DETECT FACE IN THE WILD USING CNN CASCADE WITH FEATURE AGGREGATION AT MULTI-RESOLUTION

Jingjing Deng *Xianghua Xie*

Department of Computer Science, Swansea University, Swansea, UK

http://csvision.swansea.ac.uk

ABSTRACT

Face detection in the wild is a challenging vision problem due to large variations and unpredictable ambiguities commonly existed in real world images. Whilst using hand-crafted features is generally problematic, introducing powerful but complex models is often computationally inefficient. Feature aggregation and multi-resolution are two efficient strategies for traditional visual recognition methods. In this paper, we show that such strategies can be integrated into Convolutional Neural Network (CNN) architecture via average pooling and channel-wise feature concatenation. Shallow networks with feature aggregation at multi-resolution enables the traditional cascade framework to tackle the challenging detection problems efficiently. The proposed method is tested on a public benchmark with across dataset evaluation. Both quantitative and qualitative results show promising performance improvements on detecting faces in unconstrained environment.

***Index Terms*—** Face detection, CNN, cascade, feature aggregation, and multi-resolution.

1. INTRODUCTION

"Face in the wild" is a challenging detection problem, where class distribution between face and background is extremely unbalanced and heavily biased towards the background. Faces are captured with large pose and facial expression variations, severe occlusions and clutter, and varied lighting conditions. The traditional Viola-Jones (VJ) [1] framework which uses a multi-stage cascade detector, performs poorly due to the limitations of discriminative power of its feature and classifier. Current advances in Deep Neural Network (DNN) based methods have been shown superior over many other methods, such as Deep Dense Face Detector (DDFD) [2], CNN Deformable Part Model (DPM) [3], Regions with Convolutional Neural Networks Features (R-CNNs) [4], and Graph CNN [5]. Furthermore, Fully Convolutional Neural Network (FCN) [6] was firstly introduced for semantic segmentation, and then adapted to solve object detection problems [7, 8, 9]. Although deeper models generally outperforms shallow ones, training complex models is not a trivial task, especially for binary detection problems where the distribution of target object and background is extremely unbalanced. In order to efficiently train a deep detection net, a typical strategy is to adapt a pre-trained image recognition model via fine tuning. However, normally large patch size of image input is used for recognition nets. Without sufficient visual content, it is significantly difficult for deep models to capture small objects. Multi-scale cascade detection has proved to be an efficient scheme to construct face detector with different resolutions in an ensemble fashion [10, 11, 12]. The most relevant work to ours is [13], where 3 face-nonface classification CNNs are used for separating face regions from background and 3 calibration CNNs are used to refine the location of detected bounding box. However, cascade based method make a compromise between the number of stages, accuracy and efficiency. In addition, refining the detected windows between stages introduces re-sampling the patches from the original image, which is non-trivial during the testing phase. In our method, such refinement procedure only applies at the last stage, hence no patch re-fetching is required.

Feature aggregation and multi-resolution strategies were proved to be the efficient schemes for visual recognition tasks with hand-crafted features [14, 15]. In this paper, we show that introducing such strategies into CNN architecture design also helps improving the accuracy of challenging face detection problems. The proposed Multi-Resolution Feature Aggregation (MRFA) embeds a fast elimination stage, and two verification stages into a cascade framework. A large amount of easy background patches generated by sliding window are eliminated at the very early stage using a shallow but fast net at a coarse scale. To precisely locate the face region, verification nets are designed with feature aggregation at multi-resolution via average pooling and channel-wise feature concatenation. The face-nonface binary decision is first made by the detection classifier, and then for all positive predictions, a regression procedure is applied to refine the locations, aspect ratios of major and minor axes, and angles of output bounding boxes. Our work leverages recent advances in CNNs for efficient face detection, where deep structure and large scale model adapting that require excessive resources, such as training data and time on both pre-trained and adapted models, are avoided. Our proposed solution is not overwhelmed by the

model complexity.

2. METHOD

Fig. 1 shows the basic flowchart of the proposed MRFA face detector. It consists of two main phases: fast elimination, and precise verification. Window patches are firstly generated by densely scanning the input image at multiple scales using sliding windows. The majority of window patches are quickly eliminated as background by an *ElmNet* using a patch resolution of 12×12. Then, all retained candidates from *ElmNet* are verified by two *VefNets* using a patch resolution of 48×48. At the end of cascade, the detection branch outputs the binary classification of face-nonface decision with confidence scores, meanwhile the regression branch refines the bounding box location by determining the optimal face center, angle, and aspect ratio. The final detections are obtained via removing redundant detections with a 2-step Non-Maximal Suppression (NMS).

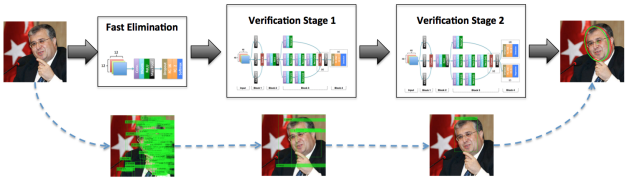

Fig. 1. The pipeline of the proposed MRFA detector.

2.1. Sliding Window Elimination Net

A large amount of patch candidates are generated by the sliding window method. The *ElmNet* is designed to quickly eliminate negative patches to reduce the computational cost for the following phases. Fig. 2 provides the details of the architecture for *ElmNet*, where only one convolutional layer and one fully connected layer are used. Adopting such simple CNN structure is motivated by the following two reasons. Firstly, *ElmNet* has a small input size of 12×12, a small kernel size of 3×3, and a small number of filters of 16. Compared to other nets, *ElmNet* has significantly smaller number of parameters, which enables a lower memory consumption and a much lower computational cost. Secondly, at this fast elimination stage, low frequency image features extracted from coarse spatial resolution is more effective in rejecting easy negative hypotheses. Since there is no hierarchical feature extraction within *ElmNet*, the discriminative power is limited. In order to retain most positive windows for the following stage, a high recall rate can be achieved by shifting the decision boundary of Softmax layer towards zero. For example, using a minimal face size of 48×48, 91.12% recall can be achieved on FDDB dataset by shifting the decision boundary to 0.01.

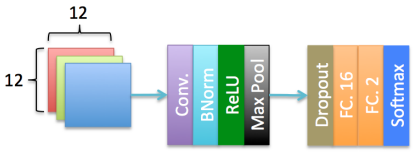

Fig. 2. Network architecture of *ElmNet*.

2.2. Multi-Task Verification Net

A multi-task *VefNet* is designed to precisely locate face regions by verifying retained face candidates at a higher image resolution of 48×48. Fig. 3 provides the details of the architecture, where *VefNet* is divided into 4 main blocks.

Block 1 consists of 3 average pooling branches which use three filters (1×1, 3×3, and 5×5) with no spatial downsampling. Three pooling branches joint together via concatenating the outputs across channels. In contrast to traditional multi-scale methods that construct Gaussian pyramid as network input, such structure embeds a simple average blurring scheme into network itself, which greatly helps the later computational blocks to identify scale-invariant features.

Block 2 extracts the first level of visual features via sequentially passing the multi-resolution images through a convolutional layer, a batch normalization layer, an ReLU non-linear transform layer, and a max pooling layer. In order to gain high speed efficiency, we aggressively reduce the spatial resolution by setting the strides of convolutional layer and max pooling layer both to 2. Batch normalization layer is inserted between convolutional layers and ReLU layers (same for other blocks) to enforce regularization to internal co-variate shift caused by weight updates during back-propagation. Inspired by GoogLeNet [16], a simplified inception module which contains three feature extraction branches, is used to generalize discriminative power further.

Each branch in Block 3 starts with a dimensionality reduction module with a 1×1 convolutional layers which removes redundant feature channels, and improves computational efficiency. Blocks 3.b and 3.c consist of two 3×3, and one 5×5 feature extraction modules, respectively. It is worth noting that although a 5×5 filters has the same reception field as two consecutive 3×3 filters, the latter could generalize even deeper structures. The outputs of three branches in Block 3 are concatenated across channels, followed by an average pooling layer to reduce the spatial resolution. Yang *et al.* [14] showed that aggregating hand-crafted features improves the detection accuracy. In our method, Block 3 embeds such multi-level feature interfusion into a learnable framework.

Block 4 contains two fully connected objective branches, detection branch and bounding box regression branch (top row and bottom row of Block 4 in Fig. 3 respectively). Previous blocks are trained with detection branch using Softmax loss, whereas regression branch is trained using smooth $\ell 1$

loss. Binary classification is carried out by the detection branch without shifting the decision boundary at the last stage.

As the face candidates are generated by sliding window, the optimal locations of faces may not be in the hypothesis set. The detection performance can be further boosted by refining the locations of output bounding boxes. The regression target is a quintuple defined by two coordinates of face center offset to top left corner, lengths of major and minor axes with respect to the size of bounding box, and the angle of major axis with vertical axis. A positive value of face angle indicates an anti-clock rotation with respect to vertical axis. The bounding box calibration procedure only applies to the positive response given by detection branch. Then a 2-step NMS is followed to remove redundancies. For the detections at the same scale, we iteratively select the detection with highest confidence score and remove the detections that has the Intersection over Union (IoU) ratio larger than 0.50 with selected window. For the detections at different scales, the redundancies can be found by measuring the Intersection over Minimum (IoM) ratio, where the threshold is set to 0.75. The first step removes the redundant detections that are spatially offset to the correct location, and the second step enables removing redundancies in scale.

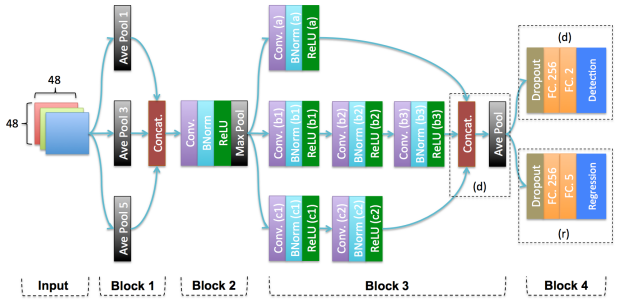

Fig. 3. Network architecture of *VefNet*.

3. EXPERIMENT AND DISCUSSION

The Annotated Facial Landmarks in the Wild (AFLW) [17] dataset was used to train the face detector. The dataset contains 22,712 labeled faces out of 21,123 images. The positive face windows were further augmented by horizontal flipping. In total, 45,424 faces were used in the training procedure. The negative images contain no face. To bootstrap non-face images, labeled face windows were replaced with non-face patches which were randomly sampled from The PASCAL Visual Object Classes (PASCAL VOC) dataset [18] (the person subset was excluded). In total, 18,089 negative images were generated using this bootstrapping approach. To train *ElmNet*, non-face samples were cropped randomly from negative images, and then resized to 12×12. The negative-positive

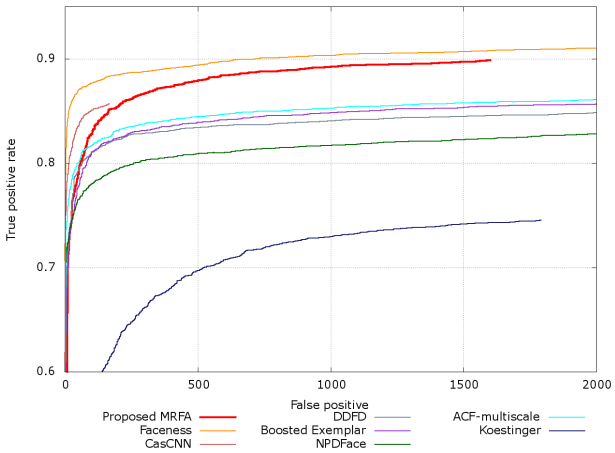

Fig. 4. ROC curves of the proposed detector and recent methods on FDDB with the discrete score metric.)

ratio of *ElmNet* was set to $10:1$. With cascading set-up, the negative samples for training the detection branch of *VefNet* were the residuals (false positives) generated by densely scanning the negative image set using previous stages. The networks were trained using MatConvNet [19]. The number of epochs was set to 50, the size of mini-batch was 128, and momentum of 0.9 were used. The learning rate gradually drops from $1e^{-2}$ to $1e^{-5}$. Regression branch of *VefNet* was trained independently, and it converges in 2 epochs.

The proposed face detector was quantitatively evaluated on the FDDB [20] dataset that contains 5,171 annotated faces in 2,845 images. The quantitative results were generated following the standard evaluation procedure with the software provided by the authors. For discrete score evaluation, detections with over 0.50 IoU with annotations are counted as true positive. Since the ground truth faces are labeled using ellipses, for fair comparison, we also fitted ellipses to our bounding boxes given the outputs of regression branch of *VefNet*. NMS was used to retain the ones with highest confidence scores while removing the redundancies. We compared proposed method with state-of-the-art representative methods which are evaluated on the same dataset. Table 1 shows the comparison of discrete and continuous detection rates given the number of false positives, and the discrete ROC curves are shown in Fig. 4. DPM based methods, Faceness [3], is leading the performance, mainly because the variations of facial parts are relatively small, thus detecting facial parts are more robust than detecting face as a whole. However, DPM methods require training part detectors, and searching optimal configuration, which makes building the detector a laborious, time-consuming task, and are known to be much slower than cascade based methods. CasCNN [13] refines the bounding boxes between each stage, and then re-fetches the image patches for the next stage. Such procedure does improve both discrete and continuous scores, however it is

Fig. 5. Typical detection results on FDDB dataset (red: ground truth, blue: detection results of the proposed method).

non-trivial. Our method only applies simple location calibration at the last stage, and no re-fetching is required. Compared to DDFD where a deeper structure is used, the proposed MRFA achieved higher True Positive (TP) rate after 100 false positives, and outperformed it by a significant margin (6.9% higher) at 500 false positives. ACF-Multiscale [14] method aggregates multiple features, such as color, gradient, local histogram, into a rich representation, and then trains multiple soft cascade with depth-2 decision tree for different views. It shows that combining multiple models and features outperforms a single model. The computational cost of aggregating feature channels is considerably more expensive. Significantly, Koestinger [21] showed that without rich features, the performance of multi-view based method drops by a significant margin. In addition, sophisticated post-processing is required to combine the multiple detection outputs given by detectors of different views. The proposed method embeds feature aggregation and multi-resolution strategies into the network architecture. The features are self-learned through training, and it outperforms the traditional methods which use the cascade framework and hand-crafted features, such as NPDFace [22], ACF-Multiscale [14] and Koestinger [21]. For image retrieval based methods, such as Boosted Exemplar [23], they generally have higher recall rate compared to those traditional methods, however, our method outperformed [23] in all aspects.

Qualitative results on the FDDB dataset are shown in Figs. 5, and 6. Red and blue ellipses represent ground truth and true positives, whereas yellow and green ellipses represent false positives and false negatives respectively. Fig. 5 illustrates some examples of typical detection results with large pose and facial expression variations, blurring, and severe occlusion and clutter. The first row of Fig. 6 shows some examples of false positives and false negatives. The false positives are usually observed at the region that contains partial face, and false negatives are mainly caused by severe blurring and faces in small scale. The second row of Fig. 6 shows some interesting detections in yellow, which are counted as false positives since there are no annotations to match. However, they are in fact correct detections. Both quantitative and qualitative results show promising performances on detecting face in unconstrained environment.

Table 1. Comparison of detection rates (%) with both discrete and continuous metrics on FDDB.

	Disc. Metric		**Cont. Metric**	
	FP=100	FP=500	FP=100	FP=500
Proposed	82.77	87.89	65.13	68.85
Faceness [3]	87.64	89.38	69.70	71.38
CasCNN [13]	85.07	*N.A.*	66.29	*N.A.*
DDFD [2]	80.99	83.40	64.45	66.41
Li *et al.* [23]	80.82	83.89	56.87	59.20
NPDFace [22]	77.97	80.89	58.04	60.25
Yang *et al.* [14]	81.65	84.45	60.43	62.49
Koestinger [21]	57.03	69.70	40.55	49.49

Fig. 6. First row: examples of false positives and false negatives. Second row: examples of correct detections but counted as false positives. (red: ground truth, blue: true positive detection, yellow: false positive detection, green: false negatives).

4. REFERENCES

[1] Paul Viola and Michael J Jones, "Robust real-time face detection," *IJCV*, vol. 57, no. 2, pp. 137–154, 2004.

[2] Sachin Sudhakar Farfade, Mohammad J Saberian, and Li-Jia Li, "Multi-view face detection using deep convolutional neural networks," in *ACM on International Conference on Multimedia Retrieval*. ACM, 2015, pp. 643–650.

[3] Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang, "From facial parts responses to face detection: A deep learning approach," in *ICCV*, 2015, pp. 3676–3684.

[4] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *NIPS*, 2015, pp. 91–99.

[5] Michael Edwards and Xianghua Xie, "Graph convolutional neural network," in *British Machine Vision Conference*, 2016.

[6] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015, pp. 3431–3440.

[7] Jiahui Yu, Yuning Jiang, Zhangyang Wang, Zhimin Cao, and Thomas Huang, "Unitbox: An advanced object detection network," in *ACM Multimedia*. ACM, 2016, pp. 516–520.

[8] Zhenheng Yang and Ram Nevatia, "A multi-scale cascade fully convolutional network face detector," *arXiv preprint arXiv:1609.03536*, 2016.

[9] Yancheng Bai, Wenjing Ma, Yucheng Li, Liangliang Cao, Wen Guo, and Luwei Yang, "Multi-scale fully convolutional network for fast face detection," in *BMVC*, September 2016.

[10] Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang, "Wider face: A face detection benchmark," *arXiv preprint arXiv:1511.06523*, 2015.

[11] Jingjing Deng, Xianghua Xie, and Michael Edwards, "Combining stacked denoising autoencoders and random forests for face detection," in *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer International Publishing, 2016, pp. 349–360.

[12] Jingjing Deng and Xianghua Xie, "Nested shallow cnn-cascade for face detection in the wild," in *IEEE Conference on Automatic Face and Gesture Recognition*, 2017.

[13] Haoxiang Li, Zhe Lin, Xiaohui Shen, Jonathan Brandt, and Gang Hua, "A convolutional neural network cascade for face detection," in *CVPR*, 2015, pp. 5325–5334.

[14] Bin Yang, Junjie Yan, Zhen Lei, and Stan Z Li, "Aggregate channel features for multi-view face detection," in *IJCB*. IEEE, 2014, pp. 1–8.

[15] Shengcai Liao, Xiangxin Zhu, Zhen Lei, Lun Zhang, and Stan Z Li, "Learning multi-scale block local binary patterns for face recognition," in *Advances in Biometrics*, pp. 828–837. Springer, 2007.

[16] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions," in *CVPR*, 2015, pp. 1–9.

[17] Martin Koestinger, Paul Wohlhart, Peter M. Roth, and Horst Bischof, "Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization," in *IEEE International Workshop on Benchmarking Facial Image Analysis Technologies*, 2011.

[18] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *IJCV*, vol. 111, no. 1, pp. 98–136, Jan 2015.

[19] A. Vedaldi and K. Lenc, "MatConvNet – convolutional neural networks for MATLAB," in *ACM Multimedia*, 2015.

[20] Vidit Jain and Erik Learned-Miller, "FDDB: A benchmark for face detection in unconstrained settings," Tech. Rep. UM-CS-2010-009, University of Massachusetts, Amherst, 2010.

[21] Martin Koestinger, *Efficient Metric Learning for Real-World Face Recognition*, Ph.D. thesis, Graz University of Technology, Faculty of Computer Science, 2013.

[22] S. Liao, A. K. Jain, and S. Z. Li, "A fast and accurate unconstrained face detector," *TPAMI*, vol. 38, no. 2, pp. 211–223, Feb 2016.

[23] Haoxiang Li, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Gang Hua, "Efficient boosted exemplar-based face detection," in *CVPR*, 2014, pp. 1843–1850.