Face Detection with End-to-End Integration of a ConvNet and a 3D Model

Yunzhu Li[1,*], Benyuan Sun[1,*], Tianfu Wu[2] and Yizhou Wang[1]

[1] School of EECS, Peking University

[2] Department of ECE and the Visual Narrative Cluster, North Carolina State University

{leo.liyunzhu, sunbenyuan, Yizhou.Wang}@pku.edu.cn, tianfu_wu@ncsu.edu

Overview

This paper presents a method for face detection in the wild, which integrates a ConvNet and a 3D mean face model in an end-to-end multi-task discriminative learning framework. There are two components:

i) **The face proposal component** computes face proposals via estimating facial key-points and the 3D transformation parameters for each predicted key-point w.r.t. the 3D mean face model.

ii) **The face verification component** computes detection results by refining proposals based on configuration pooling.

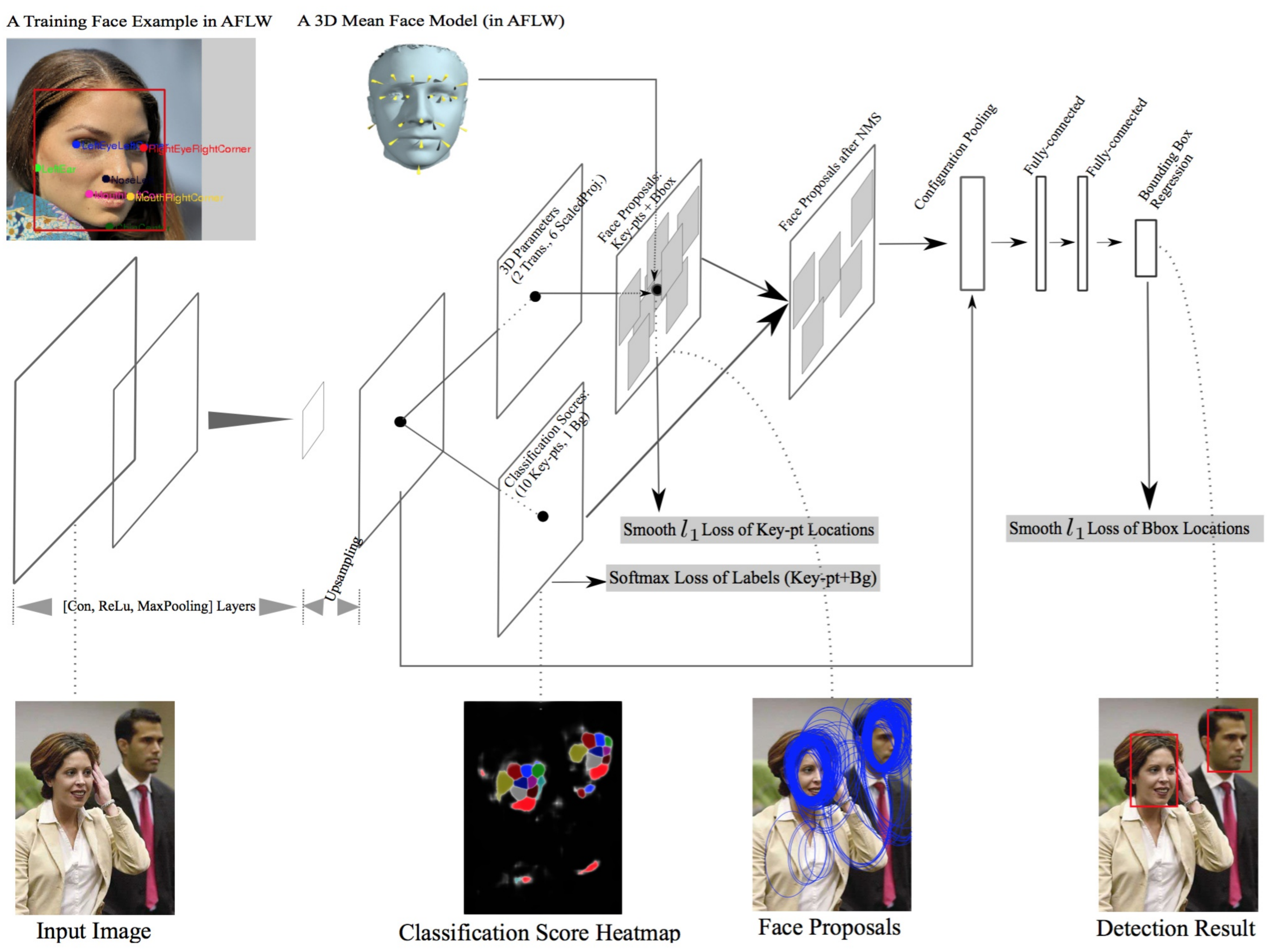

Figure 1: Illustration of the proposed method (Top), and a sample intermediate and the final detection results (Bottom).

The Proposed Method

Face Representation

A 3D mean face model is represented by a $n \times 3$ matrix, $F^{(3)}$. The 3D transformation parameters Θ are defined by,

$$\Theta = (\mu, s, A^{(3)}), \tag{1}$$

where μ represents a 2D translation (dx, dy), s a scaling factor, and $A^{(3)}$ a 3×3 rotation matrix. We can compute the projected 2D key-points by,

$$\hat{F}^{(2)} = \mu + s \cdot \pi(A^{(3)} \cdot F^{(3)}), \tag{2}$$

where $\pi()$ projects a 3D key-point to a 2D one.

ConvNet Architecture

Referring from Figure 1, the ConvNet is consisted by:

- Convolution, ReLu and MaxPooling Layers.
- An Upsampling Layer implemented by deconvolution.
- A Facial Key-point Label Prediction Layer. Samples are shown in Figure 2.
- A 3D Transformation Parameter Estimation Layer.
- A Face Proposal Layer. Samples are shown in Figure 3.
- A Key-point based Configuration Pooling Layer.
- A Face Bounding Box Regression Layer.

Figure 2: Sample detection results in the FDDB and the corresponding heat map of facial key-points.

Figure 3: Examples of face proposals computed using predicted 3D transformation parameters.

End-to-End Training

During training, the loss are three-folds:

- The Classification Softmax Loss of Key-point Labels,

$$\mathcal{L}_{cls} = -\sum \log(p_\ell^{\mathbf{x}}), \tag{3}$$

where ℓ is the label for position $\mathbf{x}$, and $p^{\mathbf{x}}$ is the predicted discrete probability distribution from our model.

- The Smooth l_1 Loss of Key-point Locations,

$$\mathcal{L}_{loc}^{pt} = \sum \text{Smooth}_{l_1}(\hat{F}^{(2)}, F^*), \tag{4}$$

where $\hat{F}^{(2)}$ is the projected 2D key-points calculated according to Eqn 2 from predicted 3D transformation parameters, and F^* is the ground truth locations.

- The Smooth l_1 Loss of Bounding Boxes, $\mathcal{L}_{loc}^{box}$.

The overall loss function is defined by,

$$\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{loc}^{pt} + \mathcal{L}_{loc}^{box} \tag{5}$$

Experiments

Our method is evaluated on FDDB and AFW. Results are shown in Figure 4 and Figure 5.

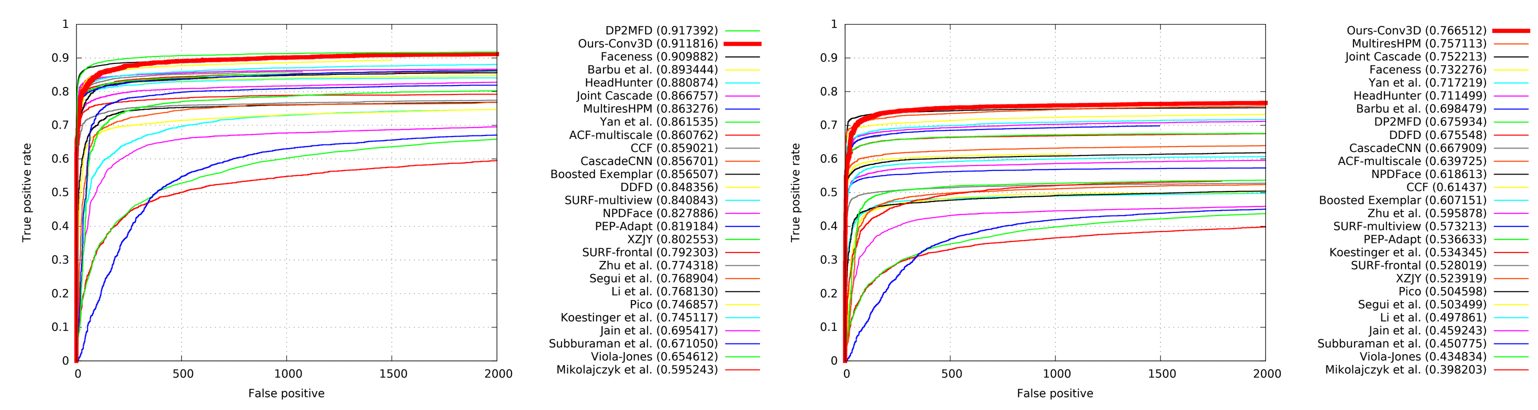

Figure 4: FDDB results based on discrete (left) and continuous scores (right).

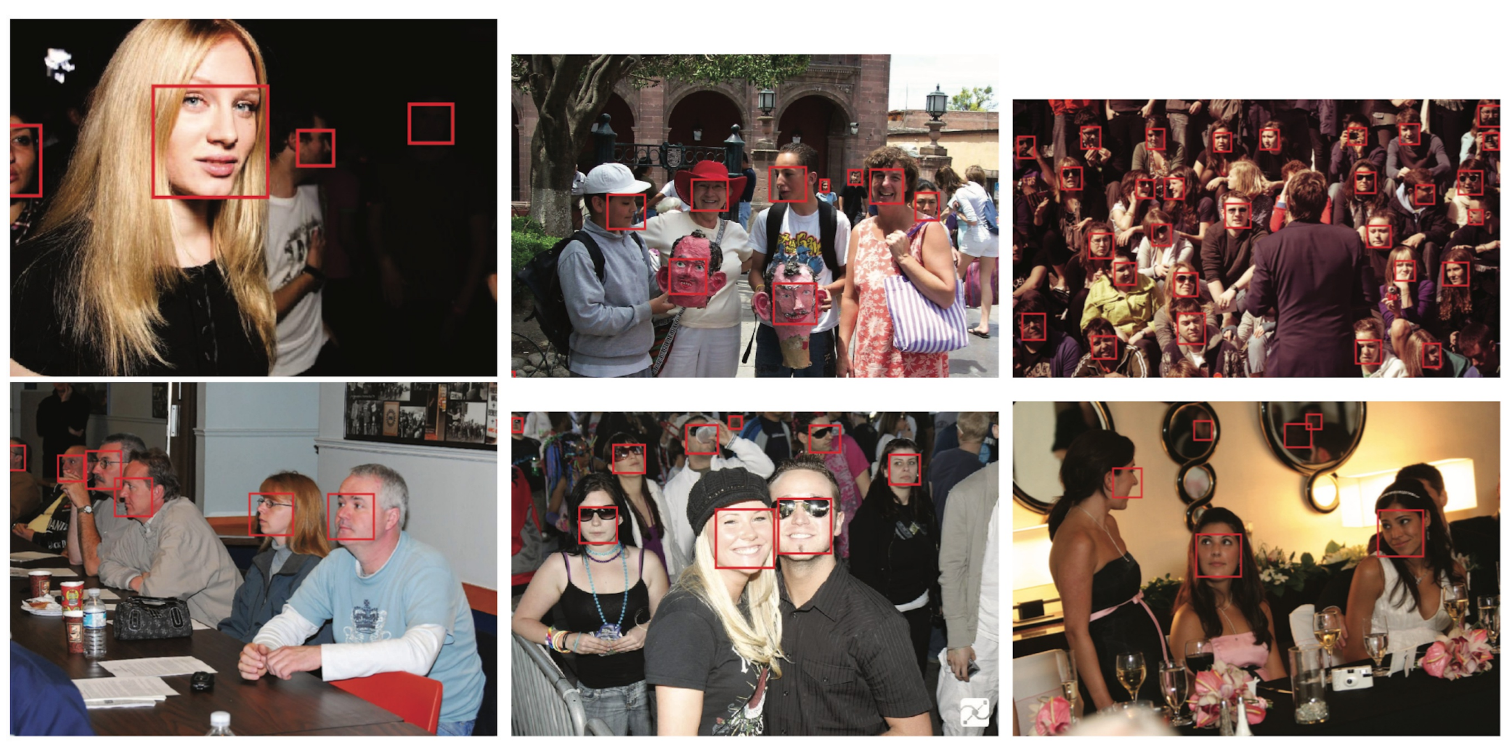

Figure 5: Sample qualitative results on the AFW dataset.

Conclusion and Discussion

Our method is a clean and straightforward solution when taking into account a 3D model in face detection, with very compatible state-of-the-art performance obtained.

We are also working on extending the proposed method for other types of rigid/semi-rigid object classes(e.g., cars). We expect that we will have a unified model for cars and faces which can achieve state-of-the-art performance.

arXiv Github