INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY

HYDERABAD

COMPUTER VISION

Face Detection, Pose Estimation, and Landmark Localization in the Wild

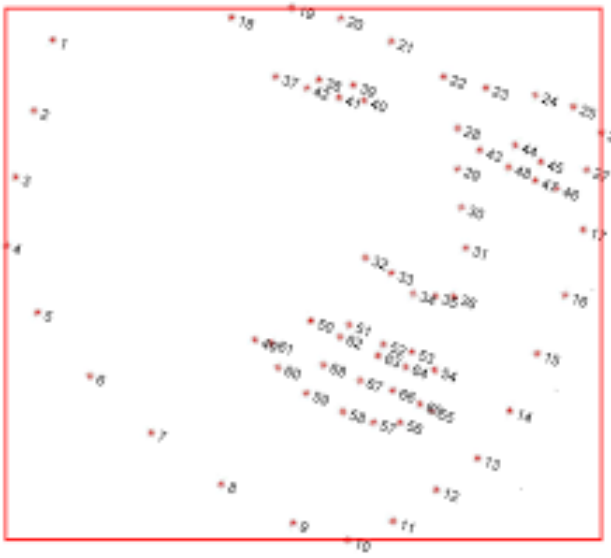
Author:	*Roll Number:*
Sivangi Singh	2018201001
Danish Mukhtar	2018201016
Shubham Pokhriyal	201820108

March 3, 2019

Contents

1 ABSTRACT

We present a unified model for face detection, pose es- timation, and landmark estimation in real-world, cluttered images. Our model is based on a mixtures of trees with a shared pool of parts; we model every facial landmark as a part and use global mixtures to capture topological changes due to viewpoint. We show that tree-structured models are surprisingly effective at capturing global elas- tic deformation, while being easy to optimize unlike dense graph structures. We present extensive results on standard face benchmarks, as well as a new “in the wild” annotated dataset, that suggests our system advances the state-of-the- art, sometimes considerably, for all three tasks.



2 INTRODUCTION

The problem of finding and analyzing faces is a founda- tional task in computer vision. Though great strides have been made in face detection, it is still challenging to ob- tain reliable estimates of head pose and facial landmarks, particularly in unconstrained "in the wild" images.These three tasks (detection, pose estimation, and land- mark localization) have traditionally been approached as separate problems with a disparate set of techniques, such as scanning window classifiers, view-based eigenspace meth- ods, and elastic graph models. In this work, we present a single model that simultaneously advances the state-of-the- art, sometimes considerably, for all three.

Figure 1: We present a unified approach to face detection, pose estimation, and landmark estimation. Our model is based on a mixture of tree-structured part models. To evaluate all aspects of our model, we also present a new, annotated dataset of "in the wild" images obtained from Flickr.

3 MODEL

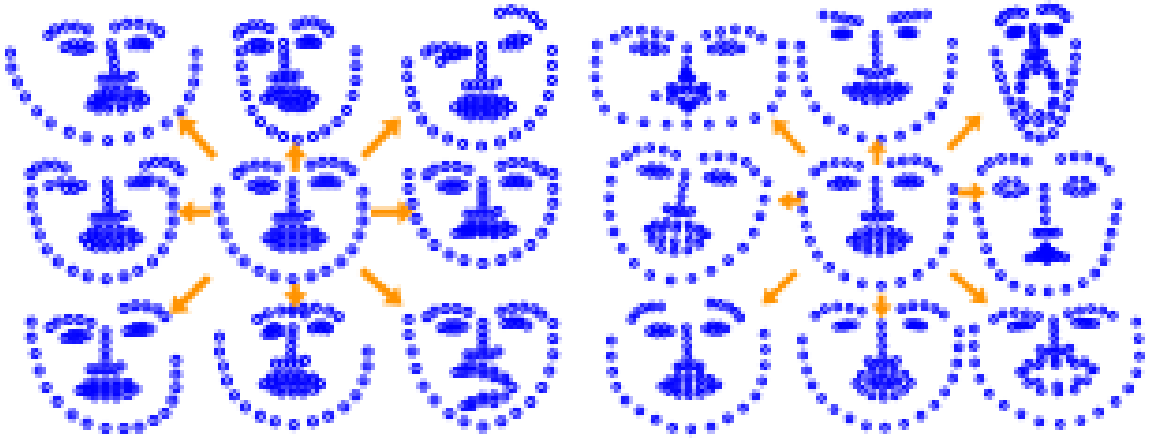
Our model is based on mixture of trees with a shared pool of parts V . We model every facial landmark as a part and use global mixtures to capture topological changes due to viewpoint.

Tree structured part model: We write each tree T m = (V m , E m) as a linearly-parameterized, tree-structured pic- torial structure., where m indicates a mixture and V m ⊆ V . Let us write I for an image, and l i = (x i , y i) for the pixel location of part i.

$$S(I, L, m) = \text{App}_m(I, L) + \text{Shape}_m(L) + \alpha^m \tag{1}$$

$$\text{App}_m(I, L) = \sum_{i \in V_m} w_i^m \cdot \phi(I, l_i) \tag{2}$$

$$\text{Shape}_m(L) = \sum_{ij \in E_m} a_{ij}^m dx^2 + b_{ij}^m dx + c_{ij}^m dy^2 + d_{ij}^m dy \tag{3}$$



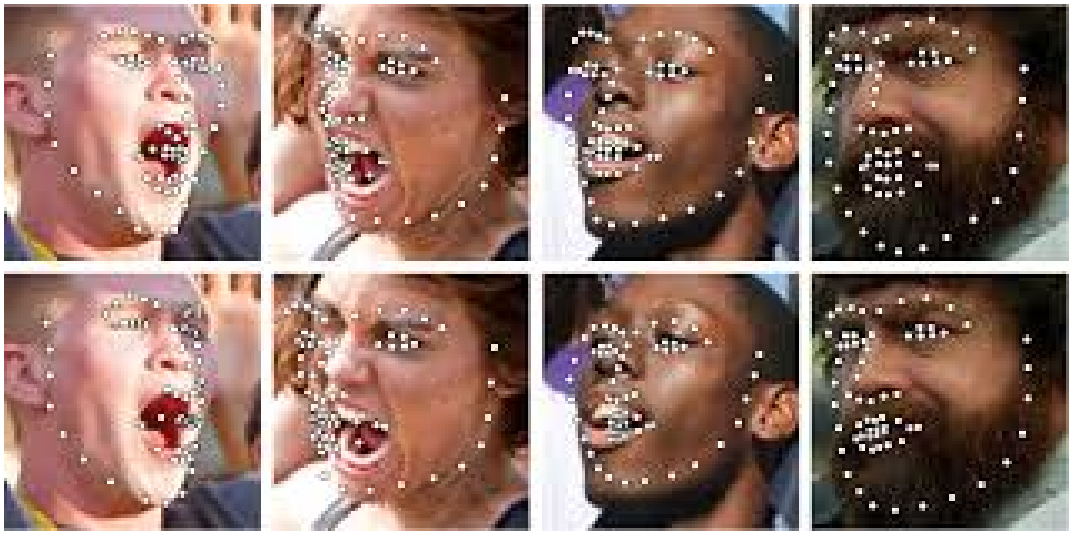
(a) Tree-based SVM (b) AAM

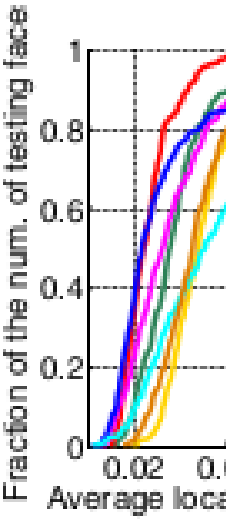
Figure 3: In (a), we show the mean shape μ_{rt} and deformation modes (eigenvectors of Λ_{rt}) learned in our tree-structured, max-margin model. In (b), we show the mean shape and deformation modes of the full-covariance Gaussian shape model used by AAMs. Note we exaggerate the deformations for visualization purposes. Model (a) captures much of the relevant elastic deformation, but produces some unnatural deformations because it lacks loopy spatial constraints (e.g., the left corner of the mouth in the lower right plot). Even so, it still outperforms model (b), presumably because it is easier to optimize and allows for joint, discriminative training of part appearance models.

4 Computation

The total number of distinct part tem- plates in our vocabulary is M 0 —V —. Assuming each part is of dimension D and assuming there exist N candidate part locations, the total cost of evaluating all parts at all locations is O(DN M 0 —V —). Using distance transforms [14], the cost of message passing is O(N M —V —). This makes our over- all model linear in the number of parts and the size of the image, similar to other models such as AAMs and CLMs. Because the distance transform is rather efficient and D is large, the first term (local part score computation) is the computational bottleneck. Our fully independent model uses M 0 = M , while our fully-shared model uses M 0 = 1, roughly an order of magnitude difference. In our experi- mental results, we show that our fully-shared model may still be practically useful as it sacrifices some performance for speed. This means our multiview model can run as fast as a single-view model. Moreover, since single-view CLMs often pre-process their images to compute dense local part scores, our multiview model is similar in speed to such popular approaches but globally-optimizable.

Figure 7: Qualitative results of our model on AFW images, tuned for an equal error rate of false positives and missed detections. We accurately detect faces, estimate pose, and estimate deformations in cluttered, real-world scenes.





(a) Localization

Figure 9: (a) the frontal fac end are the pe less than .05 (produces such set. (b) Landm Each ellipse d tion errors.

5 WHAT WE HAVE DONE

STEP 1: Localize the face in the image.

STEP 2: Detect the Key facial structures on the face ROI.

5.1 Localize the face in the image

The pre-trained facial landmark detector inside the dlib library is used to estimate the location of 68 (x, y)-coordinates that map to facial structures on the face.

5.2 Detect the Key facial structures on the face ROI.

There are a variety of facial landmark detectors, but all methods essentially try to localize and label the following facial regions: eg - mouth, eyebrow, eye, nose, jaw etc.

The facial landmark detector used is an implementation of the One Millisecond Face Alignment with an Ensemble of Regression Trees paper by Kazemi and Sullivan (2014).

The end result is a facial landmark detector that can be used to detect facial landmarks in real-time with high quality predictions.

6 What Next...

We will use different face and feature detectors to improve the accuracy of estimation.

Implementation of pose detection.