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

Xiangxin Zhu Deva Ramanan
Dept. of Computer Science, University of California, Irvine
{xzhu,dramanan}@ics.uci.edu

Abstract

We present a unified model for face detection, pose estimation, 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 elastic 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. Though our model is modestly trained with hundreds of faces, it compares favorably to commercial systems trained with billions of examples (such as Google Picasa and face.com).

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.

1. Introduction

The problem of finding and analyzing faces is a foundational task in computer vision. Though great strides have been made in face detection, it is still challenging to obtain reliable estimates of head pose and facial landmarks, particularly in unconstrained "in the wild" images. Ambiguities due to the latter are known to be confounding factors for face recognition [41]. Indeed, even face detection is arguably still difficult for extreme poses.

These three tasks (detection, pose estimation, and landmark localization) have traditionally been approached as separate problems with a disparate set of techniques, such as scanning window classifiers, view-based eigenspace methods, 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. We argue that a unified approach may make the problem easier; for example, much work on landmark localization assumes images are pre-filtered by a face detector, and so suffers from a near-frontal bias.

Our model is a novel but simple approach to encoding elastic deformation and three-dimensional structure; we use mixtures of trees with a shared pool of parts (see Figure 1). We define a "part" at each facial landmark and use global mixtures to model topological changes due to viewpoint; a part will only be visible in certain mixtures/views. We allow different mixtures to share part templates. This allows us to model a large number of views with low complexity. Finally, all parameters of our model, including part templates, modes of elastic deformation, and view-based topology, are discriminatively trained in a max-margin framework.

Notably, most previous work on landmark estimation use densely-connected elastic graphs [38, 9] which are difficult to optimize. Consequently, much effort in the area has focused on optimization algorithms for escaping local minima. We show that multi-view trees are an effective alternative because (1) they can be globally optimized with dynamic programming and (2) surprisingly, they still capture much relevant global elastic structure.

We present an extensive evaluation of our model for face detection, pose estimation, and landmark estimation. We compare to the state-of-the-art from both the academic community and commercial systems such as Google Picasa

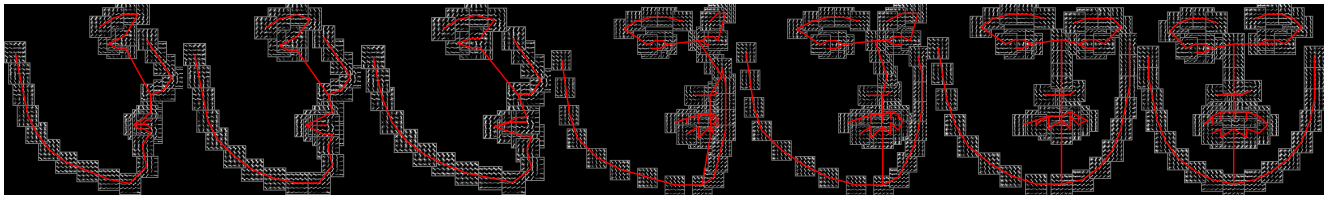


Figure 2: Our mixture-of-trees model encodes topological changes due to viewpoint. Red lines denote springs between pairs of parts; note there are no closed loops, maintaining the tree property. All trees make use of a common, shared pool of part templates, which makes learning and inference efficient.

[1] and face.com [2] (the best-performing system on LFW benchmark [3]). We first show results in controlled lab settings, using the well-known MultiPIE benchmark [16]. We definitively outperform past work in all tasks, particularly so for extreme viewpoints. As our results saturate this benchmark, we introduce a new "in the wild" dataset of Flickr images annotated with faces, poses, and landmarks. In terms of face detection, our model substantially outperforms ViolaJones, and is on par with the commercial systems above. In terms of pose and landmark estimation, our results dominate even commercial systems. Our results are particularly impressive since our model is trained with hundreds of faces, while commercial systems use up to billions of examples [35]. Another result of our analysis is evidence of large gap between currently-available academic solutions and commercial systems; we will address this by releasing open-source software.

2. Related Work

As far as we know, no previous work jointly addresses the tasks of face detection, pose estimation, and landmark estimation. However, there is a rich history of all three in vision. Space does not allow for a full review; we refer the reader to the recent surveys [41, 26, 39]. We focus on methods most related to ours.

Face detection is dominated by discriminatively-trained scanning window classifiers [32, 22, 27, 18], most ubiquitous of which is the Viola Jones detector [37] due its open-source implementation in the OpenCV library. Our system is also trained discriminatively, but with much less training data, particularly when compared to commercial systems.

Pose estimation tends to be addressed in a video scenario [41], or a controlled lab setting that assumes the detection problem is solved, such as the MultiPIE [16] or FERET [31] benchmarks. Most methods use explicit 3D models [6, 17] or 2D view-based models [30, 10, 21]. We use view-based models that share a central pool of parts. From this perspective, our approach is similar to aspect-graphs that reason about topological changes between 2D views of an object [7].

Facial landmark estimation dates back to the classic approaches of Active Appearance Models (AAMs) [9] and elastic graph matching [25, 38]. Recent work has focused on global spatial models built on top of local part detectors, sometimes known as Constrained Local Models (CLMs) [11, 34, 5]. Notably, all such work assumes a densely connected spatial model, requiring the need for approximate matching algorithms. By using a tree model, we can use efficient dynamic programming algorithms to find globally optimal solutions.

From a modeling perspective, our approach is similar to those that reason about mixtures of deformable part models [14, 40]. In particular [19] use mixtures of trees for face detection and [13] use mixtures of trees for landmark estimation. Our model simultaneously addresses both with state-of-the-art results, in part because it is aggressively trained to do so in a discriminative, max-margin framework. We also explore part sharing for reducing model size and computation, as in [36, 28].

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. We show such mixtures for viewpoint in Fig.2. We will later show that global mixtures can also be used to capture gross deformation changes for a single viewpoint, such as changes in expression.

Tree structured part model: We write each tree $T_m = (V_m, E_m)$ as a linearly-parameterized, tree-structured pictorial structure [40], where m indicates a mixture and $V_m \subseteq V$. Let us write I for an image, and $l_i = (x_i, y_i)$ for the pixel location of part i. We score a configuration of parts $L = \{l_i : i \in V\}$ as:

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

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

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

Eqn.2 sums the appearance evidence for placing a template w_i^m for part i, tuned for mixture m, at location l_i. We write $\phi(I, l_i)$ for the feature vector (e.g., HoG descriptor) extracted from pixel location l_i in image I. Eqn.3 scores

the mixture-specific spatial arrangement of parts L, where $dx = x_i - x_j$ and $dy = y_i - y_j$ are the displacement of the ith part relative to the jth part. Each term in the sum can be interpreted as a spring that introduces spatial constraints between a pair of parts, where the parameters (a, b, c, d) specify the rest location and rigidity of each spring. We further analyze our shape model in Sec.3.1. Finally, the last term α^m is a scalar bias or "prior" associated with viewpoint mixture m.

Part sharing: Eqn.1 requires a separate template w_i^m for each mixture/viewpoint m of part i. However, parts may look consistent across some changes in viewpoint. In the extreme cases, a "fully shared" model would use a single template for a particular part across all viewpoints, $w_i^m = w_i$. We explore a continuum between these two extremes, written as $w_i^{f(m)}$, where $f(m)$ is a function that maps a mixture index (from 1 to M) to a smaller template index (from 1 to M'). We explore various values of M': no sharing ($M' = M$), sharing across neighboring views, and sharing across all views ($M' = 1$).

3.1. Shape model

In this section, we compare our spatial model with a standard joint Gaussian model commonly used in AAMs and CLMs [11, 34]. Because the location variables l_i in Eqn.3 only appear in linear and quadratic terms, the shape model can be rewritten as:

$$\text{Shape}_m(L) = -(L - \mu_m)^T \Lambda_m (L - \mu_m) + \text{constant} \quad (4)$$

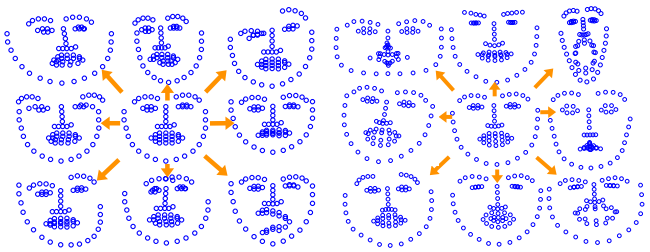
where (μ, Λ) are re-parameterizations of the shape model (a, b, c, d); this is akin to a canonical versus natural parameterization of a Gaussian. In our case, Λ_m is a block sparse precision matrix, with non-zero entries corresponding to pairs of parts i, j connected in E_m. One can show Λ_m is positive semidefinite if and only if the quadratic spring terms a and c are negative [33]. This corresponds to a shape score that penalizes configurations of L that deform from the ideal shape μ_m. The eigenvectors of Λ_m associated with the smallest eigenvalues represent modes of deformation associated with small penalties. Notably, we discriminatively train (a, b, c, d) (and hence μ and Λ) in a max-margin framework. We compare our learned shape models with those trained generatively with maximum likelihood in Fig.3.

4. Inference

Inference corresponds to maximizing $S(I, L, m)$ in Eqn.1 over L and m:

$$S^*(I) = \max_m [\max_L S(I, L, m)] \quad (5)$$

Simply enumerate all mixtures, and for each mixture, find the best configuration of parts. Since each mixture $T_m = (V_m, E_m)$ is a tree, the inner maximization can be done efficiently with dynamic programming(DP) [15]. We omit the message passing equations for a lack of space.



(a) Tree-based SVM (b) AAM

Figure 3: In (a), we show the mean shape μ_m and deformation modes (eigenvectors of Λ_m) 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.

Computation: The total number of distinct part templates in our vocabulary is $M'|V|$. 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 $O(DNM'|V|)$. Using distance transforms [14], the cost of message passing is $O(NM|V|)$. This makes our overall 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 $M' = M$, while our fully-shared model uses $M' = 1$, roughly an order of magnitude difference. In our experimental 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 [34], our multiview model is similar in speed to such popular approaches but globally-optimizable.

5. Learning

To learn our model, we assume a fully-supervised scenario, where we are provided positive images with landmark and mixture labels, as well as negative images without faces. We learn both shape and appearance parameters discriminatively using a structured prediction framework. We

first need to estimate the edge structure E_m of each mixture. While trees are natural for modeling human bodies [15, 40, 19], the natural tree structure for facial landmarks is not clear. We use the Chow-Liu algorithm [8] to find the maximum likelihood tree structure that best explains the landmark locations for a given mixture, assuming landmarks are Gaussian distributed.

Given labeled positive examples $\{I_n, L_n, m_n\}$ and negative examples $\{I_n\}$, we will define a structured prediction objective function similar to one proposed in [40]. To do so, let's write $z_n = \{L_n, m_n\}$. Note that the scoring function Eqn.1 is linear in the part templates w, spring parameters (a, b, c, d), and mixture biases α. Concatenating these parameters into a single vector β, we can write the score as:

$$S(I, z) = \beta \cdot \Phi(I, z) \tag{6}$$

The vector $\Phi(I, z)$ is sparse, with nonzero entries in a single interval corresponding to mixture m.

Now we can learn a model of the form:

$$\begin{aligned} \arg\min_{\beta, \xi_n \geq 0} \quad & \frac{1}{2}\beta \cdot \beta + C \sum_n \xi_n \\ \text{s.t.} \quad \forall n \in \text{pos} \quad & \beta \cdot \Phi(I_n, z_n) \geq 1 - \xi_n \\ \forall n \in \text{neg}, \forall z \quad & \beta \cdot \Phi(I_n, z) \leq -1 + \xi_n \\ \forall k \in K, \quad & \beta_k \leq 0 \end{aligned} \tag{7}$$

The above constraint states that positive examples should score better than 1 (the margin), while negative examples, for all configurations of part positions and mixtures, should score less than -1. The objective function penalizes violations of these constraints using slack variables ξ_n. We write K for the indices of the quadratic spring terms (a, c) in parameter vector β. The associated negative constraints ensure that the shape matrix Λ is positive semidefinite (Sec.3.1). We solve this problem using the dual coordinate-descent solver in [40], which accepts negativity constraints.

6. Experimental Results

CMU MultiPIE: CMU MultiPIE face dataset [16] contains around 750,000 images of 337 people under multiple viewpoints, different expressions and illumination conditions. Facial landmark annotations (68 landmarks for frontal faces ($-45°$ to $45°$), and 39 landmarks for profile faces) are available from the benchmark curators for a relatively small subset of images. In our experiments, we use 900 faces from 13 viewpoints spanning over $180°$ spacing at $15°$ for training, and another 900 faces for testing. 300 of those faces are frontal, while the remaining 600 are evenly distributed among the remaining viewpoints. Hence our training set is considerably smaller than those typically used for training face detectors. Fig. 5 shows example images from all the 13 viewpoints with the annotated landmarks.

Figure 4: Example images from our annotated faces-in-the-wild (AFW) testing set.

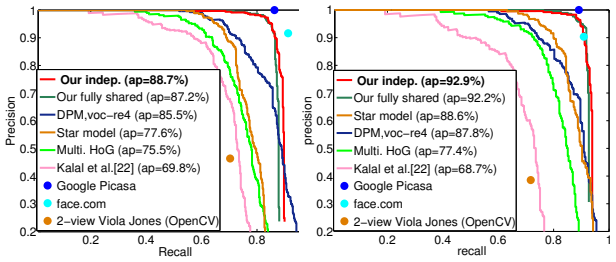
Our annotated face in-the-wild (AFW) testset: To further evaluate our model, we built an annotated faces in-the-wild (AFW) dataset from Flickr images (Fig. 4). We randomly sampled images, keeping each that contained at least one large face. This produced a 205-image dataset with 468 faces. Images tend to contain cluttered backgrounds with large variations in both face viewpoint and appearance (aging, sunglasses, make-ups, skin color, expression etc.). Each face is labeled with a bounding box, 6 landmarks (the center of eyes, tip of nose, the two corners and center of mouth) and a discretized viewpoint ($-90°$ to $90°$ every $15°$) along pitch and yaw directions and (left, center, right) viewpoints along the roll direction. Our dataset differs from similar "in-the-wild" collections [20, 3, 23, 5] in its annotation of multiple, non-frontal faces in a single image.

Models: We train our models using 900 positive examples from MultiPIE, and 1218 negative images from the INRIAPerson database [12] (which tend to be outdoor scenes that do not contain people). We model each landmark defined in MultiPIE as a part. There are a total of 99 parts across all viewpoints. Each part is represented as a 5×5 HoG cells with a spatial bin size of 4. We use 13 viewpoints and 5 expressions limited to frontal viewpoints, yielding a total of 18 mixtures. For simplicity, we do not enforce symmetry between left/right views.

Sharing: We explore 4 levels of sharing, denoting each model with the number of distinct templates encoded. *Share-99* (i.e. fully shared model) shares a single template for each part across all mixtures. *Share-146* shares templates across only viewpoints with identical topology $\{-45 : 45\}$, $\pm\{60 : 90\}$. *Share-622* shares templates across neighboring viewpoints. *Independent-1050* (i.e. independent model) does not share any templates across any mixtures. We score both the view-specific mixture and part locations returned by our various models.

Computation: Our 4 models tend to have consistent relative performance across our datasets and evaluations, with *Independent-1050* performing the best, taking roughly 40

Figure 5: Example images from MultiPIE with annotated landmarks.



(a) all faces (b) large faces only

Figure 6: Precision-recall curves for face detection on our AFW testset (a) on all faces; (b) on faces larger than 150×150 pixels. Our models significantly outperform popular detectors in use and are on par with commercial systems trained with billions of examples, such as Google Picasa and face.com.

seconds per image, while *Share-99* performs slightly worse but roughly $10\times$ faster. We present quantitative results later in Fig.12. With parallel/cascaded implementations, we believe our models could be real-time. Due to space restrictions, we mainly present results for these two "extreme" models in this section.

In-house baselines: In addition to comparing with numerous other systems, we evaluate two restricted versions of our approach. We define *Multi.HoG* to be rigid, multiview HoG template detectors, trained on the same data as our models. We define *Star Model* to be equivalent to *Share-99* but defined using a "star" connectivity graph, where all parts are directly connected to a root nose part. This is similar to the popular star-based model of [14], but trained in a supervised manner given landmark locations.

6.1. Face detection

We show detection results for AFW, since MultiPIE consists of centered faces. We adopt the PASCAL VOC precision-recall protocol for object detection (requiring 50% overlap). We compare our approach and baselines with the following: (1) OpenCV frontal + profile Viola-Jones, (2) Boosted frontal + profile face detector of [22], (3) Deformable part model (DPM) [14, 4] trained on same data as our models, (4) Google Picasa's face detector, manually scored by inspection, (5) face.com's face detector, which reports detections, viewpoints, and landmarks. To generate an overly-optimistic multiview detection baseline for (1) and (2), we calibrated the frontal and side detectors *on the test set* and applied non-maximum suppression (NMS) to generate a final set of detections.

Results on all faces are summarized in Fig.6a. Our models outperform 2-view Viola-Jones and [22] significantly, and are only slightly below Google Picasa and face.com. Our face model is tuned for large faces such that landmarks are visible. We did another evaluation of all algorithms, including baselines, on faces larger than 150 pixels in height (a total of 329, or 70% of the faces in AFW). In this case, our model is on par with Google Picasa and face.com (Fig.6b). We argue that high-resolution images are rather common given HD video and megapixel cameras. One could define multiresolution variants of our models designed for smaller faces [29], but we leave this as future work.

Fig.6b reveals an interesting progression of performance. Surprisingly, our rigid multiview HoG baseline outperforms popular face detectors currently in use, achieving an average precision (AP) of 77.4%. Adding latent star-structured parts, making them supervised, and finally adding tree-structured relations each contributes to performance, with APs of 87.8%, 88.6%, and 92.9% respectively.

A final point of note is the large gap in performance between current academic solutions and commercial systems. We will address this discrepancy by releasing open-source software.

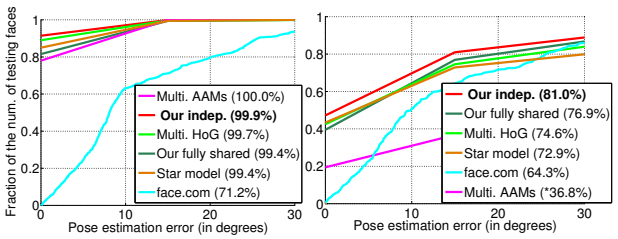
6.2. Pose estimation

We compare our approach and baselines with the following: (1) Multiview AAMs: we train an AAM for each viewpoint using the code from [24], and report the view-specific model with the smallest reconstruction error on a test image. (2) face.com.

Fig.8 shows the cumulative error distribution curves on both datasets. We report the fraction of faces for which the estimated pose is within some error tolerance. Our independent model works best, scoring 91.4% when requiring exact matching, and 99.9% when allowing $\pm 15^\circ$ error tolerance on MultiPIE. In general, we find that many methods saturate in performance on MultiPIE, originally motivating us to collect AFW.

Unlike on MultiPIE where we assume detections are given (as faces are well centered in image), we evaluate the performance on AFW in a more realistic manner: we evaluate results on faces found by a given algorithm and count missed detections as having an infinite error in pose estimation. Because AAMs do not have an associated detector, we given them the best-possible initialization with the ground-truth bounding box *on the test set* (denoted with an $*$ in Fig.8b).

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) MultiPIE (b) AFW

Figure 8: Cumulative error distribution curves for pose estimation. The numbers in the legend are the percentage of faces that are correctly labeled within $\pm 15^\circ$ error tolerance. AAMs are initialized with ground-truth bounding boxes (denoted by *). Even so, our independent model works best on both MultiPIE and AFW.

All curves decrease in performance in AFW (indicating the difficulty of the dataset), especially multiview AAMs, which suggests AAMs generalize poorly to new data. Our independent model again achieves the best performance, correctly labeling 81.0% of the faces within $\pm 15^\circ$ error tolerance. In general, our models and Multiview-HoG/Star baselines perform similarly, and collectively outperform face.com and Multiview AAMs by a large margin. Note that we don't penalize false positives for pose estimation; our Multiview-HoG/Star baselines would perform worse if we penalized false positives as incorrect pose estimates (because they are worse detectors). Our results are impressive given the difficulty of this unconstrained data.

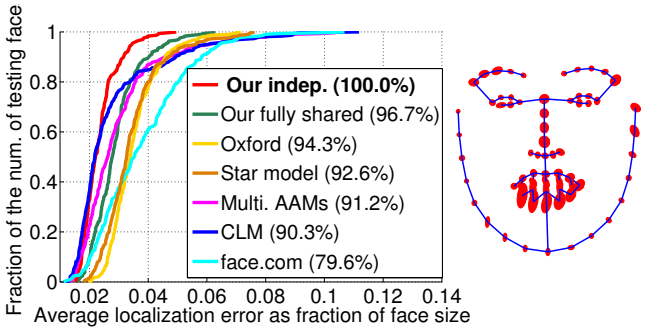
6.3. Landmark localization

We compare our approach and baselines with the following: (1) Multiview AAMs (2) Constrained local model (CLM): we use the off-the-shelf code from [34]. This work represents the current state-of-the-art results on landmark estimation in MultiPIE. (3) face.com reports the location of a few landmarks, we use 6 as output: eye centers, nose tip, mouth corners and center. (4) Oxford facial landmark detector [13] reports 9 facial landmarks: corners of eyes, nostrils, nose tip and mouth corners. Both CLM and multiview AAMs are carefully initialized using the ground truth bounding boxes *on the test set*.

Landmark localization error is often normalized with respect to the inter-ocular distance [5]; this however, presumes both eyes are visible. This is not always true, and reveals the bias of current approaches for frontal faces! Rather, we normalize pixel error with respect to the face size, computed as the mean of height and width.

Various algorithms assume different landmark sets; we train linear regressors to map between these sets. On AFW, we evaluate algorithms using a set of 6 landmarks common to all formats. On MultiPIE, we use the original 68 landmarks when possible, but evaluate face.com and Oxford using a subset of landmarks they report; note this gives them an extra advantage because their localization error tends to be smaller since they output fewer degrees of freedom.

We first evaluate performance on only frontal faces from MultiPIE in Fig.9a. All baselines perform well, but our independent model (average error of 4.39 pixels/ 2.3% relative error) still outperforms the state-of-the-art CLM model from [34] (4.75 pixels/ 2.8%). When evaluated on all view points, we see a performance drop across most baselines, particularly CLMs (Fig.10a). It is worth noting that, since CLM and Oxford are trained to work on near-frontal faces, we only evaluate them on faces between -45° and 45°



(a) Localization results on frontal faces from MultiPIE (b)

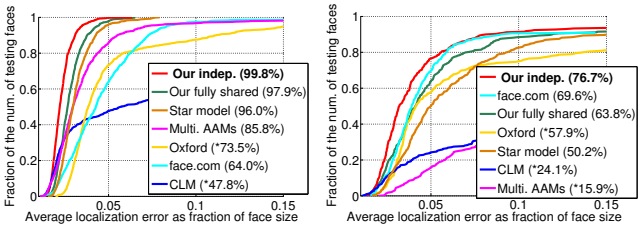
Figure 9: (a) Cumulative localization error distribution of the frontal faces from MultiPIE. The numbers in the legend are the percentage of faces whose localization error is less than .05 (5%) of the face size. Our independent model produces such a small error for *all* (100%) faces in the testset. (b) Landmark-specific error of our independent model. Each ellipse denotes the standard deviation of the localization errors.

where all frontal landmarks are visible (marked as a $*$ in Fig.10a). Even given this advantage, our model outperforms all baselines by a large margin.

On AFW (Fig.10b), we again realistically count missed detections as having a localization error of infinity. We report results on large faces where landmarks are clearly visible (which includes 329 face instances in AFW testset). Again, our independent model achieves the best result with 76.7% of faces having landmark localization error below 5% of face size. AAMs and CLM's accuracy plunges, which suggests these popular methods don't generalize well to in-the-wild images. We gave an advantage to AAM, CLM, and Oxford by initializing them with ground truth bounding boxes *on the test set* (marked with "$*$" in Fig.10b). Finally, the large gap between our models and our Star baseline suggests that our tree structure does capture useful elastic structure.

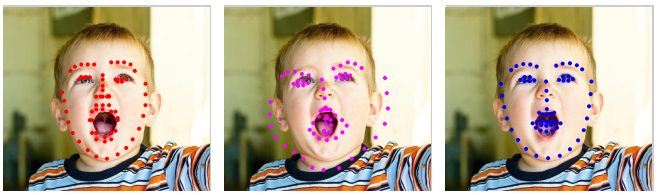
Our models outperform the state-of-the-art on both datasets. We outperform all methods by a large margin on MultiPIE. The large performance gap suggest our models maybe overfitting to the lab conditions of MultiPIE; this in turn suggests they may do even better if trained on "in-the-wild" training data similar to AFW. Our model even outperforms commercial systems such as face.com. This result is surprising since our model is only trained with 900 faces, while the latter appears to be trained using billions of faces [35].

Fig.9b plots the landmark specific localization error of our independent model on frontal faces from MultiPIE. Note that the errors around the mouth are asymmetric, due to the asymmetric spatial connectivity required by a tree-structure. This suggests our model may still benefit from additional loopy spatial constraints. However, our model still generates fairly accurate localizations even compared to baselines encoding such dense spatial constraints - we show an example AFW image with large mouth deformations in Fig.11.



(a) MultiPIE (b) AFW

Figure 10: Cumulative error distribution curves for landmark localization. The numbers in legend are the percentage of testing faces that have average error below 0.05(5%) of the face size. (*) denote models which are given an "unfair" advantage, such as hand-initialization or a restriction to near-frontal faces (described further in the text). Even so, our independent model works the best on both MultiPIE and our AFW testset.



(a) Our model (b) AAM (c) CLM

Figure 11: An example AFW image with large mouth deformations. AAMs mis-estimate the overall scale in order to match the mouth correctly. CLM matches the face contour correctly, but sacrifices accuracy at the nose and mouth. Our tree-structured model is flexible enough to capture large face deformation and yields the lowest localization error.

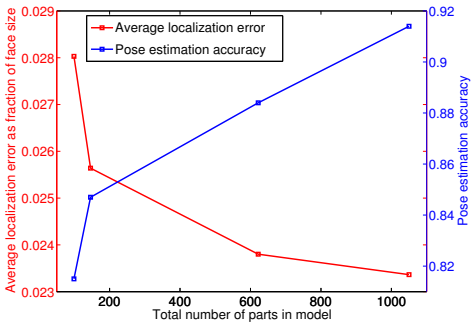


Figure 12: We show how different levels of sharing (as described at the beginning of Sec.6) affect the performance of our models on MultiPIE. We simultaneously plot localization error in red (lower is better) and pose estimation accuracy in blue (higher is better), where poses need to be predicted with zero error tolerance. The larger number of part templates indicate less sharing. The fully independent model works best on both tasks.

Conclusion: We present a unified model for face detection, pose estimation and landmark localization using a mixture of trees with a shared pool of parts. Our tree models are surprisingly effective in capturing global elastic deformation, while being easy to optimize. Our model outperforms state-of-the-art methods, including large-scale commercial systems, on all three tasks under both constrained and in-the-wild environments. To demonstrate the latter, we present a new annotated dataset which we hope will spur further progress.

Acknowledgements: Funding for this research was provided by NSF Grant 0954083, ONR-MURI Grant N00014-10-1-0933, and support from Intel.

References

[1] http://picasa.google.com/. 2

[2] http://face.com/. 2

[3] http://vis-www.cs.umass.edu/lfw/results.html. 2, 4

[4] http://www.cs.brown.edu/~pff/latent/voc-release4.tgz. 5

[5] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. In *CVPR 2011*. 2, 4, 6

[6] V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model. *IEEE TPAMI*, 2003. 2

[7] K. Bowyer and C. Dyer. Aspect graphs: An introduction and survey of recent results. *International Journal of Imaging Systems and Technology*, 1990. 2

[8] C. Chow and C. Liu. Approximating discrete probability distributions with dependence trees. *IEEE TIT*, 1968. 4

[9] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *IEEE TPAMI*, 2001. 1, 2

[10] T. Cootes, K. Walker, and C. Taylor. View-based active appearance models. In *IEEE FG 2000*, 2000. 2

[11] D. Cristinacce and T. Cootes. Feature detection and tracking with constrained local models. In *BMVC 2006*. 2, 3

[12] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR 2005*. 4

[13] M. Everingham, J. Sivic, and A. Zisserman. "Hello! My name is... Buffy" – automatic naming of characters in TV video. In *BMVC 2006*. 2, 6

[14] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE TPAMI*, 2009. 2, 3, 5

[15] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 2005. 3, 4

[16] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *Image and Vision Computing*, 2010. 2, 4

[17] L. Gu and T. Kanade. 3d alignment of face in a single image. In *CVPR 2006*. 2

[18] B. Heisele, T. Serre, and T. Poggio. A Component-based Framework for Face Detection and Identification. *IJCV*, 2007. 2

[19] S. Ioffe and D. Forsyth. Mixtures of trees for object recognition. In *CVPR 2001*. 2, 4

[20] V. Jain and E. Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. Technical Report UM-CS-2010-009, University of Massachusetts, Amherst, 2010. 4

[21] M. Jones and P. Viola. Fast multi-view face detection. In *CVPR 2003*. 2

[22] Z. Kalal, J. Matas, and K. Mikolajczyk. Weighted sampling for large-scale boosting. In *BMVC 2008*. 2, 5

[23] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *First IEEE International Workshop on Benchmarking Facial Image Analysis Technologies 2011*. 4

[24] D.-J. Kroon. Active shape model and active appearance model. http://www.mathworks.com/matlabcentral/fileexchange/26706. 5

[25] T. Leung, M. Burl, and P. Perona. Finding faces in cluttered scenes using random labeled graph matching. In *ICCV 1995*. 2

[26] E. Murphy-Chutorian and M. Trivedi. Head pose estimation in computer vision: A survey. *IEEE TPAMI*, 2009. 2

[27] M. Osadchy, Y. L. Cun, and M. L. Miller. Synergistic face detection and pose estimation with energy-based models. *JMLR*, 2007. 2

[28] P. Ott and M. Everingham. Shared parts for deformable part-based models. In *CVPR 2011*. 2

[29] D. Park, D. Ramanan, and C. Fowlkes. Multiresolution models for object detection. In *ECCV 2010*. 5

[30] A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. In *CVPR 1994*. 2

[31] P. Phillips, H. Moon, S. Rizvi, and P. Rauss. The feret evaluation methodology for face-recognition algorithms. *IEEE TPAMI*, 2000. 2

[32] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE TPAMI*, 1998. 2

[33] H. Rue and L. Held. *Gaussian Markov random fields: theory and applications*, volume 104. Chapman & Hall, 2005. 3

[34] J. Saragih, S. Lucey, and J. Cohn. Deformable model fitting by regularized landmark mean-shift. *IJCV*, 2011. 2, 3, 6

[35] Y. Taigman and L. Wolf. Leveraging Billions of Faces to Overcome Performance Barriers in Unconstrained Face Recognition. *ArXiv e-prints*, Aug. 2011. 2, 7

[36] A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing visual features for multiclass and multiview object detection. *IEEE TPAMI*. 2

[37] P. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 2004. 2

[38] L. Wiskott, J.-M. Fellous, N. Kuiger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE TPAMI*, Jul 1997. 1, 2

[39] M.-H. Yang, D. Kriegman, and N. Ahuja. Detecting faces in images: a survey. *IEEE TPAMI*, 2002. 2

[40] Y. Yang and D. Ramanan. Articulated pose estimation using flexible mixtures of parts. In *CVPR 2011*, 2011. 2, 4

[41] W. Zhao, R. Chellappa, P. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 2003. 1, 2