Multi-Scale Fully Convolutional Network for Face Detection in the Wild

Yancheng Bai *[1,2] and Bernard Ghanem †[1]

[1]King Abdullah University of Science and Technology (KAUST), Saudi Arabia
[2]Institute of Software, Chinese Academy of Science, Beijing, China

Abstract

Face detection is a classical problem in computer vision. It is still a difficult task due to many nuisances that naturally occur in the wild, including extreme pose, exaggerated expressions, significant illumination variations and severe occlusion. In this paper, we propose a multi-scale fully convolutional network (MS-FCN) for face detection. To reduce computation, the intermediate convolutional feature maps (conv) are shared by every scale model. We up-sample and down-sample the final conv map to approximate K levels of a feature pyramid, leading to a wide range of face scales that can be detected. At each feature pyramid level, a FCN is trained end-to-end to deal with faces in a small range of scale change. Because of the up-sampling, our method can detect very small faces (10×10 *pixels). We test our MS-FCN detector on four public face detection benchmarks, including FDDB, WIDER FACE, AFW and PASCAL FACE. Extensive experiments show that our detector outperforms state-of-the-art methods on all these datasets in general and by a substantial margin on the most challenging among them (*e.g. *WIDER FACE Hard). Also, MS-FCN runs at* 23 *FPS on a GPU for images of size* 640×480 *with no assumption on the minimum detectable face size.*

1. Introduction

Face detection is a very active research field and has attracted special attention in the computer vision community, primarily because of its many real-world applications including facial expression recognition, face recognition, face parsing, and human computer interaction (HCI). During the past decade, great progress has been made in developing accurate and efficient face detection methods, albeit for mostly constrained scenarios. However, it remains a difficult task in the wild conditions, which prevail in natural manifestations of the face detection problem (*e.g.* detecting faces in surveillance videos or crowds). This difficulty primarily stems from several challenges that need to be overcome, including extreme pose, exaggerated expressions, significant variations in illumination, and partial or severe occlusion as shown in Figure 1. A thorough review of face detection methods can be found in a recent survey [43].

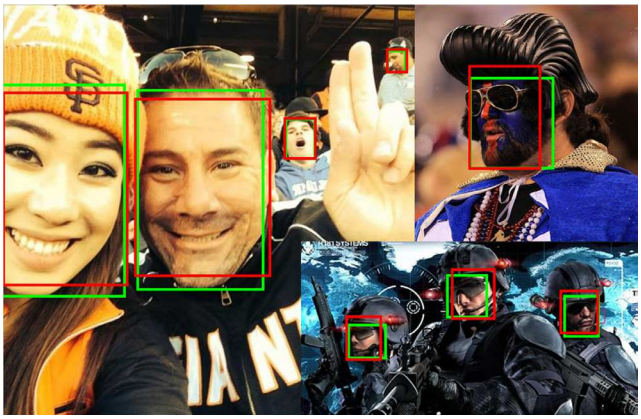


Figure 1. Example detection results of the proposed MS-FCN face detector. MS-FCN deals with faces with exaggerated expressions, large scale variations, severe makeup, and occlusion. Green and red bounding boxes denote ground-truth annotations and MS-FCN detection results, respectively.

Traditional face detection methods [33, 4, 35] are based on sliding window search and hand-crafted features. Typically, one single scale model is learned and slides on feature maps (*e.g.* HOG [5] or LBP [24]) to detect face instances in an image. To deal with scale variations, the search has to be done across different levels of an image pyramid built from the original image. This constitutes the main computational bottleneck of many modern face detectors. Also, because of the limited representation power of hand-crafted features, these traditional face detectors register subpar detection accuracy when applied to more realistic unconstrained scenarios available in challenging and recently compiled face benchmarks (*e.g.* WIDER FACE) [33, 49, 19, 35, 4].

*yancheng.bai@kaust.edu.sa, yancheng@iscas.ac.cn
†bernard.ghanem@kaust.edu.sa

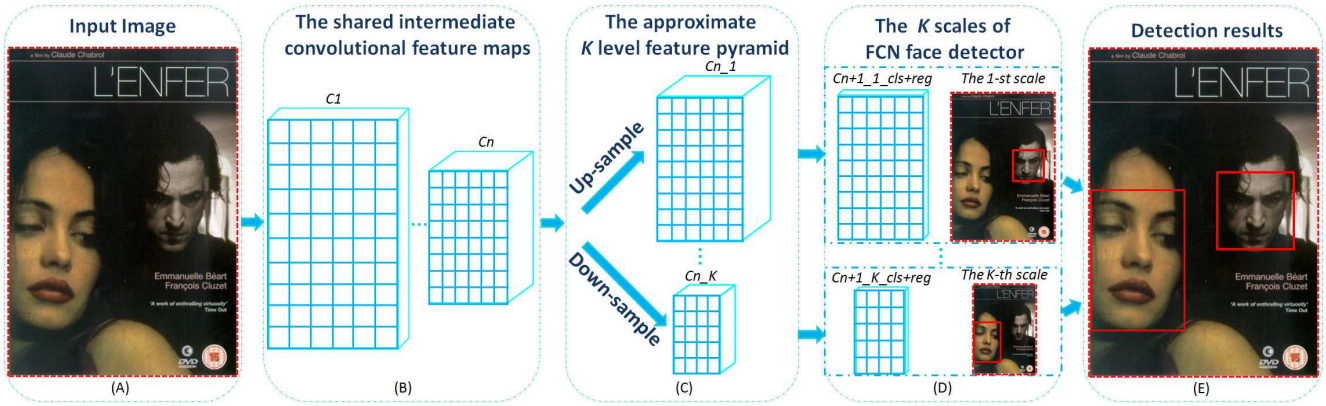


Figure 2. The pipeline of the proposed multi-scale FCN face detector. (A) One image is fed into the network. (B) After several layers of convolution, we obtain the final convolutional (*conv*) feature map. (C) We up-sample and down-sample the final *conv* feature map to approximate K levels of a feature pyramid. (D) A separate FCN is applied on each level of the pyramid and outputs classification (*cls*) and regression (*reg*) results for a range of face scales. (E) The *cls* and *reg* outputs of all levels are converted into scored bounding boxes, followed by non-maximum suppression (NMS) to obtain the final multi-scale detection results.

In recent years, superior performance in image classification and object/scene recognition has been achieved as a result of the resurgence of deep neural networks including CNNs [17, 30, 47]. Similarly, object detection has improved significantly due to richer appearance and spatial representations that are learned from image data by CNNs [9]. Among CNN-based object detectors, the Region-based CNN (RCNN) [9] can be considered a milestone for detection and has achieved state-of-the-art detection accuracy. There are two stages in the pipeline of RCNN: firstly, a proposal algorithm (*e.g.* selective search [32] or Edgebox [50]), finds candidate object locations; secondly, a deep CNN classifier classifies each of these candidates. By learning both these stages end-to-end, Faster RCNN [27] has recently registered further improvement in both detection performance and computational efficiency. In fact, it has become the *de facto* framework for general object detection.

Inspired by the success of Faster RCNN, several recent works [14, 34, 48] have emerged to apply this framework to detect faces and achieve impressive performance on the widely used FDDB benchmark [12]. However, performance drops dramatically on the more challenging WIDER FACE [41]. The main reason is that the resolution of the feature maps generated by these methods is insufficient for handling small face instances [46, 2]. Furthermore, the second stage classifier in Faster RCNN might degrade the detection accuracy due to the low resolution of its *conv* feature maps as pointed out in [46]. To overcome this problem, detectors based on Faster RCNN need to up-sample input images during training and testing, which inevitably increases memory and computation costs.

Zhang *et al*. [46] claim that feature maps with higher resolution and image pyramids are two factors that are effective for detecting small objects. Inspired by this observation, we propose a multi-scale fully convolutional network (MS-FCN) for face detection, as illustrated in Figure 2. In our detector, the intermediate *conv* feature maps are shared by every scale (Figure 2(B)). We up-sample and down-sample the final feature map to approximate K levels of a feature pyramid (Figure 2(C)). By sharing FCN layers and searching the feature maps in the pyramid (spatially much smaller than the input image), we significantly reduce the detection runtime. This reduction is especially obvious when compared to state-of-the-art detectors that up-sample and down-sample the input image and apply their detection method on each image scale separately. In Figure 2(C), on each level of the feature pyramid, a FCN is trained end-to-end to simultaneously classify whether each spatial region (at a range of scales) is a face or not and regress the resulting bounding box. Each FCN only needs to handle faces in a small scale range rather than handling all scales in one network. Interestingly, using up-sampled feature maps enables our method to detect very small faces (10×10 pixels).

Contributions. This paper makes three main contributions. **(1)** A new architecture for a multi-scale FCN is proposed, where all scales can be trained at the same time. Essentially, the MS-FCN detector consists of a set of face detectors, each of which handles faces in a small range of scale change. **(2)** A strategy is proposed to approximate K levels of a feature pyramid by sharing the same intermediate *conv* feature maps, followed by up-sampling and down-sampling, thus, effectively reducing the computation cost. **(3)** The MS-FCN detector outperforms state-of-the-art methods on these four public benchmarks, where the most impressive improvement occurs in the most challenging among them. MS-FCN is also computationally efficient, since runs at 23 FPS on a GPU for 640×480 images and with no assumption on the minimum detectable face size.

2. Related Work

2.1. Handcrafted Feature Based Face Detection

Being a classic topic, many face detection systems have been proposed during the past decade or so, beginning with very seminal work [28, 33]. Traditional face detectors can be generally categorized into the following two classes.

Boosting Cascade Detectors: The boosting cascade framework [33] proposed by Viola and Jones (VJ) is a seminal work for face detection. Haar features, the integral image, and the attentional cascade structure are the three ingredients for the success and ubiquity of the VJ framework. However, the simple Haar features have limited representation, which leads to poor performance in uncontrolled environments. HOG [5], SURF [19] and other sophisticated features [39] can be exploited to enrich the capacity of feature representation and improve detection accuracy.

DPM-based Detectors: The deformable parts model (DPM) [36, 23, 35] is another traditional paradigm for object detection. Detectors based on DPM learn root filters, part filters, and their spatial relationships via a latent support vector machine (LSVM), making them more robust to occlusion. This framework was applied successfully to face detection in several works demonstrating state-of-the-art performance at the time [36, 23, 35].

However, most of the detection systems above only train a single scale model that is applied to each level of a feature pyramid, thus, increasing the computational cost drastically, especially when complicated features are used.

2.2. CNN-Based Face Detectors

Recently, with the break-through results of CNNs for image classification and scene recognition [17, 30, 47], generic object detection based on CNNs have been proposed [9, 8, 27]. These methods share a similar *two-stage* pipeline (proposals followed by classification), which is now the *de facto* standard [9, 8, 27].

Two-stage framework: Several CNN-based face detectors have been recently proposed [6, 18, 40, 11]. Inspired by the boosting-based algorithms, Li *et al*. [18] propose a cascaded architecture called CascadeCNN for real-world face detection. A multi-task variant of CascadeCNN for face detection and alignment is proposed in [45]. Every stage of CascadeCNN [18, 45] needs to be designed carefully and is trained separately (not end-to-end). To overcome this problem, a joint training CascadeCNN is proposed in [25].

Facial attribute information can also help in detecting faces [40, 3, 31]. Yang *et al*. [40] demonstrate that facial attribute CNN models can be applied to find proposals that are further processed by another CNN model [17]. Chen *et al*. [3] use predicted facial landmarks and a supervised transformer network (STN) to learn the optimal canonical pose to best differentiate face/non-face patterns. Li *et al*. [20] compute face proposals by estimating facial keypoints and a 3D transformation. Compared to these methods, our detector is only trained on 2D bounding box information, which requires much less labeling effort than facial points/attributes. Moreover, the performance of the aforementioned detectors can drop dramatically, since finding facial points on low resolution faces remains challenging.

Compared to these two-stage detectors, MS-FCN only uses one single deep neural network and achieves top performance. This demonstrates that a single fully convolutional network, if designed carefully, can detect faces at different scales accurately and efficiently.

One-stage framework: In [11], Huang *et al*. propose an end-to-end FCN framework, called DenseBox, for face detection. Also, Bai *et al*. [38] propose a multi-scale architecture for face detection. In [42], the IoU loss is proposed to learn better bounding box regression. However, UnitBox [42] is only tested on the FDDB dataset.

Most of the methods [18, 45, 25, 3, 11, 38] mentioned above have to construct image pyramids to detect faces at different scales, which is time-consuming. In comparison, MS-FCN up-samples and down-samples the final *conv* feature map to approximate a feature pyramid and learns one specific scale model at each level of the pyramid to deal with faces at different scales. Therefore, our architecture is considerably more efficient.

2.3. Multi-Scale Generic Detectors

To detect small objects in images, the works of [1, 44] employ intermediate *conv* feature maps of different layers for more accurate representations of objects. SSD [21] uses multi-scale features to learn a generic object detector, in which there are some common design aspects between their CNN architecture and ours. However, SSD has to resize images to a specific scale and is not reliable at detecting small objects. In [2], multi-scale proposal networks are trained on intermediate *conv* feature maps for detecting proposals of small objects. Compared to this detector, our method detects small objects using deep feature maps with high resolution, which is more discriminative [17, 10].

3. Multi-Scale FCN Detection System

In this section, we will give a detailed description on the multi-scale FCN detection system, including the deep architecture, the multi-scale training and implementation details.

3.1. Architecture

As illustrated in Figure 2, the whole detection system consists of four components. **(i)** The first component is the shared intermediate backbone network, which can be of any typical architecture like AlexNet [17], VGGNet [30] or

ResNet [10]. After images are passed through the backbone network, the final *conv* map is generated. **(ii)** The second component creates the approximate feature pyramid. Up-sampling and down-sampling operate on the final *conv* map to produce K-level feature maps with different resolution. In our current implementation, deconvolution [22] and max pooling are used for up-sampling and down-sampling. Using this pyramid can save a substantial amount of computational cost compared to constructing it explicitly. **(iii)** For each level of the pyramid, there is one FCN to deal with faces at different scales between two consecutive levels. **(iv)** We convert the *reg* and *cls* outputs of every scale to scored bounding boxes, apply non-maximum suppression (NMS) to those with confidence above a predefined threshold, and obtain the final detection results.

3.2. Multi-Scale Multi-Task Training

During training, the parameters $\mathbf{W}$ of our multi-scale FCN detector are learned from the training image set $S = \{(X_i, Y_i)\}_{i=1}^{N}$, where X_i is one training example, and $Y_i = (y_i^*, \mathbf{b}_i^*)$ is the corresponding combination of its class label $y_i^* \in \{0, 1\}$ and bounding box coordinates $\mathbf{b}_i^* = [b_x^*, b_y^*, b_w^*, b_h^*]_i$. The parameters $\mathbf{W}$ are learned by minimizing the following multi-scale multi-task problem:

$$\mathcal{L}(\mathbf{W}) = \sum_{k=1}^{K} \alpha_k \sum_{i \in S^k} l^k(X_i, Y_i|\mathbf{W}) \tag{1}$$

where K is the number of scales, α_k the weight of loss l^k, which balances the importance of models at different scales. In our experiments, each α_k is set to 1, which means that all K scale models show the same importance. $S = \{S_1, S_2, \cdots, S_K\}$ and S_k denotes the subset containing the training examples for the k-th scale model. Note that only the training samples in the subset S_k contributes to the loss of the k-th FCN model. Inspired by the great success of joint learning of classification and bounding box regression [27], the loss of each FCN model combines these two objectives and is defined as following:

$$l^k(X_i, Y_i|\mathbf{W}) = l_{cls}(y_i, y_i^*) + \lambda_k y_i^* l_{loc}(\mathbf{b}_i, \mathbf{b}_i^*) \tag{2}$$

where y_i, $\mathbf{b}_i$, l_{cls} and l_{loc} denote the predicted score, parameterized coordinates of the predicted bounding box, the loss for classification and regression, respectively. The term $\lambda_k y_i^* l_{loc}(\mathbf{b}_i, \mathbf{b}_i^*)$ means that the regression loss is activated only for positive samples ($y_{ki}^* = 1$). λ_k is the balancing parameter and is set to 2, thus, leading to better localization of objects. The softmax loss is used as the classification loss, which can be defined as follows:

$$l_{cls}(y_i, y_i^*) = y_i^* \log(y_i) + (1 - y_i^*) \log(1 - y_i) \tag{3}$$

Inspired by [8], the robust $L1$ loss is used for regression:

$$l_{loc}(\mathbf{b}_i, \mathbf{b}_i^*) = \sum_{j \in \{x,y,w,h\}} \text{smooth}_{l_1}(\mathbf{b}_i^* - \mathbf{b}_i)_j, \tag{4}$$

where

$$\text{smooth}_{l_1}(x) = \begin{cases} \frac{x^2}{2} & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \tag{5}$$

The robust loss l_{loc} is less sensitive to outliers than the L_2 loss. With these definitions, the optimal parameters $\mathbf{W}^*$ can be learnt by stochastic gradient descent.

3.3. Implementation Details

Training data. The WIDER FACE dataset [41] is used to train our MS-FCN detector. During training, we randomly sample one image per batch from the training set. To fit it in GPU memory, the image is resized by the ratio $1024/\max(w, h)$, where w and h are its width and height, respectively. The candidate box is assigned with a positive label, if its intersection over union (IoU) overlap with any ground-truth bounding box is larger than 0.55; otherwise, it is negative if the maximum IoU is less than 0.35. To apply data augmentation, each image is horizontally flipped with a probability of 0.5. No other augmentation is used.

K scale models. As mentioned earlier, ResNet-50 is applied as the backbone architecture. To increase the resolution of the final *conv* feature map, the stride operations in the first conv-5 block is modified from 2 to 1 as done in [15, 10], which reduces the effective stride from 32 to 16 pixels. To detect faces with low resolution, we use the deconvolution operation [22] to up-sample the final *conv* feature map. The up-sampling ratio is 2, therefore, the stride of the up-sampled feature map is 8. To detect faces with high resolution, we use max pooling operation to down-sample the final feature map twice with different strides $(2, 4)$. Therefore, the strides of each down-sampled feature map are 32 and 64, respectively.

Finally, we generate the $K = 4$ scales or levels of the feature pyramid. The strides of these levels are $\{8, 16, 32, 64\}$, respectively. At each level, we train one FCN to detect faces within specific scale variations. At the up-sampled level, we use 7 candidates of different scales with one single aspect ratio of $1:1$, starting from 10 pixels height with a scaling stride of 1.25. Therefore, the FCN face detector at this level can deal with faces within a range of scales $10-48$ pixels in height and width. For the other three levels, there are 5 anchors of different scales in each FCN detector. And the scale ranges of each FCN are $48 - 120$, $120 - 300$, $300 - 745$ pixels, respectively. Therefore, our MS-FCN detector can handle a very wide range of face scales in an input image in a single shot. More importantly, the number of anchors and scales can be varied according to training statistics, which makes MS-FCN more flexible.

Hyper-parameters. The weights of the filters of new layers are initialized with a zero-mean Gaussian distribution with standard deviation 0.01. Biases are initialized at 0.1.

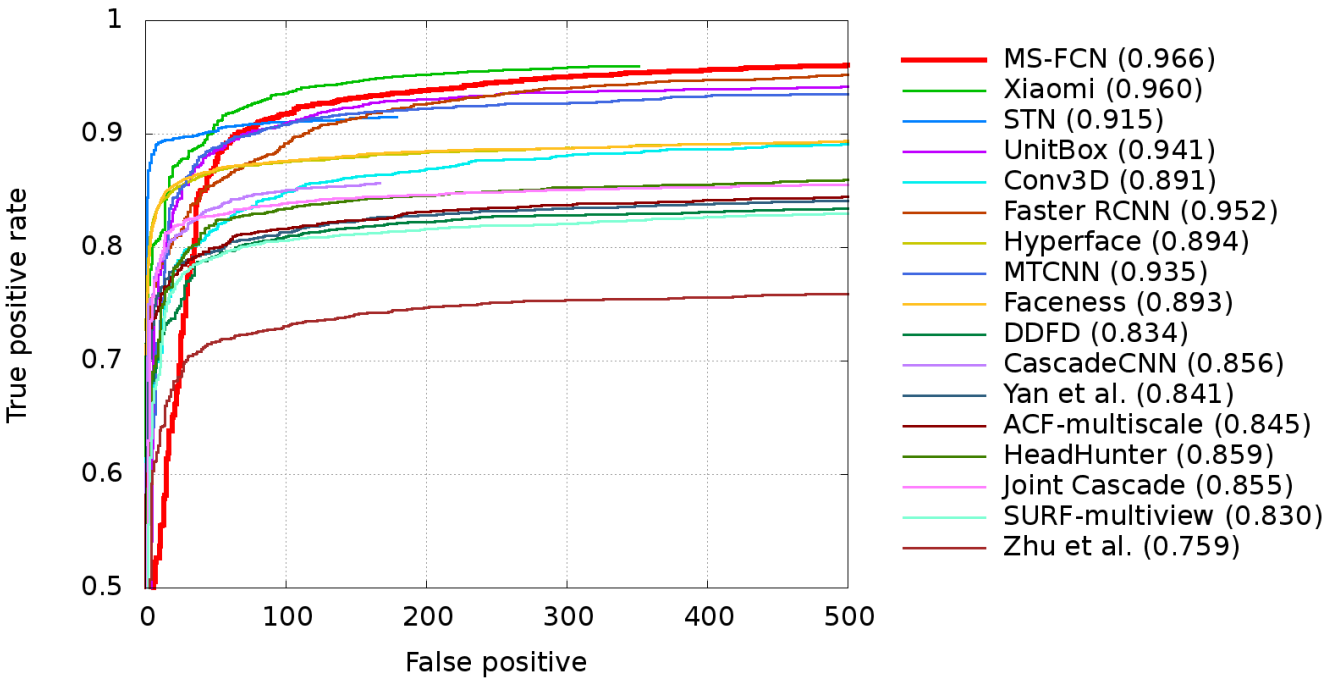


Figure 3. On the FDDB dataset, we compare our MS-FCN detector against many state-of-the-art methods: Xiaomi [34], STN [3], UnitBox [42], Conv3D [20], Faster RCNN [14], Hyperface [26], MTCNN [45], Faceness [40], DDFD [6], CascadeCNN [18], Yan *et al*. [35], ACF-multiscale [39], HeadHunter [23], Joint Cascade [4], SURF-multiview [19] and Zhu *et al*. [49]. The precision rate with 500 false positives is reported in the legend. The figure is best viewed in color.

All other layers are initialized using a model pre-trained on ImageNet. The mini-batch size is set as 128 for each FCN model. The initial learning rate is set as 0.001 and then reduced by a factor of 10 after every $40k$ mini-batches. The training process is terminated after a maximum of $80k$ iterations. A momentum of 0.9 and a weight decay of 0.0005 is applied. Our system is implemented in Caffe [13] and its source code will be made publicly available.

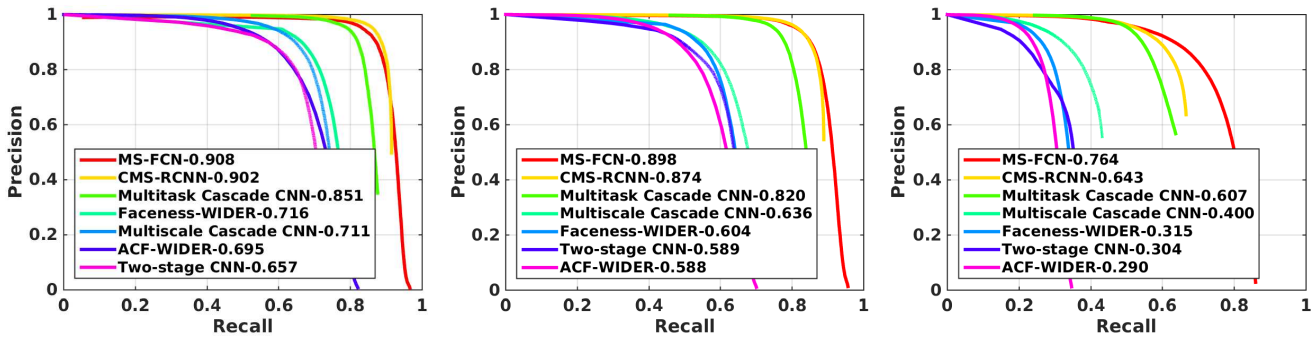
4. Experiments

In this section, we evaluate the proposed MS-FCN detector on four public face detection benchmarks, including FDDB [12], WIDER FACE [41], AFW [49], and PASCAL FACE [37] datasets and compare it against state-of-the-art methods. From our extensive empirical comparisons, we can see that MS-FCN can achieve the top detection performance, while running at about 23 FPS on a GPU for an image size of 640×480 pixels with no assumption on the minimum detectable face size.

4.1. Evaluation on FDDB [12]

The FDDB dataset is a challenging benchmark for face detection. It contains $2,845$ images with a total of $5,171$ faces, in a wide range of challenging scenarios including occlusions and out-of-focus blurred faces. However, most images are collected from news photos and the pose tends to be frontal. All faces in FDDB have been annotated with elliptical regions. We use the evaluation toolbox provided in [12] to compare the different face detectors. Following convention, we use the the discrete score metric [12] to evaluate detection performance.

To match the ellipse annotation on FDDB better, we uniformly transform our bounding box detections to ellipses, as suggested in [23]. The precision rate at 500 false positives is reported in Figure 3, which shows that CNN-based detectors outperform other baseline methods by a large margin. Compared to other CNN-based detectors, MS-FCN achieves a performance that is slightly lower (95.4% vs 96.0% at 351 false positives) than Xiaomi detector [34], which is based on Faster RCNN and also uses ResNet-50 as the backbone network. However, Xiaomi detector is fine-tuned on FDDB, while our MS-FCN detector is only trained on WIDER FACE and not fine-tuned on FDDB. MS-FCN registers a slightly better performance than Faster RCNN [14], which is also trained on WIDER FACE. STN [3] and MTCNN [45] are trained with the facial landmark information provided in the dataset. This information can filter out some false positives and boost detection performance, since most faces in FDDB are high resolution and frontal. Even without facial landmark information, our MS-FCN detector outperforms these two detectors. Note that MS-FCN can also benefit from the multi-task learning of MTCNN [45] to exploit the landmark information (if available).



(a) Easy set (b) Medium set (c) Hard set

Figure 4. On the WIDER FACE dataset, we compare our MS-FCN detector against several state-of-the-art methods: CMS-RCNN [48], Multi-task Cascade CNN [45], Faceness-WIDER [40], Multi-Scale Cascade CNN [41], Two-Stage CNN [41], and ACF-WIDER [39]. The average precision (AP) results are reported in the legend. The figure is best viewed in color.

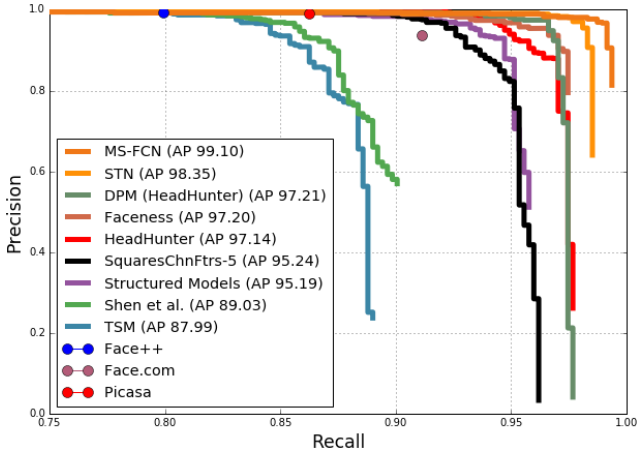


Figure 5. On the AFW dataset, we compare our MS-FCN detector against several state-of-the-art methods: STN [3], Faceness [40], HeadHunter [23], Structured Models [37], Shen *et al*. [29], DP-M [7] [23], TSM [49], Face.com, Face++, and Picasa. The AP: average precision. The average precision (AP) results are reported in the legend. The figure is best viewed in color.

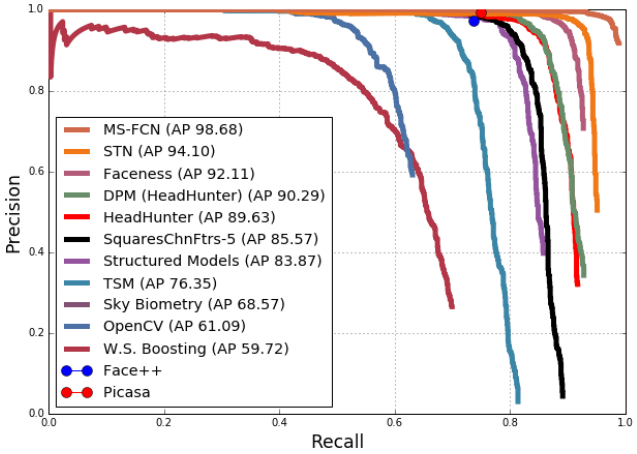


Figure 6. On the PASCAL FACE dataset, we compare our MS-FCN detector against several state-of-the-art methods: STN [3], Faceness [40], HeadHunter [23], Structured Models [37], DP-M [7] [23], TSM [49], W.S. Boosting [16], OpenCV, Sky Biometry, Face++ and Picasa. The average precision (AP) results are reported in the legend. The figure is best viewed in color.

4.2. Evaluation on WIDER FACE [41]

The WIDER FACE dataset is a recently released and large-scale face detection benchmark [41] for face detection in the wild. There are $393,703$ labeled faces with a high degree of variability in scale, pose, occlusion, expression, appearance, and illumination. Images are categorized into 61 social event classes, which have much more diversity than FDDB [12]. $40\%/10\%/50\%$ of the data is randomly divided as training, validation, and testing sets, respectively, for each social event. This dataset is divided into three subsets: Easy, Medium, and Hard, according to the heights of faces [41]. The Easy/Medium/Hard subsets include faces with height larger than $50, 30, 10$ pixels respectively. The Easy/Medium subsets are the subsets of the Medium/Hard ones, respectively. Compared to the Medium subset, there are many faces with height between $10 - 30$ pixels in the Hard subset, which explains that why it is difficult to achieve good performance on the Hard subset.

The testing results on WIDER Easy/Medium/Hard subsets are shown in Figure 4. And we see that the proposed MS-FCN detector achieves the best performance on all subsets, especially for the Hard subset, which is by far the most challenging. In comparison, CMS-RCNN [48] is a variant of Faster RCNN [14], which utilizes multi-layer *conv* feature fusion and context information to detect faces with low resolution. Compared to CMS-RCNN, MS-FCN is relatively simple, yet it achieves comparable (+0.6%), slightly better (+2.4%), and much better performance (+12.1%) on the Easy, Medium, and Hard subsets respectively. The improvement is quite significant on the latter subset, which

shows that MS-FCN is capable of accurate detection across a very wide range of face scales. Specifically, this demonstrates that up-sampling the *conv* feature maps is suitable for detecting faces, particularly at low resolutions. The multi-scale cascade CNN [41] also divides faces into different scales, on which each network is trained separately. Compared to this technique, MS-FCN adopts an approximate feature pyramid strategy to tackle multiple scales, where training can be done jointly.

4.3. Evaluation on AFW [49]

The AFW dataset was compiled by Zhu *et al*. [49], of which most images are from Flickr. It is a relatively small dataset and has only 205 images with 473 annotated faces. However, the images tend to contain cluttered background. Therefore, it is challenging for detectors to achieve good performance on this dataset.

In [23], an evaluation toolbox is provided, which contains updated annotations, since the original annotations are not comprehensive enough. We use the toolbox to evaluate MS-FCN against other detectors on AFW. The precision-recall curves are shown in Figure 5. Our MS-FCN detector achieves the best average precision (AP) value of 99.10%, which is slightly better (+0.8%) than the STN detector [3]. MS-FCN also outperforms other state-of-the-art methods by a large margin.

4.4. Evaluation on PASCAL FACE [37]

The PASCAL FACE dataset [37] is another widely used face detection benchmark, which consists of 851 images and $1,341$ annotated faces. This dataset contains large variations in both face viewpoint and appearance (*e.g*. large profile view, sunglasses, make-up, skin color, low resolution, and exaggerated expressions). This scenario is more diverse and challenging than AFW.

We evaluate MS-FCN on PASCAL FACE using the conventional toolbox in [23]. The precision-recall curves are given in Figure 6, which show our method outperforming all other detectors. In fact, MS-FCN achieves the best AP value of 98.68% with a large margin (+4.58%) separating it from the second best method (STN) [3].

4.5. Ablation Experiments

Table 1. On the WIDER FACE valuation dataset, we compare our MS-FCN (with K=4) detector against MS-CNN [2] and MS-FCN (with K=1). The average precision (AP) results are reported.

Method	Easy set	Medium set	Hard set
MS-FCN (K=4)	0.907	0.896	0.762
MS-CNN [2]	0.884	0.849	0.676
MS-FCN (K=1)	0.908	0.853	0.640

In Table 1, we add comparison experiments between MS-FCN (K=1, 4) and MS-CNN [2]. We can see that all detectors achieve nearly the same performance on Easy subset. However, on Medium and Hard subsets, MS-FCN (K=4) shows much better performance over the others. This demonstrates that using deep feature maps with high resolution is better for detecting small objects than using intermediate features. And the multi-scale model (MS-FCN, K=4) is better than the single-scale one (MS-FCN, K=1).

4.6. Efficiency Analysis

Our MS-FCN detector is very efficient in dealing with faces in a wide range of possible scales, because it contains K FCN models for detecting faces at different scales. More importantly, they share the lower intermediate *conv* feature maps, which are the computational bottlenecks of the network. When detecting, an image passes through our network in only one single shot thanks to the approximate feature pyramid. CascadeCNN [18, 45] uses the cascade framework to accelerate the detection speed and can run at 100 FPS on a GPU for VGA (640×480 pixels) images. However, this speed is reported based on the assumption that the minimum resolution of detected faces is higher than 80×80 pixels. With this assumption, many small faces would be missed, which is demonstrated in the WIDER FACE Hard subset. Decreasing the minimum detectable face size in CascadeCNN quickly increases the runtime of this method[1]. Currently, our detector runs at 23 FPS on VGA images for a very wide range of face sizes, *i.e*. with no assumption on the minimum detectable face size.

4.7. Qualitative Results

Some qualitative face detection results are shown in Figure 7. From Figure 7(A), we can see that the proposed MS-FCN detector can deal with challenging cases with extreme poses, exaggerated expressions, large scale variations, severe makeup and occlusion. However, Figure 7(B) also shows some failure cases, which are caused by very challenging nuisances in the wild scenarios.

5. Conclusion

In this paper, we propose a multi-scale fully convolutional network (MS-FCN) for face detection. In our detection system, the intermediate convolutional feature maps are shared by every scale model. Up-sampling and down-sampling are utilized to approximate K levels of an approximate feature pyramid, which encompasses a wide range of face scales. At each level of this pyramid, a FCN is learned to deal with faces within small scale variations. Our MS-FCN detector is tested on four public face detection bench-

[1]On a workstation with Intel CPU E5-2698 and NVIDIA TITAN X GPU, CascadeCNN runs at 46, 20, and 10 FPS at a minimum detectable face size of 80×80, 20×20 and 10×10 pixels, respectively.

Figure 7. Some example results of the proposed MS-FCN face detector. Green and red bounding boxes denote ground-truth annotations and MS-FCN detection results, respectively. (A) From the successful cases, we see that MS-FCN can deal with faces with extreme poses, large scale variations, exaggerated expressions, severe makeup and occlusion; (B) Some faces with extreme pose or severe occlusion can still cause failures for MS-FCN.

marks, including FDDB, WIDER FACE, AFW and PASCAL FACE datasets and it achieves superior performance when compared to state-of-the-art face detectors.

Acknowledgement

This work was supported by the King Abdullah University of Science and Technology (KAUST) Office of Sponsored Research under grant 2016-KKI-2880.

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] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos. A unified multi-scale deep convolutional neural network for fast object detection. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV*, pages 354–370, 2016.

[3] D. Chen, G. Hua, F. Wen, and J. Sun. *Supervised Transformer Network for Efficient Face Detection*, pages 122–138. Springer International Publishing, Cham, 2016.

[4] D. Chen, S. Ren, Y. Wei, X. Cao, and J. Sun. Joint cascade face detection and alignment. In *Computer Vision–ECCV 2014*, pages 109–122. Springer, 2014.

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

[6] S. S. Farfade, M. J. Saberian, and L.-J. Li. Multi-view face detection using deep convolutional neural networks. In *Pro-*

ceedings of the 5th ACM on International Conference on Multimedia Retrieval, pages 643–650. ACM, 2015.

[7] 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.

[8] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1440–1448, 2015.

[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[10] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *arXiv prepring arXiv:1506.01497*, 2015.

[11] 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.

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

[13] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM International Conference on Multimedia*, pages 675–678. ACM, 2014.

[14] H. Jiang and E. Learned-Miller. Face detection with the faster r-cnn. *arXiv preprint arXiv:1606.03473*, 2016.

[15] K. H. J. S. Jifeng Dai, Yi Li. R-FCN: Object detection via region-based fully convolutional networks. *arXiv preprint arXiv:1605.06409*, 2016.

[16] Z. Kalal, J. Matas, and K. Mikolajczyk. Weighted sampling for large-scale boosting. In *BMVC*, pages 1–10, 2008.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[18] 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.

[19] 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.

[20] Y. Li, B. Sun, T. Wu, and Y. Wang. *Face Detection with End-to-End Integration of a ConvNet and a 3D Model*, pages 420–436. Springer International Publishing, Cham, 2016.

[21] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. *SSD: Single Shot MultiBox Detector*, pages 21–37. Springer International Publishing, Cham, 2016.

[22] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[23] 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.

[24] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7):971–987, 2002.

[25] H. Qin, J. Yan, X. Li, and X. Hu. Joint training of cascaded cnn for face detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3456–3465, 2016.

[26] R. Ranjan, V. M. Patel, and R. Chellappa. Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *arXiv preprint arXiv:1603.01249*, 2016.

[27] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015.

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

[29] X. Shen, Z. Lin, J. Brandt, and Y. Wu. Detecting and aligning faces by image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3460–3467, 2013.

[30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[31] G. Trigeorgis, P. Snape, M. A. Nicolaou, E. Antonakos, and S. Zafeiriou. Mnemonic descent method: A recurrent process applied for end-to-end face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4177–4187, 2016.

[32] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013.

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

[34] S. Wan, Z. Chen, T. Zhang, B. Zhang, and K.-k. Wong. Bootstrapping face detection with hard negative examples. *arXiv preprint arXiv:1608.02236*, 2016.

[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.

[36] J. Yan, X. Zhang, Z. Lei, and S. Z. Li. Real-time high performance deformable model for face detection in the wild. In *Biometrics (ICB), 2013 International Conference on*, pages 1–6. IEEE, 2013.

[37] J. Yan, X. Zhang, Z. Lei, and S. Z. Li. Face detection by structural models. *Image and Vision Computing*, 32(10):790–799, 2014.

[38] Y. L. L. C. W. G. Yancheng Bai, Wenjing Ma and L. Yang. Multi-scale fully convolutional network for fast face detection. In *Proceedings of the British Machine Vision Conference (BMVC)*. BMVA Press, September 2016.

[39] 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.

[40] S. Yang, P. Luo, C.-C. Loy, and X. Tang. From facial parts responses to face detection: A deep learning approach. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3676–3684, 2015.

[41] S. Yang, P. Luo, C. C. Loy, and X. Tang. Wider face: A face detection benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[42] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. Huang. Unitbox: An advanced object detection network. In *Proceedings of the 2016 ACM on Multimedia Conference*, MM '16, pages 516–520, New York, NY, USA, 2016. ACM.

[43] S. Zafeiriou, C. Zhang, and Z. Zhang. A survey on face detection in the wild: Past, present and future. *Computer Vision and Image Understanding*, 138:1 – 24, 2015.

[44] S. Zagoruyko, A. Lerer, T.-Y. Lin, P. O. Pinheiro, S. Gross, S. Chintala, and P. Dollár. A multipath network for object detection. *arXiv preprint arXiv:1604.02135*, 2016.

[45] 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.

[46] L. Zhang, L. Lin, X. Liang, and K. He. *Is Faster R-CNN Doing Well for Pedestrian Detection?*, pages 443–457. Springer International Publishing, Cham, 2016.

[47] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014.

[48] C. Zhu, Y. Zheng, K. Luu, and M. Savvides. Cms-rcnn: Contextual multi-scale region-based cnn for unconstrained face detection. *arXiv preprint arXiv:1606.05413*, 2016.

[49] 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.

[50] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *European Conference on Computer Vision*, pages 391–405. Springer, 2014.