Cascaded CNN for Face Detection

Hongzhi Liu

May 23, 2018

1 Overview of Research Method

As is known to all, face detection plays an important role in face based image analysis and is one of the fundamental problems in computer vision. The performances of various face based applications, from face identification and verification to face clustering, tagging and retrieval, rely on accurate and efficient face detection.

The previous face detection research can be seen as a history of more efficiently sampling the output space to a solvable scale and more effectively evaluating per configuration. One natural idea to achieve this is using cascade, where classifier with low computation cost can be firstly used to shrink background while keeping the faces. In the thesis of Professor Qin, an expert from SenseTime Group Limited, he shows that in CNN based cascade detection, other than enjoying the advantages in efficiency as traditional cascade, different stages in the cascade can be jointly trained to achieve better performance [1].

Furthermore, the team show that the back propagation algorithm used in training CNN can be naturally used in training CNN cascade. Joint training can be conducted on naive CNN cascade and more sophisticated cascade such as region proposal network (RPN) and fast RCNN. Besides, they show that the jointly trained cascade CNN as well as the jointly trained RPN and fast R-CNN can achieve leading performance on face detection.

2 Joint Training of Cascaded CNN

Algorithms using cascaded stages are widely used in detection tasks. In the early stages, week classifiers can reject most false negatives. In the later stages, stronger classifiers can save computation with less proposals. With the development of deep CNNs, multi-stage CNNs are getting popular. Cascaded CNNs [2] and faster R-CNN [3] are such mechanisms.

The cascaded CNN for face detection contains three stages. In each stage, they use one detection network and one calibration network. There are totally six CNNs. In practice, this makes the training process quite complicated. While in faster R-CNN, it will use one CNN called Region Proposal Network (RPN) for generating proposals, the other CNN called fast R-CNN for detection.

In order to share convolutional layers between region proposal network and detection network, the typical training of RPN and fast R-CNN adopts four separate stages and uses alternating optimization. The core idea is to let region proposal and later detection network share convolutional layers. RPN and the first stage FCN of cascaded CNNs are highly similar. They both use fully convolutional neural network to generate proposals. Naturally FCN can better handle more scales than RPN, while RPN can save computation with only one input scale. Therefore, the joint training of cascaded CNNs can apply to RPN and fast R-CNN.

Professor Qin designs a joint training architecture to train the network once for all and call this architecture FaceCraft. Fig. 1 demonstrates this joint training architecture. During training, the network takes an image of size 48×48 as input, and outputs one joint loss of three branches. The three branches are called $x12$, $x24$, $x48$ respectively, corresponding to the input size of each network. They use *ReLU* for non-linear layers and drop-out before classification or regression layer.

Each branch has a face v.s. non-face classification loss and a bounding-box regression loss. Adding them with loss weights, the team get the joint loss function as Equation (1):

$$L_{joint} = \lambda_1 L_{x12} + \lambda_2 L_{x24} + \lambda_3 L_{x48} \tag{1}$$

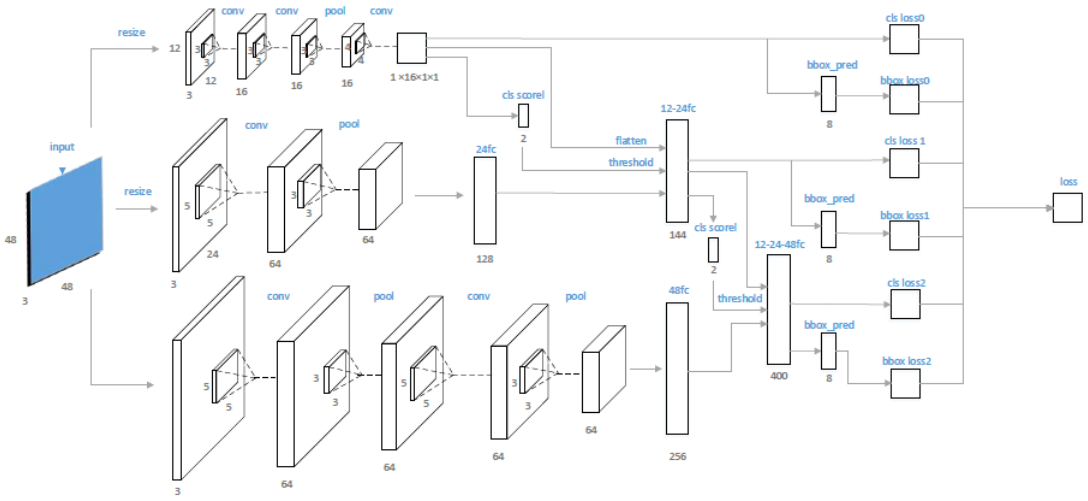


Figure 1: Joint training architecture. During training, the network takes an image of size 48×48 as input, and outputs one joint loss of three branches. The network is optimized through back-propagation. Compared to separate networks, the joint network also use threshold control layers, to decide which proposals from up branches contribute to the loss of the down branches.

where L_{x12}, L_{x24} and L_{x48} denote different losses of three branches. The loss of each branch is calculated by Equation (2). λ_1, λ_2 and λ_3 are loss weights of the three branches.

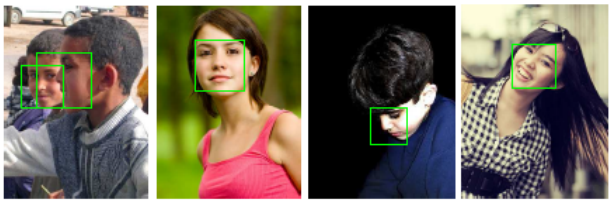
They use a multi-task loss of classification and bounding-box regression to jointly optimize this branch. Besides, they use softmax loss for classification and smooth L_1 loss for bounding-box regression as Equation (2):

$$L(p, u, t^u, v) = L_{cls}(p, u) + \lambda[u \geq 1]L_{loc}(t^u, v) \quad (2)$$

where $L_{cls}(p, u) = -\log p_u$ is log loss for true class u.

3 Evaluation of the Pipeline

They carry out experiments on face detection dataset to evaluate the joint training pipeline. In the test results, non-frontal face bounding-box centred on the nose, which is consistent with their training ground-truth shown in Fig. 2. While in AFW [4] ground-truth, nose is on the bounding-box edge.



(a) lateral (b) frontal (c) downward (d) upward

Figure 2: Face annotation examples.

As shown in Table 1, with our presented RPN + FR-CNN (fast R-CNN) joint training pipeline, the AP (average precision) on AFW is 98.7%, compared to the baseline result 97.0% trained with 4-stage training method proposed in [3]. On FDDB [5], the recall (1000 false positives) is 91.2% v.s. 89.7%. For the F-RCNN branch, the final joint training loss decreases 64% compared to separate training. In joint RPN + F-RCNN, the detection results mostly have much higher confidence scores than separate training results, which have lower confidence scores

because of FRCNN domination in convolution layers.

Table 1: Comparison of training methods of RPN +F-RCNN

Benchmark	Separate	Joint
AFW	97.0%	98.7%
FDDB	89.7%	91.2%

References

[1] H. Qin, J. Yan, X. Li, and X. Hu, "Joint training of cascaded cnn for face detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3456–3465. 1

[2] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Computer Vision and Pattern Recognition*, 2015, pp. 5325–5334. 1

[3] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017. 1, 2

[4] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Computer Vision and Pattern Recognition*, 2012, pp. 2879–2886. 2

[5] V. Jain and E. Learned-Miller, *FDDB: A Benchmark for Face Detection in Unconstrained Settings*, 2010. 2