Scale-Aware Face Detection

Zekun Hao[1], Yu Liu[1], Hongwei Qin[2], Junjie Yan[2], Xiu Li[2], Xiaolin Hu[2]
[1]SenseTime, [2]Tsinghua University
{haozekun, yanjunjie}@outlook.com, liuyuisanai@gmail.com,
{qhw12@mails., xlhu@, li.xiu@sz.}tsinghua.edu.cn

Abstract

Convolutional neural network (CNN) based face detectors are inefficient in handling faces of diverse scales. They rely on either fitting a large single model to faces across a large scale range or multi-scale testing. Both are computationally expensive. We propose Scale-aware Face Detection (SAFD) to handle scale explicitly using CNN, and achieve better performance with less computation cost. Prior to detection, an efficient CNN predicts the scale distribution histogram of the faces. Then the scale histogram guides the zoom-in and zoom-out of the image. Since the faces will be approximately in uniform scale after zoom, they can be detected accurately even with much smaller CNN. Actually, more than 99% of the faces in AFW can be covered with less than two zooms per image. Extensive experiments on FDDB, MALF and AFW show advantages of SAFD.

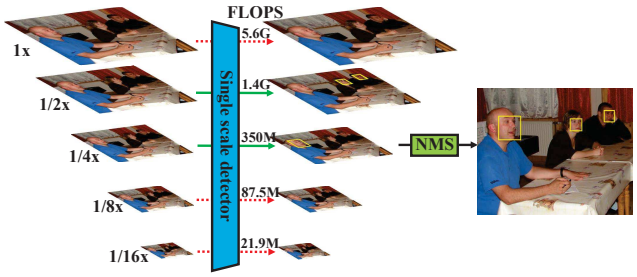


Figure 1. **The motivation of SAFD.** Single-scale detectors need to perform multi-scale testing on image pyramids in order to cover a large scale range. However, in most cases only a few layers in the image pyramids contain faces of valid scales (green arrow). Finding faces on those invalid scales is a waste of computation (red dashed arrow). In the proposed method, we show that the prediction of those valid scales can be done efficiently by a CNN, which considerably reduces computation.

1. Introduction

Face detection is one of the most widely used computer vision applications. Popular face detectors have been proposed, including the Viola-Jones[34]and its extensions, part model [9] and its successors and the convolutional neural network (CNN) based approaches [33]. The CNN based approaches have recently shown great successes [13, 39, 4].

A face detection system should be able to handle faces of various scales, poses and appearances. For CNN-based face detectors, the variance in pose and appearance can be handled by the large capacity of convolutional neural network. The variance in scale, however, is not carefully considered and there is room for improvement. The popularity of CNN in computer vision domain largely comes from its translation invariance property, which significantly reduces computation and model size compared to fully-connected neural networks. However, as for scale invariance, CNN meets the limitation that is similar to the limitation of translation invariance for fully-connected networks. The CNN does not inherently have scale invariance. A CNN can be trained to have certain extent of scale invariance, but it needs more parameters and more complex structures to retain performance. Despite the importance, works that involve scale are rarely seen, and no work focuses on the essence of the scale problem. One possible reason is that in academic research, the simple multi-scale testing on image pyramids can be used to avoid the problem and get good accuracy. However, multi-scale testing leads to heavy computation cost. Another way to avoid this problem is to fit a CNN model to multiple scales. This may also lead to an increase in model size and computation.

To solve this problem, we consider estimating the scale explicitly. If we know the face scales in each image, we can resize the image to suitable scales that best fit the detector. It eliminates the need to cover variances caused by scales so that smaller detector network can be used while achieving even better performance. It also prevents exhaustively testing all the scales in an image pyramid, which saves computation, as illustrated in Figure 1.

In this way, the face detection procedure can be divided into face scale estimation and single scale detection.

The scale proposal stage is implemented through a light-weight, fully-convolutional network called *Scale Proposal*

Network (SPN). The network can generate a global face scale histogram from an input image of arbitrary sizes. A global max-pooling layer is placed at the end of the network, so it outputs a fixed-length vector regardless to the size of input image. The histogram vector encodes the probability of the existence of faces at certain scales. The input image is resized according to the histogram to ensure all the faces are within the valid range of the following detection stage. The SPN can be trained with the image-level supervision of ground truth histogram vectors and no face location information is required.

The second stage is single-scale face detection. The face scales of the training images have already been normalized to a narrow range prior to detection, so a simple detector that covers a narrow scale range can achieve high performance. We use a Region Proposal Network(RPN) as the detector in all the experiments because it is simple, fast and accurate on face detection task because there is only one object class.

By using the two-stage SA-RPN method, the average computation cost can be reduced while achieving state-of-the-art accuracy. The reasons are two-fold. On one hand, the single-scale detector adopts a smaller network than a multi-scale detector. Experiments show that a small network performs better if it only focuses on faces within a narrow scale range. On the other hand, when a face occupies a large part of the image, it can be down-sampled to save computation in detection. When a face is smaller than the optimal range, up-sampling makes it easier to be detected.

Contributions. The contributions are in the following:

1. We propose to divide face detection problem into two sub-problems: scale estimation and single-scale detection. Both problems are cheap in computation and overall computation is reduced while achieving state-of-the-art performance on FDDB, MALF and AFW.
2. We introduce SPN for generating fine-grained scale proposals and the network can be trained easily via image-level supervision.

2. Related works

The CNN based face detection approaches emerged in 1990s [33]. Some of the modules are still widely used, such as sliding window, multi-scale testing and the CNN based classifier to distinguish faces from background. [31] shows that CNN achieves good performance for frontal face detection and [32] further extends it for rotation invariant face detection by training faces of different poses. Despite their good performance, they are too slow when considering the hardware of early years.

One breakthrough in face detection is the Viola-Jones framework [34], which combines Haar feature, Adaboost and cascade in face detection. It becomes very popular due to its advantages in both speed and accuracy. Many works have been proposed to improve the Viola-Jones framework and achieves further improvements, such as local features [41, 20, 36], boosting algorithms [40, 21, 11], cascade structure [2] and multi-pose [22, 17, 12].

The HOG based methods are firstly used in pedestrian or general object detection, such as the famous HOG [6] and deformable part model [9]. These methods achieves better performance than Viola-Jones based methods on standard benchmarks such as AFW [42] and FDDB [16], and progressively become more efficient, including [42, 25, 35, 10].

The CNN based methods again become popular thank to their great performance advantages. Early works combine CNN based features with traditional features. [28] combines CNN with deformable part model and [37] combines CNN with channel feature [7]. [39] predicts face part score map through fully convolutional networks and uses it to generate face proposals for further classification. [19] proposes a CNN cascade for efficient face detection. This work is further improved in [26] with joint training. [13] gives an end-to-end training version of detection network to directly predict bounding boxes and object confidences. [8] shows that simple fine-tuning the CNN model from ImageNet classification task for face/background classification leads to good performance. In [4], the supervised spatial transform layer is used to implement pose invariant face detection. Popular general object detection methods, such as Faster-RCNN [30], R-FCN [5], YOLO [29] and SSD [24] can also be used directly for face detection. Our proposed scale-aware face detection method is also a CNN-based method. However, it focuses on scale problem in face detection, in the way that, to our best knowledge, no one has ever explored yet. Our method is orthogonal to these CNN-based methods and they can benefit from each other.

There are some successful attempts on better handling of scale in object detection. They either construct stronger network structure by combining features from different depths of a network [1] or directly predicting objects at different depth of a network [3, 24]. All of them share the same motivation. Intuitively, larger faces require a network with larger receptive field to be detected correctly, while smaller faces need a network with high resolution (and possibly smaller receptive field) to have it detected and localized correctly. But these methods have two major drawbacks. First, they fail to explicitly share feature between scales. These methods only share feature implicitly by sharing part of the convolution layers. The network still have to cover large scale variance, possibly needing more parameters to work well. Second, in order to cover largest and smallest faces simultaneously in a single pass, the input image has to be large to prevent small faces from missing, even if the image doesn't contain small faces at all. This hurts speed considerably,

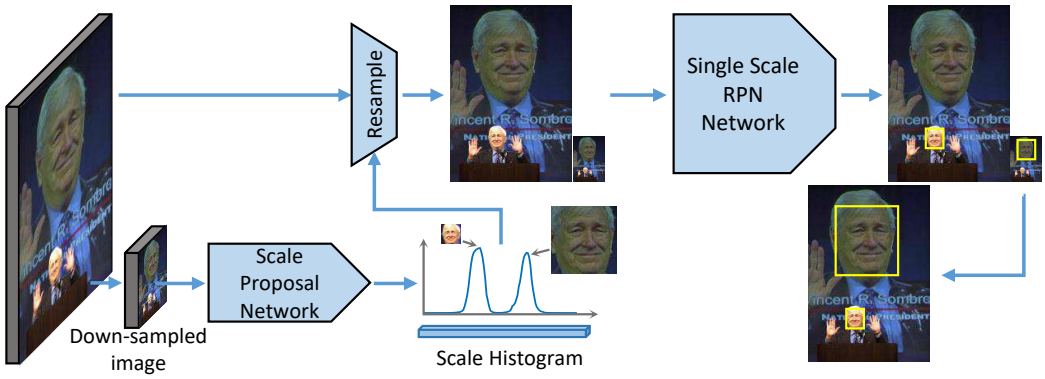


Figure 2. **The pipeline of Scale-Aware Face Detector.** Firstly, the input image is resampled to a small size and forwarded through Scale Proposal Network (SPN) to obtain Scale Histogram. The Scale Histogram encodes the possible sizes of faces in the image but it doesn't contain any location information. The SPN network needs little computation. Then the input image is resampled according to the Scale Histogram so that all the faces in the image fall in the coverable range of RPN. Computation can be reduced if the image contains only large faces. Finally, the resampled image set is individually detected for faces and the results are combined to obtain the final result.

and can be inferred from the FLOPs comparison in Figure 1. Both problems are tackled in SAFD.

3. Scale-aware detection pipeline

We propose SAFD that implicitly considers face scale variation. As illustrated in Figure 2, our method consists of two stages, which disassembles face detection problem into two sub-problems: (1) global scale proposal and (2) single-scale detection. The goal of global scale proposal stage is to estimate the possible sizes of all the faces appearing in the image as well as assign a confidence score to each scale proposals. Then the image is scaled according to the scale proposals and detected for faces using single-scale RPN. If multiple scale proposals are generated in one image, it is scaled and detected for multiple times and results are combined to form the final detection result.

3.1. Scale Proposal Network (SPN)

We define scale proposals to be a set of estimated face sizes along with their confidences. The definition of face size is discussed in Section 4.2. In scale proposal stage, scale proposals are generated by Scale Proposal Network(SPN), a specially-designed convolutional neural network that aims at generating scale histogram with minimum human-introduced constraints.

The Scale Proposal Network is a fully convolutional network that has a global max-pooling layer after the last convolution layer for generating a fixed-length histogram vector from an input image of arbitrary size. Figure 3 shows the structure of Scale Proposal Network. It takes the down-sampled image as input, and produces a scale response heatmap (of size $w \times h \times n$). After global max-pooling the heatmap is reduced to a histogram vector of size $1 \times 1 \times n$, with each of its element corresponding to the probability of having faces of certain scale in the image. The histogram vector can be interpreted as a scale-vs-probability histogram. The output feature length is equal to the number of bins in the scale histogram. The histogram is normalized by Sigmoid function so that each element is within $[0, 1]$ and represents probability.

The detailed explanation of scale histogram goes as follows. For a scale histogram with n equally placed bins in log scale, with left edge corresponding to face size s_0 and right edge corresponding to face size s_n, the histogram vector h is defined as:

$$h = [a_1, a_2, a_3, ..., a_n], \tag{1}$$

$$a_i = P(\exists \mathrm{x} | s_i^l \leq \log_2(\mathrm{size}(\mathrm{x})) < s_i^r), \quad (i = 1, 2, ..., n), \tag{2}$$

where d is the width of each bin in base-2 logarithmic scale, $d = (s_n - s_0)/n$, s_i^l and s_i^r are the left and right edge of ith bin, so $s_i^l = s_0 + (i-1)d$ and $s_i^r = s_0 + id$. The x represents a face and $\mathrm{size}(\mathrm{x})$ is the size of face x.

In other words, ith histogram bin corresponds to faces whose sizes are within the following range:

$$[2^{s_0+(i-1)d}, 2^{s_0+id}) \tag{3}$$

With the network structure mentioned above, the global max-pooling layer essentially becomes a response aggregator, which discards location information and picks the maximum response of each histogram bin from all locations. This is a big advantage since it removes the location constraint that presents in standard RPN. The training process

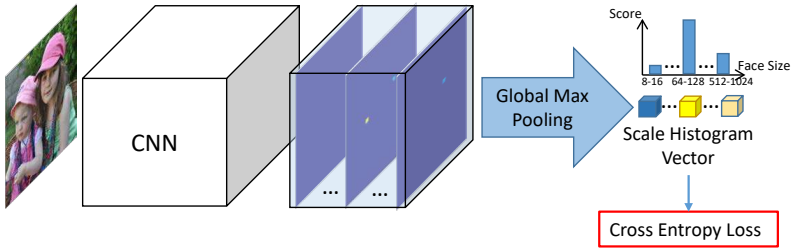


Figure 3. **The construction of Scale Proposal Network.** The SPN is a CNN with a global max pooling layer at its end so that it can produce a fixed-dimensional Scale Histogram Vector disregard the input size and face locations. Each element in Scale Histogram Vector represents the possibility of the presence of faces that have sizes within a certain range. During training, SPN only requires image-level supervision.

of RPN inherently holds the assumption that the response on the classification heatmap should be high if its projected position on input image is close to the center of an object. However, in SPN, the scale estimation response of a face can be at arbitrary location of the heatmap. Ignoring the location information helps the network to selectively learn highly representative features from faces and from context, even if the face is much larger or much smaller than the receptive field of the network. Moreover, this arrangement enables response from multiple face parts to contribute to scale estimation independently. Only the highest response will be selected, thus robustness can be improved. The training strategy for SPN is discussed in Section 4.1.

3.2. Scaling strategy generation

There may be more than one face in an image. To save computation, we hope that faces that are close in size can be covered by detector in a single pass. Thanks to the high-resolution scale estimation generated by SPN, this can be implemented easily by non-maximum suppression (NMS).

When the estimated scale histogram has a large number of bins (e.g. 60 bins between face size of 2^3 and 2^9, with each bin having an interval of $2^{0.1}$), the histogram tends to be noisy. Moreover, the presence of a face in the image usually brings high response to its corresponding bin together with its adjacent bins, which makes it impossible to simply thresholding out the high-response proposals(Figure 4).

To extract useful signal from the histogram, the histogram is smoothed using moving average method with a window of half the length of the detector's covered range. This reduces high-frequency noise and spikes while retaining enough resolution. Then a one-dimensional NMS is applied to extract peaks from the smoothed histogram. The position of the peaks corresponds to face size while the heights of the peaks are regarded as their confidence scores. The window size for NMS is set to be slightly smaller than the cover range of the detector so it will not miss out useful signals (e.g. the scale response generated from another face).

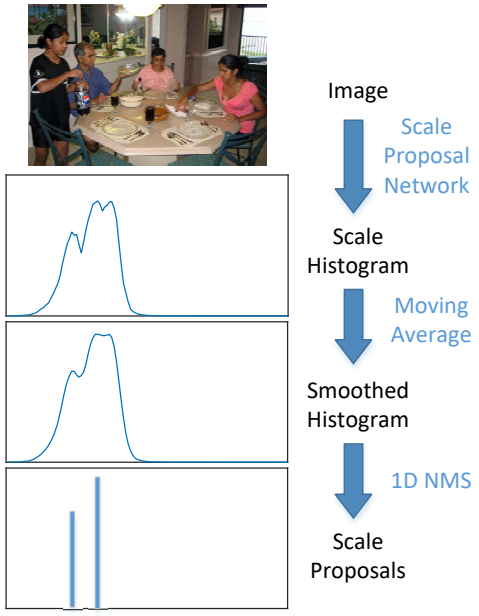


Figure 4. **Process of generating scale proposals from input image.** At first, the Scale Histogram of the image is generated by SPN. Then, the histogram is smoothed by moving average to reduce noise. Finally, non-maximum suppression is performed on smoothed histogram to obtain the final scale proposals. By using NMS, neighboring scale proposals can be efficiently combined to one proposal, which greatly saves computation. After NMS, only a few proposals left.

After NMS there are only a very small number of scale proposals left. Proposals that have a confidence higher than a threshold will be selected as final proposals and images are resized accordingly prior to detection. Although the above-mentioned strategy cannot guarantee to get the minimum number of scales per image, this sub-optimal solution can already achieve high recall rate while keeping number of final proposals small.

3.3. Single-scale RPN

We adopt Region Proposal Network (RPN) as face detector in our pipeline, though any detector should behave similarly. The RPN is a fully convolutional network that has two output branches: classification branch and bounding box regression branch. Each branch may have one or many sub-branches, which handle objects of different scales. The reference box of each sub-branch is called *anchor box*. The detailed information about RPN can be found in[30].

Since the face size variation is already handled in the first stage, in this stage, we only use an RPN with one anchor. The largest detectable face size is set to be twice the size of the smallest detectable face. This configuration is enough to achieve high accuracy while keeping average zooms per image low and the RPN computationally cheap. The RPN we use is called *Single-Scale RPN*, since it has only one anchor and has a narrow face size coverage.

4. Implementation details

4.1. Global supervision

The output histogram vector of SPN is directly supervised by sigmoid cross entropy loss:

$$L = -\frac{1}{N}\sum_{n=1}^{N}[p_n \log \hat{p}_n + (1-p_n)\log(1-\hat{p}_n)], \quad (4)$$

where N denotes the total number of bins, $\hat{p}$ is the histogram vector estimated by the network (normalized by sigmoid function), and p is ground truth histogram vector.

Unlike the training process of RPN, no location information is provided to the SPN during training. What really happens during training is that, in each iteration, the gradient only back-propagates through the location with highest response. Although the SPN is trained from random initialization and the location selection may not always be correct especially in the first few iterations, it will be sticking to right location after thousand iterations' trial and error as long as the training data is sufficient. Owing to the fact that similar feature from irrelevant locations cannot be generalized to all the training samples, the SPN under global supervision will automatically learn features that can easily be generalized, as well as quickly rejecting features that are most likely to cause false scale proposals.

No localization constraints is one of the desirable property of global supervision. When training fully-convolutional detectors or segmentation networks, the location of ground-truth samples are assigned on the heatmap using a set of strategies. These manually-assigned ground truths introduce strong constraints to the training process. One of the examples of those constraints is that, for RPN, the location on the heatmap must correspond to the same location on input image. By removing these constraints and allowing the network to learn to adjust to good features and suitable response formats itself, performance can be improved. One obvious benefit of global supervision is that this enables networks with small receptive fields to generate correct scale proposals for faces several times larger than the receptive field, thus reducing the need of deep networks. The SPN under global supervision can automatically generate scale proposal according to feature-rich facial parts, as shown in Figure 5. Another desirable property of global supervision is its inherent hard-negative mining nature. Global max-pooling always select highest response location for back propagation, thus highest response negative sample will always be selected in each iteration.

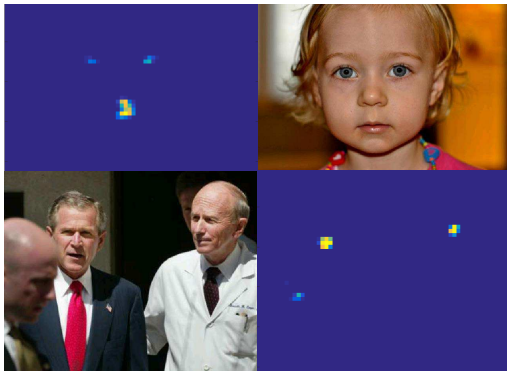


Figure 5. **Scale response map for face larger and smaller than the receptive field of SPN.** The upper-right face is significantly larger than the receptive field. Its corresponding response map on the upper left reveals facial landmark locations, which suggests that even if the face is larger that receptive field, SPN can still correctly recall it according to parts of faces. Also, although we don't supervise the locations of faces at SPN stage, the response map before global max pooling can still reveals some location information.

Although scale proposals can also be generated by a more complex, wide-range and single view detector such as a multi-anchor RPN, its speed cannot match SPN.

4.2. Ground truth preparation

Definition of bounding box. The size of faces that used for generating ground truth histogram is defined to be the side length of the square bounding box. One problem regarding to this is that how to define the bounding box of a face and keep it consistent throughout the training samples. Noise in bounding box annotation can impair the performance of scale proposal network. Also, any misalignment of the bounding box between two stages can severely affect the performance.

However, manual labeling of face bounding boxes is a very subjective task and prone to noise. So we prefer to derive bounding boxes from the more objectively-labeled 5-

point facial landmark annotations using the transformation described below. Note that the bounding boxs we define are always square.

$$\begin{bmatrix} x_i^b \\ y_i^b \\ s_i^b \end{bmatrix} = \begin{bmatrix} \text{mean}(x_i^{l1}, x_i^{l2}, x_i^{l3}, x_i^{l4}, x_i^{l5}) + o_x \\ \text{mean}(y_i^{l1}, y_i^{l2}, y_i^{l3}, y_i^{l4}, y_i^{l5}) + o_y \\ \text{std}(y_i^{l1}, y_i^{l2}, y_i^{l3}, y_i^{l4}, y_i^{l5}) * o_s \end{bmatrix} \quad (5)$$

where ith landmark annotation (x_i^{lk}, y_i^{lk}) corresponds to the location of *left eye center*, *right eye center*, *nose*, *left mouth corner* and *right mouth corner* for $k = 1, 2, ..., 5$ respectively. The corresponding bounding box is defined as (x_i^b, y_i^b, s_i^b), where (x_i^b, y_i^b) is center location of the box and s_i^b is its side length. o_x, o_y and o_s are offset parameters that are shared among all samples.

Ground truth generation. One of the most intuitive way to derive ground truth histogram from face sizes is by simply treating the histogram as multiple binary classifiers, setting the corresponding bin for each face to positive. But such nearest-neighbor approach is very prone to annotation noise even if the less-noisy annotation protocol is used. Though we managed to make nearest neighbor approach work on very large binning interval (e.g. 2^1 bin width in log scale), its performance drops rapidly with the reducing of binning interval and can even prevent SPN from converging.

For the reasons above, we adopt a more stable approach for generating ground-truth histogram vector. For each ground truth face size s, we assign a Gaussian function:

$$f(x) = e^{-\frac{(x - \log_2 s)^2}{2\sigma^2}}. \quad (6)$$

The target value for ith bin is sampled from $f(x)$:

$$a_i = f((s_i^l + s_i^r)/2). \quad (7)$$

By doing so, the model is more immune to the noise introduced by imperfect ground truth since the Gaussian function provides a soft boundary. The selection of σ mainly depends on the error distribution of ground truth and the window size of the detector. In our case, we use $\sigma = 0.4$ in all the experiments.

If more than one faces appear in a single image, the ground truth histogram is generated by doing element-wise maximum over the ground-truth histograms of each individual faces, which is coherent to the use of max-pooling layer.

4.3. Receptive field problem

Like all the fully-convolutional networks, in SPN the heatmap before global max-pooling has a limited receptive field. But unlike RPN, this receptive field limitation does not prevent the network from accurately estimating the size of faces that are many times larger than the receptive field. This is because some sub-regions from a large face contain enough information to inference the size of the whole face, as is described in Section 4.1 and illustrated in Figure 5. Though the network we use has a receptive of 108×108 pixels, it can obtain sensible estimation of face sizes as large as 512×512 pixels.

4.4. Training RPN

The training of single scale RPN is straightforward. All the faces within the detectable range are regarded as positive samples and the faces outside the detectable range belong to negative samples.

5. Experiments

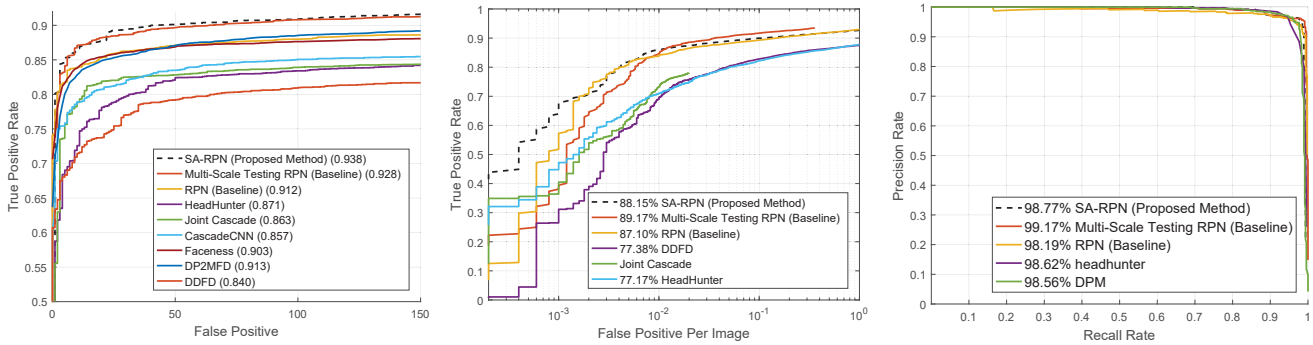
In this section, we will evaluate the performance of our pipeline on three face detection datasets: FDDB [15], AFW [27] and MALF [38]. We will also provide theoretical data for computational cost analysis.

To make the experiment result comparable, we train both our model and other models under the same condition, using the same training data and the same network. The performance curves of our method along with several previous methods on each dataset are reported. Computational costs and time consumptions are listed and investigated. Extensive ablation experiments are conducted to validate the effectiveness of doing scale proposal prior to detection. In addition to overall performance, the performance of SPN is also separately evaluated.

Training data overview. For training samples, we collect about 700K images from the Internet, of which 350K contain faces. To improve the diversity of faces, we also include images from Annotated Facial Landmarks in the Wild (AFLW) dataset [18]. All the above-mentioned images are exclusive from FDDB, MALF and AFW datasets. For negative samples, we use both images from the Internet and COCO [23] dataset, excluding images with people

All the easily-distinguished faces are labeled with 5 facial landmark points (left eye, right eye, nose, left mouth corner, right mouth corner) and bounding boxes were derived from the landmarks using the transformation described in Section 4.2. Faces and regions that were too hard to annotate were marked as ignoring regions. In the training of SPN, these regions will be filled with random colors before being fed into the network. In the training of RPNs, neither positive nor negative sample are drawn from these regions.

Network structure. Both SPN and RPN use a truncated version of GoogleNet, down to inception-3b. However, for SPN, the output channel of each convolution layer (within GoogleNet) is cut to 1/4 to further reduce computation. Table 1 shows the computational cost of each network. Batch normalization [14] is used for both networks during training.



(a) FDDB (b) MALF - Whole (c) AFW

Figure 6. **Comparison with previous methods on FDDB, MALF and AFW datasets.** The numeric metrics shown in figures are: (a) recall rate at 1000 false positives; (b) MALF proprietary "mean recall rate"; (c) average precision. Best viewed in color.

Layer	MFLOPs	
	Full GoogleNet	1/4 GoogleNet
conv1	118	30
conv2	360	22
inception(3a)	171	11
inception(3b)	203	13
feature128	289	72
Total	1141	148

Table 1. Architectures and computation analysis for Scale Proposal Network (1/4 GoogleNet) and Region Proposal network (full GoogleNet). All the data assume an input size of $224 \times 224 \times 3$. Batch Normalization layers are not shown and can be removed at test time. Auxiliary convolution layers are not shown for clarity.

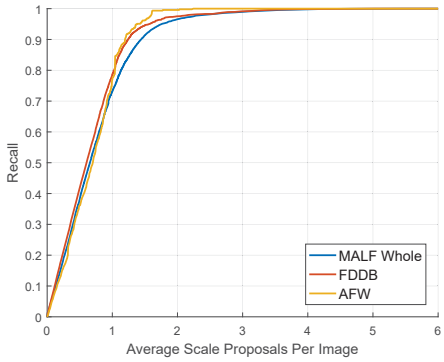


Figure 7. Recall-Average Scale Proposals Per Image curves of SPN on FDDB, MALF and AFW dataset.

Multi-scale testing RPN. Each image is resampled to have long sides of $1414 \times 2^k (k = 0, -1, -2, -3, -4 - 5)$. They are detected for faces respectively using the same RPN in our method. These intermediate results are combined to form the final result.

Single view RPN. A standard RPN that has 6 anchors to cover faces within the range of 8 to 512. The input image is always resized to have a long side length of 1414 pixels.

5.1. Evaluation of scale proposal stage

In this section, we first evaluate the performance of SPN separately from the whole pipeline. Since the scale proposal stage and detection stage essentially form a cascaded structure, any face that is missed by this stage will not be recalled by the detector. So, it is crucial to make sure that the scale proposal stage is not the performance bottleneck of the whole pipeline. We expect a high recall from this stage while keeping average resizes per image low.

The SPN can handle faces within the range of $(2^3, 2^9)$, with a resolution of $2^{0.1}$. When testing, every image is resized so that its long side has a length of 448 pixels. A face is recalled only if its ground truth face size falls into the detectable range of detector (in our case 36-72 pixels) after being scaled according to the proposal.

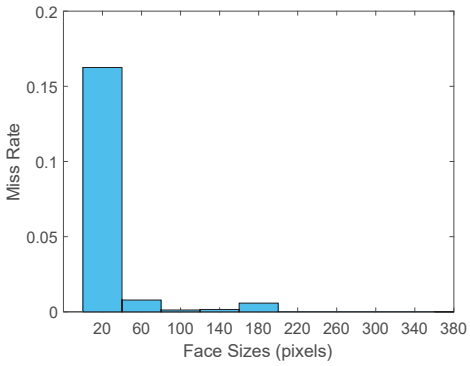


Figure 8. **The miss rate of SPN versus face size.** Miss rate is calculated as the proportion of faces not recalled in each bin. Evaluated on FDDB dataset.

Method	FDDB		MALF Whole		AFW	
	MFLOPs	Time (ms)	MFLOPs	Time (ms)	MFLOPs	Time (ms)
SA-RPN	441 + 2704	5.21 + 60.23	437 + 8854	11.87 + 190.15	432 + 6383	13.17 + 153.66
MST-RPN	50240	754.75	49807	981.47	49139	427.32
RPN	33846	588.37	33554	549.68	33104	360.20

Table 2. Comparison of Scale-aware RPN (SA-RPN), multi-scale testing RPN (MST-RPN) and standard single-shot multi-anchor RPN (RPN) on computation requirements. The reported data are the average result for a single image.

We report the performance of SPN using Recall-Average Scale Proposals Per Image curves, as shown in Figure 7. We also analyze the SPN's performance on different face sizes. Figure 8 shows that most failures come from small faces while faces larger than receptive field can be handled well.

5.2. Overall performance

We benchmark our method on FDDB, MALF and AFW following the evaluation procedure provide by each dataset. For scale proposals, we discard proposals that have a confidence lower than a fixed threshold. Figure 6 displays the performance of our method alongside with our baseline methods (Multi-Scale RPN, RPN) and state-of-the-art algorithms. Our method achieves best performance on FDDB and best accuracy in high confidence regions on MALF. The MALF dataset contains many challenging faces, having large face size diversity and a high proportion of small faces, which affect the recall rate of SPN and reduce the maximum possible recall of SAFD pipeline.

Though the chart does reveal that on MALF and AFW the SPN results in a drop on recall for low quality faces, our SAFD pipeline still outperforms previous methods. Moreover, the SA-RPN is several times faster than the slow but high recall multi-scale testing baseline and has fewer high-confidence false-positive detections. For multi-scale testing method, every image is detected in 6 different scales. Scale estimation reduces the average detection passes of each image, which can reduce the probability of getting false positives and improve speed.

The chart also shows that under the same condition, a single-shot multi-anchor RPN has significantly lower performance than SA-RPN and multi-scale testing RPN, which coincides with our expectation. Apart from the fact that such a RPN needs to fit to more diversified training data, the network has a receptive field of only 107 pixels, making it extremely hard to detect large faces correctly.

5.3. Computational cost analysis

In this section, we analyze the computational cost of SA-RPN along with baseline methods. Table 2 shows the theoretical average FLOPs per image as well as empirical testing time on each database. Since the theoretical computation of CNN is proportional to the input image size (when taking padding area into account), the total FLOPs can easily be calculated by accumulating the input image size (in pixels) of CNNs on each forwarding pass. The test times contain system overheads so they are for reference purpose only.

Unlike multi-scale testing RPN and standard RPN which has a fixed computational requirement on the same input size, the computational cost of our model is largely dependent on the content of images, which reveals in Table 2 as large average FLOPs variance between datasets. But even on the worst-performing MALF dataset, our Scale Aware RPN can still outperforms baseline methods by a large margin in terms of speed.

6. Conclusion

In this paper, we proposed SAFD, a two-stage face detection pipeline. It contains a scale proposal stage which automatically normalizes face sizes prior to detection. This enables computationally cheap single-scale face detector to handle large scale variation without using computationally expensive multi-scale pyramid testing. The SPN is designed to generate scale proposals. Our method achieves state-of-the-art performance on AFW, FDDB and MALF. The performance is similar to multi-scale testing based detectors but requires much less computation. The proposed method can also be applied to general object detection problems. Moreover, the SPN is essentially a weakly-supervised detector, which could be used to generate coarse region proposals and further improves speed. SPN can also share convolution layers with RPN to further reduce model size.

References

[1] S. Bell, C. L. Zitnick, K. Bala, and R. Girshick. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. *Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[2] L. Bourdev and J. Brandt. Robust object detection via soft cascade. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 236–243. IEEE, 2005. 2

[3] Z. Cai, Q. Fan, R. Feris, and N. Vasconcelos. A unified multi-scale deep convolutional neural network for fast object detection. In *ECCV*, 2016. 2

[4] D. Chen, G. Hua, F. Wen, and J. Sun. Supervised transformer network for efficient face detection. In *European Conference on Computer Vision*, pages 122–138. Springer, 2016. 1, 2

[5] j. dai, Y. Li, K. He, and J. Sun. R-fcn: Object detection via region-based fully convolutional networks. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 379–387. Curran Associates, Inc., 2016. 2

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005. 2

[7] P. Dollár, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(8):1532–1545, 2014. 2

[8] S. S. Farfade, M. Saberian, and L.-J. Li. Multi-view face detection using deep convolutional neural networks. *arXiv preprint arXiv:1502.02766*, 2015. 2

[9] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, 2010. 1, 2

[10] G. Ghiasi and C. C. Fowlkes. Occlusion coherence: Detecting and localizing occluded faces. *arXiv preprint arXiv:1506.08347*, 2015. 2

[11] C. Huang, H. Ai, Y. Li, and S. Lao. Vector boosting for rotation invariant multi-view face detection. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 1, pages 446–453. IEEE, 2005. 2

[12] C. Huang, H. Ai, Y. Li, and S. Lao. High-performance rotation invariant multiview face detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(4):671–686, 2007. 2

[13] L. Huang, Y. Yang, Y. Deng, and Y. Yu. Densebox: Unifying landmark localization with end to end object detection. *arXiv preprint arXiv:1509.04874*, 2015. 1, 2

[14] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 6

[15] V. Jain and E. Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. Technical Report UM-CS-2010-009, University of Massachusetts, Amherst, 2010. 6

[16] V. Jain and E. G. Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. *UMass Amherst Technical Report*, 2010. 2

[17] M. Jones and P. Viola. Fast multi-view face detection. *Mitsubishi Electric Research Lab TR-20003-96*, 3:14, 2003. 2

[18] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 2144–2151. IEEE, 2011. 6

[19] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua. A convolutional neural network cascade for face detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5325–5334, 2015. 2

[20] J. Li and Y. Zhang. Learning surf cascade for fast and accurate object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3468–3475, 2013. 2

[21] S. Z. Li and Z. Zhang. Floatboost learning and statistical face detection. *IEEE Transactions on pattern analysis and machine intelligence*, 26(9):1112–1123, 2004. 2

[22] S. Z. Li, L. Zhu, Z. Zhang, A. Blake, H. Zhang, and H. Shum. Statistical learning of multi-view face detection. In *ECCV 2002*, pages 67–81. Springer, 2002. 2

[23] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollr, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, Zrich, 2014. Oral. 6

[24] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *ECCV*, 2016. 2

[25] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool. Face detection without bells and whistles. In *Computer Vision–ECCV 2014*, pages 720–735. Springer, 2014. 2

[26] H. Qin, J. Yan, X. Li, and X. Hu. Joint training of cascaded cnn for face detection. In *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, 2016. 2

[27] D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR '12, pages 2879–2886, Washington, DC, USA, 2012. IEEE Computer Society. 6

[28] R. Ranjan, V. M. Patel, and R. Chellappa. A deep pyramid deformable part model for face detection. In *BTAS*, pages 1–8. IEEE, 2015. 2

[29] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. *arXiv preprint arXiv:1506.02640*, 2015. 2

[30] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 91–99. Curran Associates, Inc., 2015. 2, 5

[31] H. Rowley, S. Baluja, T. Kanade, et al. Neural network-based face detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(1):23–38, 1998. 2

[32] H. Rowley, S. Baluja, T. Kanade, et al. Rotation invariant neural network-based face detection. In *Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference on*, pages 38–44. IEEE, 1998. 2

[33] R. Vaillant, C. Monrocq, and Y. Le Cun. Original approach for the localisation of objects in images. *IEE Proceedings-Vision, Image and Signal Processing*, 141(4):245–250, 1994. 1, 2

[34] P. Viola and M. J. Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004. 1, 2

[35] J. Yan, Z. Lei, L. Wen, and S. Li. The fastest deformable part model for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2497–2504, 2014. 2

[36] B. Yang, J. Yan, Z. Lei, and S. Z. Li. Aggregate channel features for multi-view face detection. In *Biometrics (IJCB), 2014 IEEE International Joint Conference on*, pages 1–8. IEEE, 2014. 2

[37] B. Yang, J. Yan, Z. Lei, and S. Z. Li. Convolutional channel features for pedestrian, face and edge detection. *arXiv preprint arXiv:1504.07339*, 2015. 2

[38] B. Yang, J. Yan, Z. Lei, and S. Z. Li. Fine-grained evaluation on face detection in the wild. In *Automatic Face and Gesture Recognition (FG), 11th IEEE International Conference on*. IEEE, 2015. 6

[39] S. Yang, P. Luo, C.-C. Loy, and X. Tang. From facial parts responses to face detection: A deep learning approach. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015. 1, 2

[40] C. Zhang, J. C. Platt, and P. A. Viola. Multiple instance boosting for object detection. In *Advances in neural information processing systems*, pages 1417–1424, 2005. 2

[41] L. Zhang, R. Chu, S. Xiang, S. Liao, and S. Z. Li. Face detection based on multi-block lbp representation. In *Advances in biometrics*, pages 11–18. Springer, 2007. 2

[42] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2879–2886. IEEE, 2012. 2