, 7
- [28] Borui Jiang, Ruixuan Luo, Jiayuan Mao, Tete Xiao, and Yuning Jiang. Acquisition of localization confidence for accurate object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 3, 6, 7

- [29] Thorsten Joachims, Thomas Hofmann, Yisong Yue, and Chun-Nam Yu. Predicting structured objects with support vector machines. *Commun. ACM*, 52(11):97–104, Nov. 2009. 4, 5
- [30] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1
- [31] Tao Kong, Fuchun Sun, Huaping Liu, Yuning Jiang, Lei Li, and Jianbo Shi. Foveabox: Beyond anchor-based object detector. *IEEE Transactions on Image Processing*, pages 7389–7398, 2020. 2
- [32] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 2
- [33] Jian Li, Yabiao Wang, Changan Wang, Ying Tai, Jianjun Qian, Jian Yang, Chengjie Wang, Jilin Li, and Feiyue Huang. Dsf: dual shot face detector. In *CVPR*, pages 5060–5069, 2019. 1, 2, 7
- [34] Ping Li, Christopher J. C. Burges, and Qiang Wu. Mcrank: Learning to rank using multiple classification and gradient boosting. In John C. Platt, Daphne Koller, Yoram Singer, and Sam T. Roweis, editors, *NIPS*, pages 897–904. Curran Associates, Inc., 2007. 3
- [35] Zhihang Li, Xu Tang, Junyu Han, Jingtuo Liu, and Ran He. Pyramidbox++: High performance detector for finding tiny face. *arXiv preprint arXiv:1904.00386*, 2019. 2, 3, 7
- [36] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2
- [37] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2980–2988, 2017. 2, 3
- [38] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, pages 21–37. Springer, 2016. 2
- [39] Yang Liu and Xu Tang. Bfbox: Searching face-appropriate backbone and feature pyramid network for face detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2, 7
- [40] Yang Liu, Xu Tang, Junyu Han, Jingtuo Liu, Dinger Rui, and Xiang Wu. Hambox: Delving into mining high-quality anchors on face detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2, 3, 7
- [41] Dazhi Luo, Guihua Wen, Danyang Li, Yang Hu, and Eryang Huan. Deep-learning-based face detection using iterative bounding-box regression. *Multimedia Tools and Applications*, 77(19):24663–24680, Feb. 2018. 3, 7
- [42] Jingdong Ma and Yupin Luo. Iterative face detection from the global to local. pages 320–331, 11 2019. 3, 7
- [43] Xiang Ming, Fangyun Wei, Ting Zhang, Dong Chen, and Fang Wen. Group sampling for scale invariant face detection. In *CVPR*, pages 3446–3456, 2019. 2
- [44] Mahyar Najibi, Mohammad Rastegari, and Larry S. Davis. G-cnn: An iterative grid based object detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 3
- [45] Mahyar Najibi, Pouya Samangouei, Rama Chellappa, and Larry S Davis. Ssh: Single stage headless face detector. In *ICCV*, pages 4875–4884, 2017. 3, 5, 7
- [46] Mahyar Najibi, Bharat Singh, and Larry S. Davis. Fa-rpn: Floating region proposals for face detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2, 7
- [47] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 5
- [48] R. Qi, R. Jia, Q. Mao, H. Sun, and L. Zuo. Face detection method based on cascaded convolutional networks. *IEEE Access*, 7:110740–110748, 2019. 3, 7
- [49] Rong Qi, Rui-Sheng Jia, Qi-Chao Mao, Hong-Mei Sun, and Ling-Qun Zuo. Face detection method based on cascaded convolutional networks. *IEEE Access*, PP:1–1, 08 2019. 3, 7
- [50] H. Qin, J. Yan, X. Li, and X. Hu. Joint training of cascaded cnn for face detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3456–3465, 2016. 3, 7
- [51] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2
- [52] Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2
- [53] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015. 2
- [54] Danila Rukhovich, Konstantin Sofiiuk, Danil Galeev, Olga Barinova, and Anton Konushin. Iterdet: Iterative scheme for objectdetection in crowded environments, 2020. 3
- [55] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019. 3
- [56] Mingxing Tan, Ruoming Pang, and Quoc V. Le. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 5
- [57] Xu Tang, Daniel K Du, Zeqiang He, and Jingtuo Liu. Pyramidbox: A context-assisted single shot face detector. In *ECCV*, pages 797–813, 2018. 2, 3, 7
- [58] Ben Taskar, Carlos Guestrin, and Daphne Koller. Max-margin markov networks. *NIPS'03*, pages 25–32, 2003. 4, 5

- [59] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. FCOS: Fully convolutional one-stage object detection. In *IEEE International Conference on Computer Vision (ICCV'19)*, 2019. [2](#)
- [60] Maksims N. Volkovs and Richard S. Zemel. Boltzrank: learning to maximize expected ranking gain. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09*, pages 1089–1096, New York, NY, USA, 2009. ACM. [3](#)
- [61] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *TPAMI*, 2019. [2, 3](#)
- [62] Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank: theory and algorithm. In *ICML*, pages 1192–1199, 2008. [3](#)
- [63] Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang. Wider face: A face detection benchmark. In *CVPR*, pages 5525–5533, 2016. [1, 5](#)
- [64] Dmitry Yashunin, Tamir Baydasov, and Roman Vlasov. Maskface: multi-task face and landmark detector, 2020. [7](#)
- [65] Jiahui Yu, Yuning Jiang, Zhangyang Wang, Zhimin Cao, and Thomas Huang. Unitbox: An advanced object detection network. In *ACMM 2016*, pages 516–520. Association for Computing Machinery, Inc, Oct. 2016. [2](#)
- [66] Dan Zeng, Fan Zhao, Shiming Ge, and W. Shen. Fast cascade face detection with pyramid network. *Pattern Recognit. Lett.*, 119:180–186, 2019. [3](#)
- [67] Bin Zhang, Jian Li, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, Yili Xia, Wenjiang Pei, and Ron-grong Ji. Asfd: Automatic and scalable face detector, 2020. [7](#)
- [68] Faen Zhang, Xinyu Fan, Guo Ai, Jianfei Song, Yongqiang Qin, and Jiahong Wu. Accurate face detection for high performance. *CoRR*, abs/1905.01585, 2019. [7](#)
- [69] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, Oct 2016. [2, 3, 7](#)
- [70] S. Zhang, C. Chi, Z. Lei, and S. Z. Li. Refineface: Refinement neural network for high performance face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020. [7](#)
- [71] Shifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, and Stan Z. Li. Single-shot refinement neural network for object detection. In *CVPR*, 2018. [2, 3](#)
- [72] Shifeng Zhang, Xiangyu Zhu, Zhen Lei, Hailin Shi, Xiaobo Wang, and Stan Z Li. S3fd: Single shot scale-invariant face detector. In *ICCV*, pages 192–201, 2017. [1, 2, 3, 7](#)
- [73] Yundong Zhang, Xiang Xu, and Xiaotao Liu. Robust and high performance face detector. *CoRR*, abs/1901.02350, 2019. [7](#)
- [74] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *CoRR*, abs/1904.07850, 2019. [2](#)
- [75] Xingyi Zhou, Jiacheng Zhuo, and Philipp Krahenbuhl. Bottom-up object detection by grouping extreme and center points. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. [2](#)
- [76] Chenchen Zhu, Ran Tao, Khoa Luu, and Marios Savvides. Seeing small faces from robust anchor’s perspective. In *ICCV*, pages 5127–5136, 2018. [2](#)
- [77] Jiashu Zhu, Dong Li, Tiantian Han, Lu Tian, and Yi Shan. Progressface: Scale-aware progressive learning for face detection. 2020. [2, 3, 7](#)