Nonparametric Context Modeling of Local Appearance for Pose- and Expression-Robust Facial Landmark Localization

Brandon M. Smith[1] Jonathan Brandt[2] Zhe Lin[2] Li Zhang[1]

[1]University of Wisconsin–Madison [2]Adobe Research

http://www.cs.wisc.edu/~lizhang/projects/face-landmark-localization/

Abstract

We propose a data-driven approach to facial landmark localization that models the correlations between each landmark and its surrounding appearance features. At runtime, each feature casts a weighted vote to predict landmark locations, where the weight is precomputed to take into account the feature's discriminative power. The feature voting-based landmark detection is more robust than previous local appearance-based detectors; we combine it with nonparametric shape regularization to build a novel facial landmark localization pipeline that is robust to scale, in-plane rotation, occlusion, expression, and most importantly, extreme head pose. We achieve state-of-the-art performance on two especially challenging in-the-wild datasets populated by faces with extreme head pose and expression.

1. Introduction

Facial landmark localization seeks to automatically locate predefined facial landmarks (*e.g.*, the nose tip, mouth corners, eye centers) in face images. It is an important research area in computer vision in part because digital face portraits are ubiquitous. Many compelling applications depend on it, including face recognition and retrieval, face animation, and face image editing wizards. At the same time, robust facial landmark localization is very challenging in practice. Real-world images can be cluttered, faces can be partially occluded, and they can exhibit large variations in appearance, shape, expression, and head pose. In particular, when current approaches fail, they often fail on faces with extreme expression and/or head pose.

One primary source for failure is the practical challenge of initialization. Many landmark localization methods rely heavily on a reasonable initialization as a prerequisite for success, and can fail to find the correct solution if the initial face shape is too far from the true optimum. A popular strategy, even for recent approaches (*e.g.*, [1, 3, 4, 19, 22] to name just a few), is to first detect the face (*i.e.*, using [21]), and then fit a mean face shape (where the shape is defined by the facial landmarks) to the detection window. For upright near-frontal faces (*i.e.*, within 30 degrees yaw, pitch, and/or roll head rotation from frontal) detection and initialization is seldom a problem. However, for extreme poses and some expressions, traditional face detectors (*e.g.*, [21]) may fail, or the true shape of the face inside the detection window will differ significantly from the initial shape, making a good initialization unlikely, thereby challenging even recent methods like [22], as we observe in our results section. This problem is often minimized in the literature, where popular evaluation datasets often make initialization easy.

Figure 1. Our robust exemplar-based algorithm locates landmarks on challenging faces with extreme head pose.

Part-based models [8, 24] can be used to address the initialization problem, but learning an accurate part graph parameterization and inferring part labels from the graph can be challenging. Recent works [26, 29] simplify the graph structure to a tree and produce impressive results. Failure cases suggest that human faces, unlike the human body, still prefer a loopy graph structure. Furthermore, a landmark graph typically only models interactions between landmarks but does not model the interactions between landmarks and non-landmark image patches.

Our goal is to accurately localize landmarks on faces with extreme head pose and/or expression. To achieve this goal, we have developed an exemplar-based approach that requires only a weak initialization. More specifically, we generalize and combine a recent exemplar-based approach for shape regularization [2] with an exemplar-based approach for face detection [20] to model context interactions between landmarks and their surrounding local appearance features in a nonparametric way. By "weak initialization" we mean that our algorithm does not require (nor does it position) a face shape for initialization. Our algorithm relies on a face detector (*i.e.*, [20]) only to establish an initial size for the face and the region of interest in the image. Our algorithm also searches over multiple face scales and rotations during landmark localization, which allows for a large margin of error

in the initial face detection. The end result is a new pipeline that achieves state-of-the-art results on unconstrained face datasets populated with challenging poses and expressions. The robustness and flexibility of our method comes from its ability to efficiently and effectively leverage the information from a large database of face exemplars.

We make three main contributions in this paper:

1. We propose a data-driven approach for modeling the correlations between each landmark and its surrounding appearance features. At runtime, each feature casts a *weighted* vote to predict landmark locations, where the weight is precomputed to take into account the feature's discriminative power.

2. We combine nonparametric local appearance modeling with nonparametric shape regularization to build a novel facial landmark localization pipeline that is robust to common types of real-world variation, including scale, rotation, occlusion, expression, and most importantly, extreme head pose.

3. We compare our approach to several recent approaches and show state-of-the-art accuracy on two especially challenging in-the-wild datasets populated by faces with extreme head pose and expression.

2. Related Work

Early facial landmark localization and face alignment methods, *e.g.*, Active Shape Models (ASMs) [6] and Active Appearance Models (AAMs) [5], relied on global parametric models for face shape and appearance. Parametric models work well for favorable face images, *e.g.*, where the illumination, pose, and expression do not vary greatly.

To overcome the well-known generalization problem in AAMs, Zhao *et al*. [27] proposed computing a separate AAM for each test face using k-nearest neighbor training faces (w.r.t. the test face) rather than all training faces. Using k-NN exemplars is an important part of our approach (see Section 3.3) and others [2, 20, 28], although it is not our main contribution. Like other AAM-based methods, [27] involves a gradient decent-type optimization over the whole face (holistic), which is sensitive to initialization.

Constrained Local Models (CLMs) [1, 7, 19] handle a wider range of faces than generic AAMs by employing an ensemble of local texture patches or landmark detectors that are constrained by a global shape model. The local appearance models are more robust to a range of challenges including occlusion and global illumination changes, but CLMs still rely on parametric shape models for regularization, which may not generalize well to a broad range of poses.

Belhumeur *et al*. [2], and more recently Zhou *et al*. [28], use nonparametric global shape models, which generalize better to challenging real-world faces. However, these two methods still rely on local landmark detectors (linear support vector machines (SVMs)) to form the landmark response. Because landmark appearance changes dramatically with large head pose variation, these two methods are limited to constrained head poses.

Recent regression-based approaches [3, 22] have also demonstrated increasingly impressive performance on real-world faces without the need for a parametric shape model. Although these methods have some tolerance to inaccurate initialization, for many challenging faces, where pose can easily vary past 30 degrees yaw, pitch, or roll head rotation, a good initialization is much more difficult. As a result, descent-based methods can get stuck in local optima.

Zhu and Ramanan [29] address this initialization problem in their work. They use a tree structured part model of the face, which both detects faces and locates facial landmarks. One of the major advantages of their approache is that it can handle extreme head pose. However, their method only models pairwise landmark interactions on a tree structure. Yu *et al*. [26] speed up [29] by simplifying the mixture of parts for face detection and initial landmark localization, and they extend [29] by adding a two-step local refinement procedure, which resembles the approach in [19] followed by the optimization of several additional constraints via a gradient descent method. Like [29] and [26], we focus on locating facial landmarks without relying on a good initialization, but we model the full interactions between each landmark and its surrounding local features. That is, our context interactions are not limited to a tree structure, and our approach does not involve graphical model inference.

We are most inspired by the recent face detection work of Shen *et al*. [20], who rely on a Hough transform based feature voting scheme to transfer many face hypotheses from a large database of exemplar faces to the test image. The votes capture the appearance and geometric correlations between local image features and the face center. While [20] focuses on face detection, we focus on landmark localization, and we compute a set of sparse feature weights, which naturally amplify reliable features and suppress noisy or unreliable features across the database; our weights are tailored to each {feature, landmark, exemplar} combination and are computed in a data-driven way.

Several recent object detection [12, 17], tracking [25], and face landmark localization methods [23] also rely on similar feature voting schemes to generate object/landmark response maps. Among these methods, Yang and Patras [23] is most similar to our work. They use image patches to cast votes for the location of each facial landmark. Our core approach is much simpler than [23]: we do not train regression forests or use SVM classifiers. Instead, we simply use an approximate nearest neighbor algorithm [16] for image and feature retrieval, followed by weighted vote accumulation.

3. Our Approach

In this section we first give an overview of our pipeline followed by technical details of each step. Please see our supplementary material for a visual synopsis of our approach.

3.1. Overview

Database construction Our database is composed of a large collection of exemplars. Each exemplar has four components: a face image, a set of dense quantized SIFT [15] features, a sparse set of semantic facial landmarks corresponding to mouth corners, nose tip, chin contour, *etc*., and a unique set of weights, one weight per {feature, landmark} pair. Following the approach in [20], we quantize each SIFT descriptor using fast approximate k-means [16], which efficiently maps each descriptor to a *visual word*. The weights are an important aspect of our approach; Section 3.7 describes in more detail how they are useful and how we compute them.

Runtime Preprocessing Given a test image, we first use a state-of-the-art face detector [20] to locate the face and roughly estimate its scale. The test image is cropped to the face region, and then rescaled to approximately match the scale of the exemplar faces (scale estimation will be refined in later steps). Dense SIFT descriptors are then extracted over the test face at multiple orientations. Finally, each descriptor is quantized for efficient matching in later steps.

Step 1: Top exemplar retrieval Given a detected face region, retrieve a subset of top similar k exemplar faces from the database. The goal is to retrieve exemplars that are similar to the test face in appearance, shape, expression, and pose so that features in the exemplars will produce accurate landmark votes in the test image.

Step 2: Landmark voting For each type of landmark, generate voting maps using a multi-scale and multi-rotation generalized Hough transform [14]. Each matched feature from the top k exemplars casts a vote (for each scale and rotation) for a possible landmark location in the test image. The result is a table of voting maps for each landmark, where each table row corresponds to an in-plane rotation, and each table column corresponds to a scale estimate.

Step 3: Voting map selection Select a single voting map for each landmark. We define the score of each voting map as the height of the maximum peak in the voting map. For each landmark, we save the top-scoring voting map in each row of the table of voting maps (each row corresponds to a rotation), which gives a unique scale estimate for each {landmark, rotation} pair. Among these voting maps, we select the single global rotation (and corresponding voting maps) that produce the maximum total score across all landmarks.

Step 4: Final landmark estimation Given a single voting map for each landmark, estimate a final set of landmark locations. Due to local ambiguities, noise, occlusions, *etc*. each voting map may contain multiple peaks. We employ a robust nonparametric shape regularization technique [2] that avoids false peaks in the voting maps and estimates a final arrangement of landmarks.

3.2. Database construction

We use 17685 images exclusively from the Multi-PIE Face Database [9] as our exemplars. The Multi-PIE authors annotated 4685 face images, each with 68 landmarks (or 39 for profile faces). Some Multi-PIE faces have landmarks for one pose (*e.g*., -30 degrees yaw), but not the opposite (*e.g*., +30 degrees yaw). For such faces we synthesized the 'missing' landmarks by horizontally flipping the labeled image and its landmarks. 400 additional Multi-PIE images were labeled by [29]; we manually labeled 960 more (mostly non-frontal faces with non-neutral expressions). We used each unique ground-truth face shape on multiple images with the same {session, subject, expression, pose} combination, but different lighting, to obtain 17685 labeled images. Prior to feature extraction and quantization, we used Procrustes analysis to align all exemplar faces. Please see our project website for our complete set of landmark annotations and a script for generating our exemplar dataset.

3.3. Step 1: Top exemplar retrieval

In order to transfer landmarks from the database to the test image, the shape and appearance of the exemplar faces and the test face should not be drastically different. For example, a left-profile face has a much different shape and appearance than a right-profile face; there are few feature-landmark correlations between the two. We therefore select a top subset of exemplars for further processing.

Many strategies exist for retrieving similar face images from a database. We use our generalized Hough transform framework to score each exemplar image. First, we use a bag-of-words score to efficiently select the top 3000 exemplars. Next, we use the features on the test face to vote for the center of each exemplar face among the top 3000. The final score for each exemplar is the height of the maximum peak in the voting map associated with each exemplar face. We sort the scores, and select the top $k = 200$. Shen *et al*. [20] adopt a similar strategy for retrieving exemplar faces in the validation step of their face detection algorithm.

3.4. Step 2: Landmark voting

For efficiency, rather than exhaustively sliding each exemplar over the test image, we use quantized features and employ an inverted index file to efficiently retrieve matched features from the top k exemplars. When a feature in the test image is matched with an exemplar feature, the feature-to-landmark offset in the exemplar is transferred to the test image. The offset vector extends from the test feature toward a potential landmark location, and produces a vote. After

many such votes, a voting map is formed, where the votes tend to cluster at landmark locations.

This Hough voting strategy is sensitive to scale and rotation differences between the test image and the exemplars. We therefore produce votes at several different scales (0.7 to 1.3 in increments of 0.1) and in several orientations (-30 to $+30$ degrees roll in increments of 10) on the test face. For efficiency, we use the same extracted features across multiple scales (this is possible because the scale differences between the test face and the exemplar faces are close to 1); only the vote offset vectors are scaled. For votes at different in-plane rotation angles, we use the corresponding set of orientation-specific features computed during the runtime preprocessing step. This is the same approach taken in [20] for face detection and alignment, except they omit rotation search. In our evaluation, we found that the rotation search is critical to our performance, as Figure 4 (a) shows. This is partly due to the fact that our exemplar faces are aligned and thus exhibit very little in-plane rotation variation.

3.5. Step 3: Voting map selection

In selecting voting maps, we enforce the constraint that all voting maps must come from the same in-plane rotation. The intuition is that faces tend to rotate globally, not locally. On the other hand, the scale of face regions can vary locally due to expression or pose. We therefore select scale separately for each landmark.

3.6. Step 4: Final landmark estimation

There are many approaches in the literature for enforcing shape constraints (*e.g.*, [6, 7, 19, 26] to name just a few). We use an exemplar-based approach to shape regularization [2], which fits nicely within our exemplar-based framework.

Belhumeur *et al.* [2] use SVM-based landmark detectors to establish an initial set of landmark location hypotheses, which forms the input to their final shape optimization algorithm. The SVM-based detectors are limited in that each appearance feature provides information only about its own location. In contrast, by aggregating votes from many features, our method takes advantage of the appearance context around each landmark, which provides more robustness to local noise, occlusions, *etc*. We therefore use our landmark voting maps in place of the local detector response maps used in [2].

Additionally, rather than using the entire set of exemplar face shapes as input, which is the approach taken in [2], we use only the top k exemplars retrieved in Step 1 of our pipeline. The top k exemplar shapes tend to be better tailored to the test face than the general set of exemplar shapes, which further aids the optimization.

3.7. Computing exemplar feature weights

In this section we describe our approach for computing a unique weight for each {feature, landmark} pair in each exemplar images. Each weight is a score on the appearance and geometric consistency of each {feature, landmark} pair relative to similar pairs in other images. Intuitively, if a {feature, landmark} pair is consistent with similar pairs in other images (*i.e.*, the features map to the same visual word and the feature-to-landmark offsets are similar), then the feature is a good predictor of the true landmark location, and its vote should have higher weight. Conversely, if, for example, a feature is corrupted due to occlusion, or if it describes an ambiguous local region, then its vote should have lower weight.

For simplicity, our discussion below focuses on a single generic landmark, but the same procedure applies to all landmarks. We first define some new notation. Let $L(f_i^r)$ be the location of feature i in exemplar r, and let $L(l^r)$ be the location of landmark l in exemplar r. The offset vector from $L(f_i^r)$ to $L(l^r)$ is denoted $\Delta L(f_i^r, l^r) = L(l^r) - L(f_i^r)$. Let $w(f)$ denote the mapping from feature f to its visual word, *i.e.* $w(f_i^r) = w(f_j^{r'})$ means that feature i in exemplar r matches feature j in another image r'.

Our goal is to compute the probability that f_i^r will vote for the correct landmark location $L(l^{r'})$ in other face images. Here, we assume that if two features f_i^r and $f_j^{r'}$ in two exemplar images r and r' correspond to the same visual word, *i.e.*, $w(f_i^r) = w(f_j^{r'})$, the relative location offsets from the feature to the landmark should be approximately the same. For "other similar face images" we use the top k retrieved exemplar images from Step 1 of our approach.

We estimate the above probability by counting the number of correct votes (*i.e.*, counting the number of matched offset vectors that are approximately the same) and dividing by the total number of votes. This can be written as

$$P(\Delta L(f_i^r, l^r)) = \frac{1}{N} \sum_{r' \neq r} \sum_{\substack{j \in r' \text{ s. t.} \\ w(f_j^{r'}) = w(f_i^r)}} \Psi\left(\left\| \Delta L(f_i^r, l^r) - \Delta L(f_j^{r'}, l^{r'}) \right\| \right), \quad (1)$$

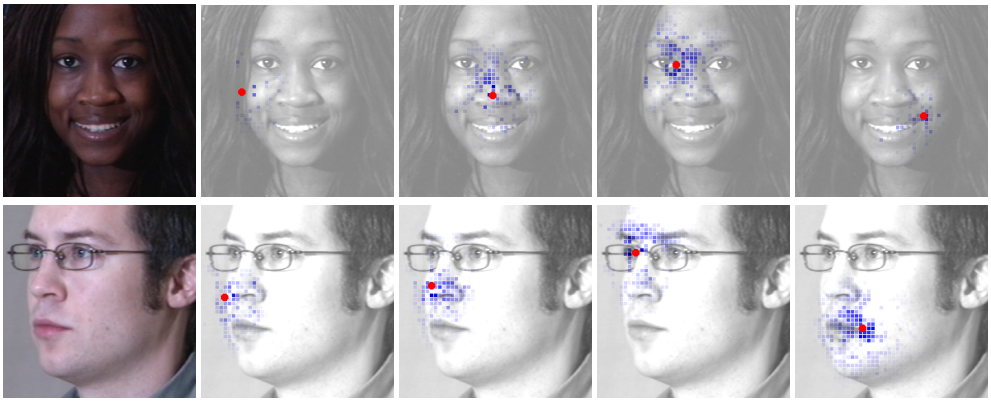
where the summation is over all features in the other exemplar images that share the same visual word, and N is the total number of votes cast by f_i^r. The function $\Psi(\cdot)$ in Eq. (1) quantifies the notion of "approximately the same offset." In our implementation we use $\Psi(x) = \exp\left\{-\frac{x^2}{2\sigma_g^2}\right\}$. We observe in Eq. (1) that

$$||\Delta L(f_i^r, l^r) - \Delta L(f_j^{r'}, l^{r'})|| = ||L(l^r) - V||, \quad (2)$$

where $V = L(f_j^r) + \Delta L(f_j^{r'}, l^{r'})$. This implies that we can evaluate Eq. (1) by first generating a single voting map for f_j^r, where each vote is cast at location V in exemplar r by features in other exemplars; we then count the number of votes near $L(l^r)$ and divide by N to compute $P(\Delta L(f_i^r, l^r))$.

As a kind of regularization, we modulate $P(\Delta L(f_i^r, l^r))$ by a spatial weight that gradually decreases with distance from the landmark,

$$s_j^r = \exp\left\{ -\frac{||L(l^r) - L(f_j^r)||^2}{2\sigma_s^2} \right\}. \quad (3)$$



Exemplar image | Landmark 3 | Landmark 31 | Landmark 40 | Landmark 55

Figure 2. Visualization of landmark-specific feature weights on two different exemplar faces. The red dot in each image shows the ground truth landmark location. The intensity of blue is proportional to the weight of the underlying feature. We see that higher weights naturally correspond to more locally discriminative regions and fully visible landmarks, while lower weights occur in uniform regions. For example, for Landmark 31, we see that features near the edges of the nose have higher weight than features on more uniform areas like the cheeks or philtrum. For Landmark 3, which is occluded by hair in the top row and partially occluded due to head pose in the bottom row, the feature weights are generally lower. We do not use a parametric form to model the weight; the weight values are completely derived from the data.

Regularization is important because N in Eq. (1) can be small, especially for features far from landmarks. Thus, the final weight is

$$v_j^r = s_j^r \cdot P(\Delta L(f_j^r, l^r)). \quad (4)$$

Figure 2 shows four sets of weights for two exemplar images. Several qualities emerge naturally from the data. For example, features in uniform regions, such as the cheeks and forehead, receive smaller weights, while features in less ambiguous regions receive larger weights. We see that nose and chin landmarks require wider spatial support, while very distinctive landmarks such as the eye and mouth corners require only nearby support. Because the weights are specific to each feature, landmark, and exemplar, they can adapt to whatever global or local conditions exist, including different poses, illuminations, facial expressions, occlusions, *etc*.

3.8. Implementation Details

For a large database of exemplar faces, storing weights v_j^r for all j and r across all types of landmarks could be memory prohibitive. Fortunately, few features have any nonzero weights (23.7% in our implementation), which means we can significantly reduce the size of the database by completely removing them. Additionally, few weights from remaining features are significantly larger than zero (20.5% in our implementation), which means they can be efficiently stored in sparse arrays. Thus, even with 68 landmarks, all of our weights fit into a 1.1GB file after 8-bit quantization.

We use $k = 200$ top retrieved exemplars for landmark localization and for training weights. We empirically set $\sigma_s = 0.1 \cdot \text{size}_r$ in Eq. (3), where size_r is the size of the r-th exemplar face, defined as the average height and width of the tightest bounding box that encloses all 68 ground truth landmarks. Intuitively, this setting ensures that a feature's influence will be effectively limited to landmarks on only the most nearby face part(s) (*e.g.*, a left eye feature will have negligible influence on right eye landmark estimates).

We empirically set $\sigma_g = 0.03 \cdot \text{size}_r$ in the Gaussian Ψ in Eq. (1). In practice, σ_g controls the degree to which the voting maps are smoothed. $\sigma_g = 0.03 \cdot \text{size}_r$ effectively smoothes together votes that are within a few pixels of one another.

For face detection, we used our implementation of Shen *et al.*'s face detector [20]. We used their exemplar database to train their algorithm; most of their exemplars come from the AFLW dataset [11]. Please see [20] for more details. For landmark localization, all of our exemplar images come exclusively from the Multi-PIE Face Database [9]; please see Section 3.2 for details.

4. Results and Discussion

In this section we evaluate the accuracy of our approach and compare with several recent works [1, 2, 22, 26, 27, 29]. We show that our approach produces more accurate landmark estimates on especially challenging faces.

4.1. Experimental Datasets

We have evaluated our method on two publicly available datasets: AFW [29] and IBUG [18]. We chose these two datasets because they each contain a large portion of faces with challenging head pose and/or facial expression. In contrast, other popular datasets such as LFPW [2], LFW [10], and Helen [13] contain predominantly frontal, and otherwise less widely varying test cases, which are consequently well-addressed by current, less robust methods. For example, the average landmark localization accuracy in [2] was shown to be slightly better than human labelers on LFPW.

For our quantitative results, we compared our landmark

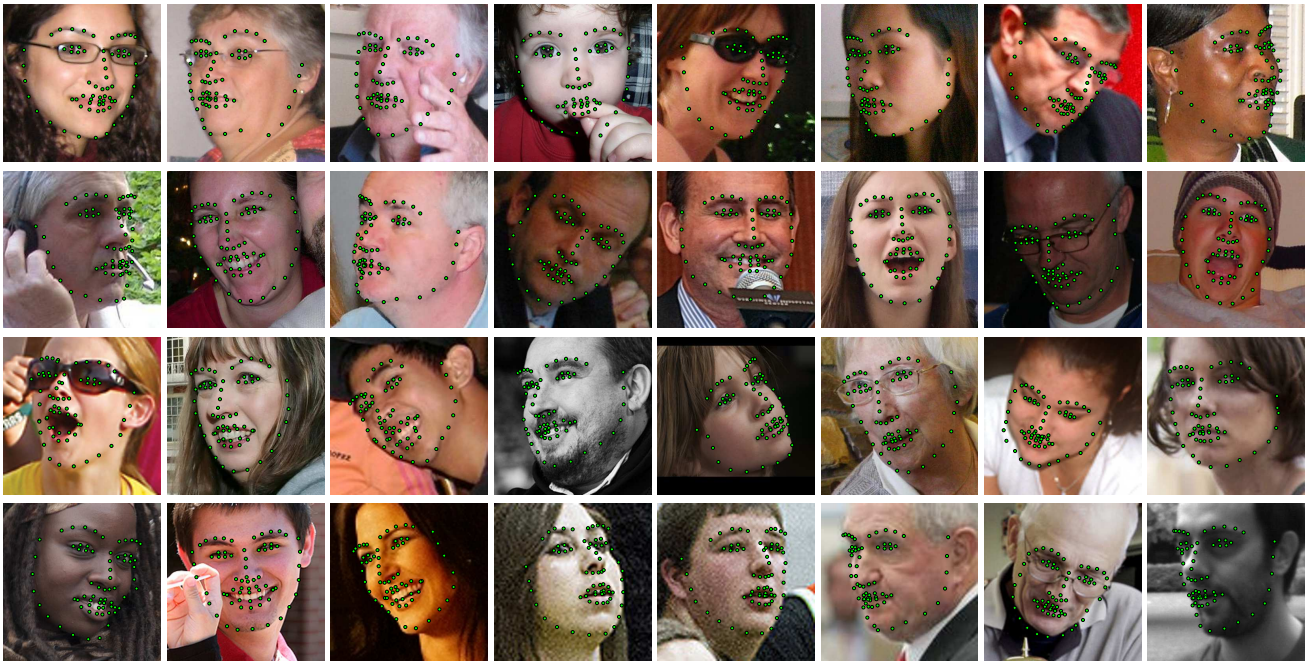


Figure 3. Selected qualitative results on two challenging evaluation datasets: AFW [29] (top two rows) and IBUG [18] (bottow two rows). Our method can handle a wide variety of very challenging conditions, including significant image noise and blur, occlusions, and extreme expressions and head poses. Please see our supplementary material for additional results. **Best viewed electronically in color.**

estimates with the ground truth annotations provided as part of the 300 Faces In-the-Wild Challenge (300-W) [18]. Specifically, 300-W provides 68 landmarks per face according to the Multi-PIE arrangement [9] for 337 faces in AFW and 135 faces in IBUG. Typical AFW and IBUG faces are shown in Figure 3 with landmarks estimated by our algorithm overlaid in green.

4.2. Comparisons with Recent Works

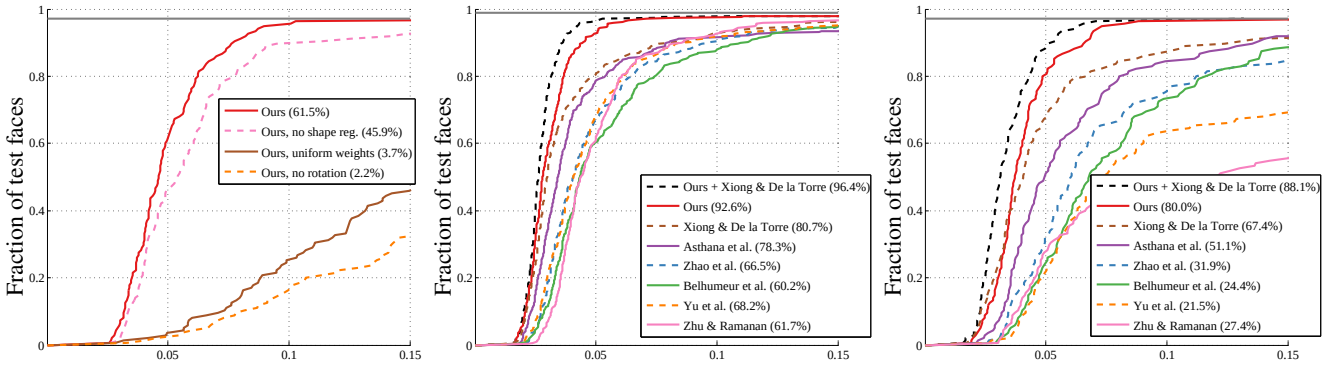
We present cumulative error distribution (CED) curves in Figure 4 to quantitatively compare the accuracy of our method with six other state-of-the-art methods. For fair comparison in Figure 4 (b) and (c), we evaluated only a subset of landmarks (49 out of 68: the eye, nose, and mouth landmarks, with inside mouth corners omitted) common to all method shown (*i.e.*, Xiong and De la Torre's [22] publicly available executable outputs only 49 landmarks). Unless otherwise noted, we evaluated each algorithm using the authors' original implementation.

Comparisons with Zhu and Ramanan [29] In their original evaluation, Zhu and Ramanan assigned an infinite localization error to the entire face if their algorithm incorrectly estimated the landmark arrangement (*i.e.*, if a frontal face was incorrectly labeled as a profile face with only 39 visible landmarks). In our evaluation, if an incorrect arrangement is given by their algorithm, we simply ignore the missing landmarks, and measure the mean error among the given landmarks.

Zhu and Ramanan's landmark localization algorithm is tied to their detection algorithm, and so we do not provide it with any kind of initialization. For each ground truth face annotation, we select the output face that has the largest bounding box overlap (the area of intersection divided by the area of union), and we ignore all false positives. We set their detection threshold to $-\infty$ to avoid missing faces. Zhu and Ramanan provide three off-the-shelf models with their implementation, all trained on Multi-PIE. Although it requires the most computation time, we used their *Independent-1050* model for our experiments since it generally performed best.

Comparisons with Yu *et al.* [26] Yu *et al.* rely on a simplified version of Zhu and Ramanan's algorithm for initialization, and so we do not provide a separate initialization. However, the authors' implementation only returns landmark estimates for the highest scoring face in each image, which is a problem for test images with multiple faces. To obtain results for all annotated faces, we isolated each face from the rest of the image. Specifically, we cropped each annotated face using a box centered on the true face location with it's height and width set to be approximately twice the face height and width.

Comparisons with Belhumeur *et al.* [2] We used our own implementation of Belhumeur *et al.*'s algorithm, which we trained on our Multi-PIE exemplar dataset. We endeavored to reproduce their algorithm as faithfully as possible, although some subtle differences are inevitable. As suggested in [2], we placed a mean face shape over each face detection to initialize the location of each landmark detector win-



(a) IBUG, 68 landmarks (b) AFW, 49 landmarks (c) IBUG, 49 landmarks

Figure 4. Three sets of cumulative error distribution (CED) curves. In all cases, the average localization error is normalized by the face size, defined as the average height and width of the tightest bounding box that encloses all 68 ground truth landmarks [29]. (a) shows the impact of different parts of our pipeline. For fair comparison with other works in (b) and (c), we evaluate on a subset of 49 landmarks (eye, nose, and mouth contour points, minus the inner mouth corners) common to all methods above (*i.e.*, Xiong and De la Torre output 49 landmarks). We see that our method performs significantly better than Zhu and Ramanan [29] and Yu *et al.* [26] despite the fact that these methods are designed to handle faces with non-frontal pose. Our method performs significantly better than Belhumeur *et al.* [2] and Xiong and De la Torre [22], although most of the errors in [22] are due to the initialization. When we initialize [22] using our landmark estimates, the accuracy is better than either approach in isolation, which suggests that our method is complementary to methods like [22].

dow. Belhumeur *et al.* set the size of each landmark detector window to approximately 33% of the height/width of the tight face bounding box in their original work. Unfortunately, we found that this size did not always cover the true landmark locations in our experimental datasets, especially for non-frontal faces. As a compromise, we initialized the height and width of each detector window to the larger of: (1) 33% of the size of the tight face bounding box, or (2) large enough to cover the true landmark location plus 5% of the face size. To evaluate [2] on more challenging datasets (*i.e.*, AFW and IBUG), we had to use a more robust face detector, *e.g.* [20], for initialization. The Viola-Jones [21] detector used in [2] works well for near-frontal faces typical in datasets like LFPW and LFW, but misses many faces in AFW and IBUG (*e.g.*, Viola-Jones missed 62 out of 135 faces in IBUG, whereas [20] missed only 4).

Comparisons with Xiong *et al.* [22] Because the training code for [22] is not publicly available, we used their off-the-shelf model. According to [22], their model is trained on Multi-PIE and LFW. Therefore, comparing their model to ours is reasonable. By default, Xiong and De la Torre's executable uses the Viola-Jones face detector for initialization. Because Viola-Jones misses so many faces in AFW and IBUG, we instead initialized [22] using the same procedure described above for Belhumeur *et al.*'s algorithm. We used this initialization because it is more realistic than simply using the ground truth face bounding box.

It is possible that the additional faces detected by [20] negatively impacted the performance of [22]. We observe that most of the localization errors from [22] arise when the initialization is far from the true landmark locations (*e.g.*, on faces with extreme head pose and/or expression), which suggests that [22] is sensitive to initialization. In such cases, [22] fails to converge to the correct solution. However, in cases where the initialization is relatively close to the true landmark locations, their algorithm performs slightly better than ours. We demonstrate this in Figure 4 (b) and (c), where we also show that the accuracy of their algorithm initialized using our estimates (labeled "Ours + Xiong & De la Torre" in each plot), is higher than either approach in isolation. In this way our approach is complementary to [22].

Comparisons with Asthana *et al.* [1] Asthana *et al.*'s implementation provides three modes for initialization: (1) the localization results from [29], (2) MATLAB's Viola-Jones face detector, or (3) a face bounding box. For fair comparison, we elected to use (3) with each bounding box computed by [20] (*i.e.*, the same face detector used in our pipeline).

Comparisons with Zhou *et al.* [27] Zhou *et al.*'s implementation relies on eye center locations (*e.g.*, provided by an eye detector) to initialize the face shape. We note that, like other AAM-based approaches, their algorithm is sensitive to initialization. Therefore, we provided their algorithm with ground truth eye centers. Rather than using a single AAM model, Zhou *et al.* compute a separate AAM for each test image at runtime. For training, we provided their algorithm with our Multi-PIE exemplar dataset.

4.3. Runtime

On a 900×600-pixel image with one face, the overall runtime of our MATLAB implementation of Shen *et al.*'s face detector [20] is 42.8 seconds on an Intel Xeon E5-2670 workstation; our landmark localization algorithm, also

implemented in MATLAB, requires an additional 25.5 seconds. This is similar to the runtime of [29] (using their *Independent-1050* model) and our MATLAB implementation of [2], but is much slower than several recent methods (*e.g.*, [1, 22, 26]) that are designed to run in real time. However, we note that many strategies exist to speed up our implementation. For example, although the stages of our pipeline must run sequentially, each stage represents an embarrassingly parallel workload (in the parlance of parallel computing), and the size of each landmark voting map, which currently span the entire face, could be reduced significantly by emplying a multi-resolution image pyramid.

5. Conclusions and Future Work

In this paper, we have proposed a novel landmark localization pipeline that combines a feature voting approach to landmark detection with nonparametric shape regularization. Figure 4 (a) demonstrates that shape regularization noticeably boosts our accuracy. Our feature weights are key to the success of our method. We see in Figure 4 (a) that performance drops dramatically without these weights. We also see in Figure 4 (a) that our voting-based approach is sensitive to rotation misalignment between the test face and the exemplar faces, thus necessitating a rotation search. We showed that, despite using exemplars from Multi-PIE, which include limited pitch head rotation, and a limited variety of facial expressions, we can effectively locate landmarks in very challenging images, as shown qualitatively in Figure 3 and quantitatively in Figure 4.

In the future, we will investigate how to combine multiple datasets together as exemplars. To achieve this in practice we must address the problem that each dataset has its own definition of landmarks. In a way, each dataset serves as partially labeled training data relative to one another. An intereting research problem to pursue is how to exploit partially labeled datasets for exemplar based landmark localization.

6. Acknowledgements

This work is supported by NSF IIS-0845916, NSF IIS-0916441, a Sloan Research Fellowship, a Packard Fellowship for Science and Engineering, and Adobe Systems Inc.

References

[1] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. Robust discriminative response map fitting with constrained local models. In *CVPR*, 2013.

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

[3] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. In *CVPR*, 2012.

[4] A. Caunce, D. Cristinacce, C. Taylor, and T. Cootes. Locating facial features and pose estimation using a 3d shape model. In *ISVC*, LNCS 5875, pages 750–761. 2009.

[5] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In *ECCV*, 1998.

[6] T. F. Cootes and C. J. Taylor. Active shape models – 'smart snakes'. In *British Machine Vision Conference*, 1992.

[7] D. Cristinacce and T. Cootes. Automatic feature localisation with constrained local models. *Pattern Recognition*, 41(10):3054–3067, Oct. 2008.

[8] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 32(9):1627–1645, 2010.

[9] R. Gross, I. Matthews, J. F. Cohn, T. Kanade, and S. Baker. Multi-PIE. In *IEEE International Conference on Automatic Face and Gesture Recognition*, 2008.

[10] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[11] M. Koestinger, 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 *IEEE Intl. Workshop on Benchmarking Facial Image Analysis Tech.*, 2011.

[12] P. Kontschieder, S. Rota Bulò, M. Donoser, M. Pelillo, and H. Bischof. Evolutionary hough games for coherent object detection. *CVIU*, 116(11):1149–1158, Nov. 2012.

[13] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang. Interactive facial feature localization. In *ECCV*, 2012.

[14] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *ECCV Workshop on Statistical Learning in Comp. Vis.*, 2004.

[15] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, Nov. 2004.

[16] M. Muja and D. G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *VISAPP Intl. Conf. on Computer Vision Theory and Applications*, 2009.

[17] N. Razavi, J. Gall, P. Kohli, and L. van Gool. Latent Hough transform for object detection. In *ECCV*, 2012.

[18] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 Faces in-the-Wild Challenge: The first facial landmark localization challenge. In *ICCV, 300 Faces in-the-Wild Challenge Workshop (300-W)*, 2013.

[19] J. M. Saragih, S. Lucey, and J. F. Cohn. Face alignment through subspace constrained mean-shifts. In *ICCV*, 2009.

[20] X. Shen, Z. Lin, J. Brandt, and Y. Wu. Detecting and aligning faces by image retrieval. In *CVPR*, 2013.

[21] P. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, 2004.

[22] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *CVPR*, 2013.

[23] H. Yang and I. Patras. Sieving regression forest votes for facial feature detection in the wild. In *ICCV*, 2013.

[24] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011.

[25] K. M. Yi, H. Jeong, B. Heo, H. J. Chang, and J. Y. Choi. Initialization-insensitive visual tracking through voting with salient local features. In *ICCV*, 2013.

[26] X. Yu, J. Huang, S. Zhang, W. Yan, and D. Metaxas. Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model. In *ICCV*, 2013.

[27] X. Zhao, S. Shan, X. Chai, and X. Chen. Locality-constrained active appearance model. In *ACCV*, 2012.

[28] F. Zhou, J. Brandt, and Z. Lin. Exemplar-based graph matching for robust facial landmark localization. In *ICCV*, 2013.

[29] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *CVPR*, 2012.