

# Face Alignment Refinement

Andy Zeng    Vishnu Naresh Boddeti    Kris M. Kitani    Takeo Kanade  
Robotics Institute, Carnegie Mellon University

## Abstract

Achieving sub-pixel accuracy with face alignment algorithms is a difficult task given the diversity of appearance in real world facial profiles. To capture variations such as perspective, illumination or occlusion, with adequate precision, current face alignment approaches rely on detecting facial landmarks and iteratively adjusting deformable models that encode prior knowledge of facial structure. However, these methods involve optimization in latent subspaces, where user-specific face shape information is easily lost after dimensionality reduction. Attempting to retain this information to capture this wide range of variation requires a large training distribution, which is difficult to obtain without high computational complexity. Subsequently, many face alignment methods lack the pixel-level accuracy necessary to satisfy the aesthetic requirements of tasks such as face de-identification, face swapping, and face modeling. In many applications, the primary source of aesthetic inadequacy is a misaligned jawline or facial contour. In this work, we explore the idea of an image-based refinement method to fix the landmark points of a misaligned facial contour. We propose an efficient two stage process - an intuitively constructed edge detection based algorithm to actively adjust facial contour landmark points, and a data-driven validation system to filter out erroneous adjustments. Experimental results show that state-of-the-art face alignment combined with our proposed post-processing method yields improved overall performance over multiple face image datasets.

## 1. Introduction

Given an estimated facial contour returned from face alignment, our objective is to refine the contour such that it is closer to the true facial boundary. Accurately detecting facial boundaries is a challenging problem because the contours of facial profiles in the real world are subject to a broad range of variations in illumination, occlusion, noise, and individual differences. A facial contour in an image may be partially occluded by hair, faded into the wrinkles, or hidden by shadows. The challenge presented by these

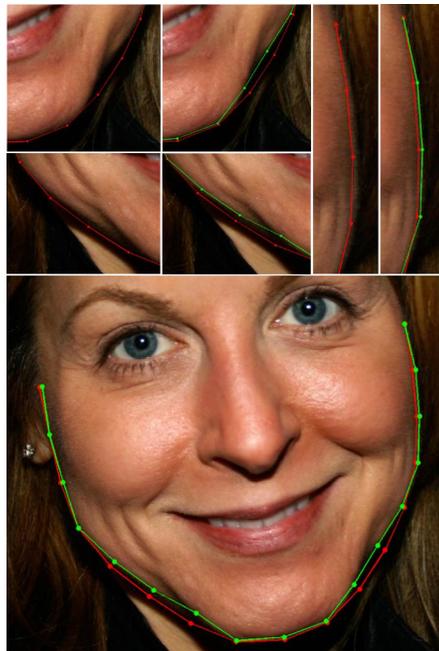


Figure 1: Alignment refinement result. Initial facial contour (red) and refinement result (green).

problems is further compounded by having to consider the variations in jawline structure and individual facial features that may cause irregularities in the facial outline. Handling such variation at a high level of detail is the key to designing a robust face alignment contour refinement algorithm.

With the explosive increase in personal photos across the web nowadays, the popularity of face alignment in modern applications is rapidly growing. For many of these applications, e.g., face de-identification, face swapping, and face modeling, the aesthetic quality of the aligned facial boundary is quite sensitive to slight misalignment. In the case of face swapping, an estimated facial contour extending past the true facial boundary will introduce background artifacts onto the output face. The main motivation behind the work is an application-side demand for more accurate face alignment results.

Despite the abundance of research on face alignment, many state-of-the-art methods are not able to align an esti-

mated facial shape to the true facial boundary with sub-pixel accuracy. Face alignment typically involves an optimization problem where the goal is to match some deformable face model to the true face shape as closely as possible using detected facial features as anchor points. During this optimization, the face shape model is often parametrized or constrained resulting in the loss of the fine detailed information about the facial contour. In this paper, we propose a flexible facial contour refinement method to correct the facial contour inaccuracies of generic face alignment methods using a data-driven post-processing technique.

Based on observations over various state-of-the-art face alignment results [4, 12, 14, 1], we propose a two-step approach for fixing facial contour misalignment. In the first step, we introduce the active adjustment algorithm responsible for shifting individual landmark points that constitute the facial contour. The shifting is performed heuristically based on edge response, the distance from the initial contour estimate returned from alignment, and edge direction. In the second step, we introduce a data-driven validation process that reinforces the overall performance of the active adjustment algorithm by training a classifier to deter the refinement process from making potentially erroneous adjustments.

## 2. Related Works

Face alignment is a very challenging and well-studied problem. Active Shape Models [6] and Active Appearance Models [5] are the most well known and widely used models for shape-fitting. Constrained Local Models [1, 16, 7] are another class of approaches for face alignment that are largely focused on global spatial models built on top of local landmark detectors. Recently many discriminative shape-based regression approaches [4, 15] have been proposed in the literature. Instead of relying on parametrized appearance and shape models, these approaches leverage large amounts of training data to learn a cascade of regressors, mapping image features to the final facial shape.

The task of refining the contour of a face shape is similar to the problem of contour fitting. Contour fitting generally requires some form of boundary detection, followed by an optimization step, where the fitting of a deformable contour model over the boundaries of interest is performed. Some methods iteratively re-sample adaptive spline models [9, 13] while other methods apply dynamic programming to energy-minimizing deformable contours [2]. The task of facial contour refinement however, differs from the task of contour fitting in that facial contour refinement is given a close initial alignment. Assuming that the results from face alignment return a reasonable estimate of the facial contour, refinement needs to actively work with this information in order to accurately adjust the contour under a wide range of image variation. Furthermore, because the results from

boundary detection can sometimes be noisy or misleading, refinement also needs to be conservative in order to minimize the number of erroneous adjustments. In this work, our goal is to construct a refinement algorithm that maximally improves the accuracy of an estimated facial contour only for those images that are problematic.

## 3. Problem

In a two-dimensional digital face image  $I$ , a face shape  $S = \{p_i \in \mathbb{R}^2\}_{i=1}^N$  consists of  $N$  facial landmark points  $p_i = (x_i, y_i)$ . The goal of face alignment is to estimate a shape  $S$  as close as possible to the true shape  $\hat{S}$ , *e.g.* to minimize

$$\|S - \hat{S}\|_2 \quad (1)$$

Among the  $N$  points that constitute a face shape  $S$ , there are  $M < N$  points that make up the facial contour  $C = \{p_i \in S\}_{i=1}^M$ . Given  $S$ , our objective is to fine-tune alignment contour  $C$  to be closer to the true contour  $\hat{C} = \{\hat{p}_i \in \hat{S}\}_{i=1}^M$  after refinement, *e.g.* to maximize

$$\text{Error}(C_{\text{before}}, \hat{C}, m_p) - \text{Error}(C_{\text{after}}, \hat{C}, m_p) \quad (2)$$

where  $m_p$  is the performance metric,  $C_{\text{before}}$  and  $C_{\text{after}}$  are the alignment contours before and after refinement respectively. Equation 2 semantically represents contour improvement, and will be used to guide training and evaluate the performance of our post-processing approach. As part of our objective, we want this value to be as consistently positive as possible.

## 4. Facial Contour Refinement

In this section, we first introduce our active observation-based adjustment process. Conceptually, the algorithm individually adjusts each landmark point  $p_i$  of a given alignment facial contour  $C \subset S$  by shifting it to the nearest, strongest edge that is closely parallel to the facial outline originally generated by alignment contour  $C$ . But since this method is constructed on the basis of human intuition, it remains incapable of performing robustly under the wide range of misalignment variations in illumination, noise, occlusions, etc. Hence, if used without proper discretion, this algorithm is susceptible to performing "bad" adjustments. Therefore, we present a compatible data-driven validation framework, in which we conditionally perform the active adjustments based on prior post-refinement observations. Given that each facial contour consists of  $M$  landmark points, we train  $M$  distinct SVMs over a large collection of training faces in order to be able to determine, per contour at test time, which alignment landmark points should undergo active adjustments and which points should be left alone in order to maximize the overall contour improvement. As we shall see later, this form of preemptive

filtering is necessary in our refinement approach in order to conceivably maintain positive contour improvement as consistently as possible.

#### 4.1. Active Refinement

Our active adjustment algorithm is based on three major observations. The first observation is that a true facial boundary is more likely to be located on an edge than anywhere else. Given a landmark point  $p_i$  in facial contour  $C$ , a naïve adjustment algorithm reflecting this observation would shift  $p_i$  to the point with the strongest edge response within some small search radius  $r$ , *e.g.*

$$P = \{s \in \mathbb{R}^2 : \|p_i - s\|_2 < r\}$$

$$f_h(p) = \text{Edge}(p)$$

$$\text{Refine}(p_i) = \{p \in P \text{ s.t. } f_h(p) = \max_{s \in P} f_h(s)\} \quad (3)$$

where  $\text{Edge}(x) \in [0, 1]$  returns the edge response for a point  $x$ . Our second observation is that the pre-refined alignment facial contour serves as an adequate estimate for the true facial boundary. To incorporate this into the first observation reflected in Eq. 3, we add a distance factor to the heuristic function  $f_h(p)$ , *e.g.*

$$f_h(p) = w_1 \text{Edge}(p) + w_2 \left(1 - \frac{\|p_i - p\|_2}{r}\right)$$

$$\text{Refine}(p_i) = \{p \in P \text{ s.t. } f_h(p) = \max_{s \in P} f_h(s)\} \quad (4)$$

where  $w_1$  and  $w_2$  are weights. The refinement algorithm in Eq. 4 using the new heuristic function, as it currently stands, may adjust landmark points to edges that do not retain the innate facial structure estimated from alignment. In other words, the variation in edge direction is not properly constrained, *i.e.* erroneously shifting a landmark point around the chin to the edge of a collar directionally perpendicular to the outline generated by the true facial contour. So our final observation, conceptually derived from our second observation, is that the outline generated by pre-refined alignment facial contour should be near parallel to the outline generated by the true facial boundary.

Under all three observations, the ultimate goal of our active refinement algorithm is to move each landmark point to the nearest, strongest edge segment that is near parallel to the outline generated by the alignment facial contour. See Figure 2. More specifically, for each alignment landmark point  $p_i$  in facial contour  $C$ , we generate a series of cascading line segments parallel to the outline generated by  $C$ , each line segment explicitly defined as a collection of points in a single direction *e.g.*

$$v_{i_c} = p_{i+1} - p_{i-1} \quad v_{i_p} = (-v_{i_c y}, v_{i_c x})$$

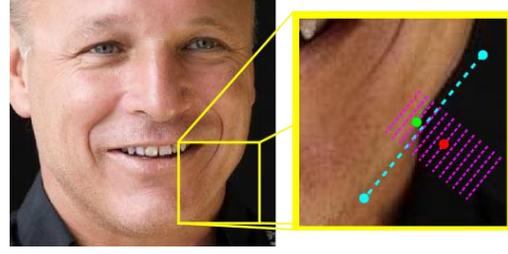


Figure 2: Active adjustment. The landmark point (red) is updated to a new location (green) by searching along a line perpendicular to the tangent line (dashed cyan line).

$$b_i = p_i - \frac{r}{2} \left( \frac{v_{i_c}}{\|v_{i_c}\|} \right) - \frac{r}{2} \left( \frac{v_{i_p}}{\|v_{i_p}\|} \right)$$

$$p_{k_j} = b_i + \frac{r}{3\sigma} k(v_{i_p}) + \frac{r}{3\sigma} j(v_{i_c})$$

$$L = \{L_k\}_{k=1}^{3\sigma} \quad L_k = \{p_{k_j}\}_{j=1}^{\sigma} \quad (5)$$

where  $p_{i+1}$  and  $p_{i-1}$  are neighboring alignment contour points of  $p_i$ , and  $\sigma$  is a saturation value. A score is computed for each line segment in  $L$  based on the heuristic function  $f_h(p)$  in Eq. 4

$$\text{score}(L_k) = \sum_{j=1}^{\sigma} \frac{f_h(p_{k_j})}{\sigma} \quad (6)$$

and  $p_i$  is shifted to the midpoint of the line segment with the highest computed score from Eq. 6, *e.g.*

$$L_{\text{best}} = \{L_k \in L \text{ s.t. } \text{score}(L_k) = \max_{l \in L} \text{score}(l)\}$$

$$L_{\text{best}} = \{p_{\text{best}_j}\}_{j=1}^{\sigma}$$

$$\text{Refine}(p_i) = p_{\text{best}_{\lceil \frac{\sigma}{2} \rceil}} \quad (7)$$

#### 4.2. Data-Driven Validation

An observation-based edge detection approach to refinement is sufficient to fix the easy misalignment cases. However, a large percentage of misaligned landmark points is still difficult to assess and fix, even to the human eye. We address this problem by adopting a data-driven approach to recognize and preemptively avoid such difficult cases. For each of the  $M$  landmark points that make up a facial contour  $C$ , we train a binary SVM to classify each corresponding landmark as an easy or difficult case. In order to minimize the total number of erroneous adjustments, these classifiers are used to limit the refinement algorithm from adjusting the difficult cases.

Contrast normalized pixel values extracted from a small region around each point serve as the features used to train the classifiers. These patches are rotated with respect to the outline generated by the alignment facial contour such that

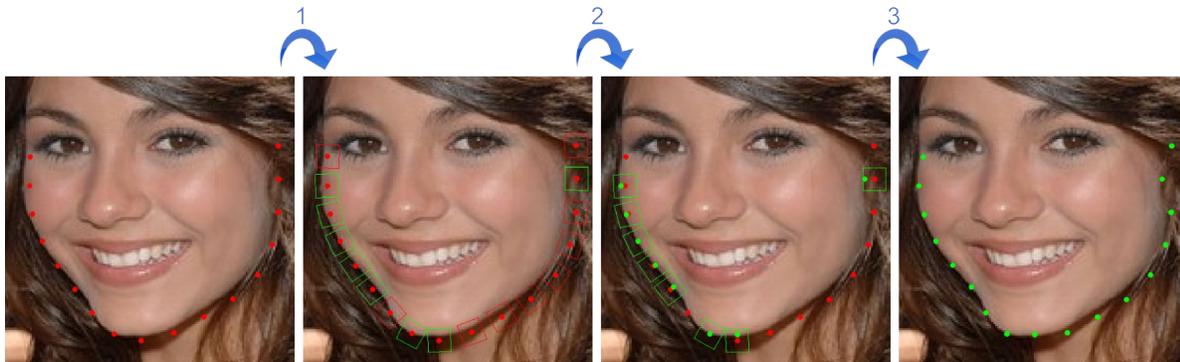


Figure 3: Refinement pipeline: (1) select landmark patches to be updated (green boxes), (2) adjust landmark positions, (3) return updated landmarks.

the right side of the patch is further away from the face than the left side. At test time, prior to running the active adjustment algorithm for each landmark point, we generate a feature vector reflecting the small rotated square region around the point. If the SVM classifies the region as likely to facilitate negative improvement after refinement, then we skip active adjustment for that particular landmark point. This is done before the adjustment of each landmark point of every test facial contour. Since our objective for face alignment post-processing is to maintain positive improvement as consistently as possible, this validation framework essentially reinforces the dependability of the active adjustment algorithm by learning and avoiding potential erroneous adjustments. Figure 3 illustrates the role of the validation framework within the refinement approach.

## 5. Experiments

In this section, we provide some experimental analysis which highlights the advantages of our proposed facial contour refinement approach. The experiments are designed to demonstrate the validity of our active adjustment method, illustrate the intuition behind the validation framework, and evaluate the quantitative and qualitative performance of our refinement approach as a whole.

**Face Alignment** For our experiments, we use Face Alignment Robust To Occlusion [1], which approximates face shape  $S$  and returns  $N$  binary labels corresponding to the estimated state of occlusion for each individual landmark point in  $S$ . Since our approach was not designed to be robust for occluded landmark points, during refinement we limit our adjustments to the non-occluded points in order to minimize the number of misalignment cases attributed to occlusions.

**Datasets** We demonstrate the efficacy of our contour refinement approach, by evaluating it on three different face datasets namely, HELEN [11], “Labeled Face Parts in the Wild” (LFPW) [3] and “Annotated Faces in the Wild”

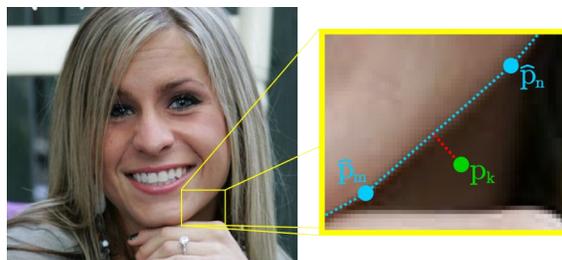


Figure 4: A conceptual visualization of the error metric introduced in Equation 9. The accuracy of some point  $p_k \in C$  is measured by its distance to the facial outline generated by the true contour  $\hat{C}$  (blue).

(AFW) [16]. For consistent cross-database annotations, we used the generated annotations provided by IBUG [10]. Each dataset presents a different challenge due to varying degrees of image quality and facial variation. The HELEN dataset contains 2,000 training and 300 testing high resolution images obtained from Flickr. The LFPW face image dataset features 1,132 training and 300 testing images pulled from the Internet using simple search queries. The facial images in both of these datasets feature exhibit a wide range of appearance variations including pose, lighting, facial expressions, occlusion, and individual differences. The AFW dataset consists of 250 images with 468 faces out of which we isolate 100 images for testing, and use the rest for training. Most of the faces from this dataset have poor image quality and/or low resolution and consist of faces captured under unconstrained conditions.

**Edge Detection** In our experiments, we use the fast edge detection method proposed by Dollár and Zitnick [8]. Capable of multi-scale edge detection, this edge detector features superior run-time complexity while maintaining state-of-the-art edge detection performance.

**Error Metric** If we derive from the metric in Equation 1,

then the facial contour error is computed as

$$\text{Error}(C, \hat{C}) = \|C - \hat{C}\|_2 \quad (8)$$

The relative position of each estimated landmark point  $p_i$  on the alignment facial contour  $C$  is likely to differ from the relative position of its ground truth counterpart  $\hat{p}_i$  in  $\hat{C}$ . Unfortunately, the error metric in Eq. 8 does not capture this variation. So we adopt a new error metric  $m_p$  that conceptually reflects the perpendicular distance between each landmark point of  $C$  to the outline generated by the true contour  $\hat{C}$  (see Fig. 4), *e.g.*

$$\text{Error}(p_k, m_p) = \frac{|(\hat{p}_n - \hat{p}_m) \times (\hat{p}_k - \hat{p}_m)|}{\|\hat{p}_n - \hat{p}_m\|_2} \quad (9)$$

$$\text{Error}(C, \hat{C}, m_p) = \sum_{i=1}^M \frac{\text{Error}(p_i, m_p)}{\hat{\varphi}} \quad (10)$$

where  $\hat{p}_n, \hat{p}_m \in \mathbb{R}^2$  denote the two landmark points in  $\hat{C}$  closest to  $p_k$  via Euclidean distance and  $\hat{\varphi}$  represents the true inter-pupillary distance. We use the per-point error metric in Equation 9 to guide validation training, and the relative accuracy improvement metric (semantically defined as the % error reduced after refinement)

$$\frac{\text{Error}(C_{\text{before}}, \hat{C}, m_p) - \text{Error}(C_{\text{after}}, \hat{C}, m_p)}{\text{Error}(C_{\text{before}}, \hat{C}, m_p)} \quad (11)$$

and the absolute accuracy improvement metric

$$\text{Error}(C_{\text{before}}, \hat{C}, m_p) - \text{Error}(C_{\text{after}}, \hat{C}, m_p) \quad (12)$$

in conjunction with Eq. 11 to evaluate the performance of our post-processing approach.

### 5.1. Performance of observation-based refinement

In section 4.1, we described the three major observations around which our active adjustment algorithm is structured. Recall that with each major observation, we intuitively modified our adjustment algorithm to reflect that observation. In this experiment, we demonstrate how the integration of each modification works to boost the overall performance of our refinement approach. We train and test the complete refinement approach three times while swapping out the adjustment algorithm each time; once using Eq. 3 (one incorporated observation), once using Eq. 4 (two incorporated observations), and once using Eq. 7 (all three incorporated observations). Since the training of the data-driven validation framework learns from the pre-filtered performance of the active adjustment algorithm over the training image dataset, the adjustments algorithms are changed before each training session to reflect the swapped adjustment equations used respectively during testing. Additionally, it is important to note that even though we used

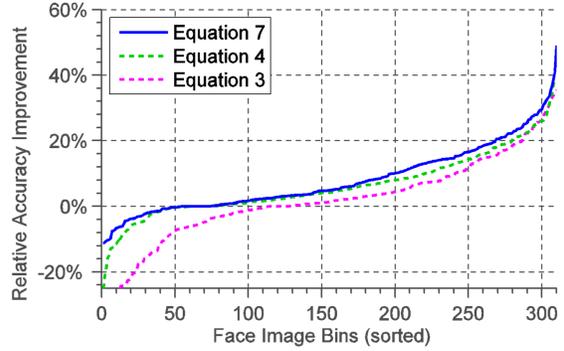


Figure 5: Comparison of different active adjustment algorithms in Equation 3, Equation 4, and Equation 7. Relative accuracy improvement is based on Equation 11.

the Helen dataset to generate the results of this experiment in Fig. 5, the results generated by the LFPW and AFW datasets were consistently similar. For this experiment, as well as the following experiments, we empirically chose the active adjustment parameters to be  $\sigma = 5$ ,  $r = \varphi/4$  (where  $\varphi$  is the detected inter-pupillary distance) for a good trade-off between accuracy and computational cost.

Figure 5 illustrates the relative accuracy improvement (Eq. 11 with Eq. 10) of every test face (sorted by improvement). We see that incorporating the second observation made in section 4.1 to formulate Eq. 4 worked very well in reducing the number of cases where the naive implementation using Eq. 3 would have inaccurately shifted a landmark point to an outlier edge. Reducing the search space effectively reduced the possibility of misalignment. Additionally, we see that the active adjustment algorithm using Eq. 7 was able to further reduce some of the outlier misalignment cases attributed to the edge direction variation. Note that incorporating all three observations made in section 4.1 yields the best overall performance.

### 5.2. Verifying the data-driven validation framework

For this experiment, we verify the effectiveness of the data-driven validation framework used to reinforce the overall performance of the active adjustment algorithm.

**Parameter Settings** In our experiments, the C-SVC SVM was trained with the polynomial kernel  $K(u, v) = (\gamma u^T v + c)^d$  where the parameters were empirically chosen as  $\gamma = 2$ ,  $c = 1$ ,  $d = 3$  for consistent cross-dataset performance. For both training and testing, given detected face shape width  $w$ , the size of the localized square patches around each landmark point were set at  $l \times l$  pixels where  $l = \frac{w}{10}$ .

Figure 6 presents a comparison between the refinement results with and without the data-driven validation framework. We see that although the accuracy improvement of

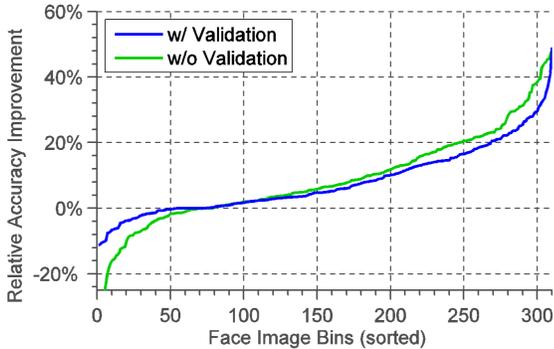


Figure 6: Comparison of refinement with and without the data-driven validation framework described in section 4.2. Relative accuracy improvement is based on Eq. 11.

some of the face shapes were not as high with the validation process as they were without it, nonetheless the total number of erroneous adjustments was significantly reduced with the validation process. The results of the refinement with the validation framework are much more desirable since our objective is to improve alignment accuracy through active adjustment while reducing the total number of erroneous adjustments as much as possible.

### Visualizing the validation framework

Figure 7 illustrates the localized patches around landmark points with the lowest and highest decision values from the SVM, and their weighted averages. From the visualization of the averages, we see that the SVM learns to avoid adjusting landmark points that are already located near a strong gradient (presumed to be the true facial contour). On the other hand, the SVM also learns to favor adjustments to be made for points that are only slightly off from a strong gradient, since the active adjustment algorithm is more likely to successfully improve the accuracy of a smaller case of misalignment.

It is also interesting to note that since the patches are localized such that the right side of the patch is further away from the face than the left side, the average patches over the highest decision values seem to imply that better adjustments are made for points that located away from the face, as opposed to points that lie directly on the face. This makes sense, because an edge detection based adjustment algorithm is much more likely to fail due to wrinkles, facial hair, or other similar edge-like facial features. This is why the validation framework is important and necessary to minimize the possibility of erroneous adjustments.

### 5.3. Quantitative Evaluation

In this experiment, we directly evaluate the absolute contour accuracy improvement of the refinement method over

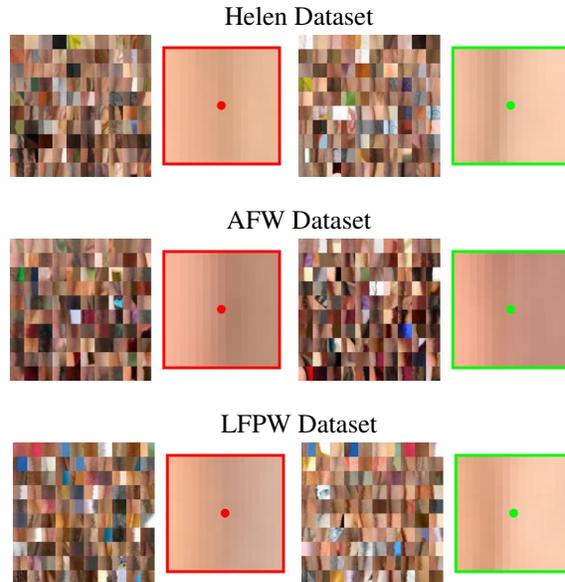


Figure 7: For each dataset, the left mosaic presents the top 100 patches around the landmark points with the lowest decision values from the SVM. The patch highlighted in red is the weighted average over the pixel intensities of all patches in the left mosaic. Similarly, the right mosaic presents the top 100 patches around the landmark points with the highest decision values, and the patch highlighted in green is their weighted average over pixel intensities.

all three datasets. Fig. 8 illustrates the average absolute accuracy improvement for each landmark point for every face in each database after refinement. Refinement does reasonably well in the Helen dataset where image quality and ground truth annotation accuracy are both high. Refinement performs quite consistently with the LFPW dataset. And finally, as expected, refinement did not do so well in the AFW dataset, where ground truth annotations lacked sub-pixel accuracy, and image quality was sometimes very low (featuring some faces with widths  $< 200$  pixels). Relative differences in accuracy improvement between landmark point indices can reflect the structural weaknesses of the deformable facial models being optimized during face alignment. Overall, the refinement process generally does well to improve the accuracy of face alignment [1]. Table 1 summarizes the computed average contour error (Eq. 12) over every face for each dataset before and after refinement.

### 5.4. Qualitative Evaluation

It is important to keep in mind though that the quantitative experiments may not be completely representative of the true performance of the refinement approach. The main

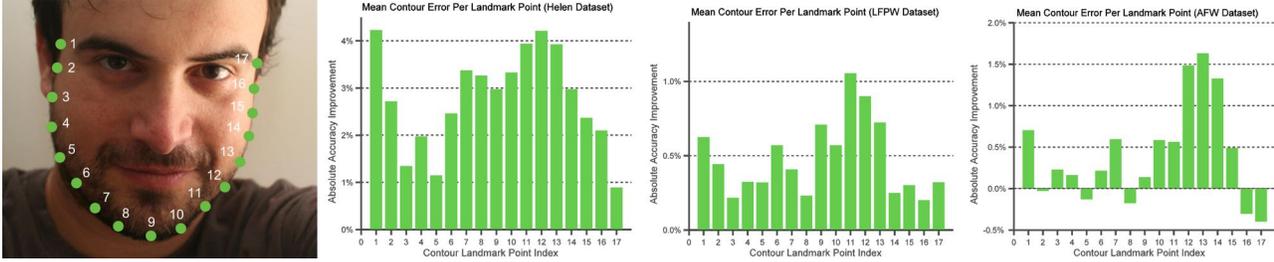


Figure 8: Comprehensive quantitative results over the Helen[11], AFW [16], and LFPW [3] datasets. On the left we show the 17 landmark points from alignment [1] used to describe the facial contour and the graphs display the average absolute accuracy improvement (Equation 12) for each landmark point over all test images in each dataset.

Table 1: Mean Contour Error of [1] (in %)

Dataset	Before Refinement	After Refinement
Helen [11]	3.6639	<b>3.4587</b>
AFW [16]	4.9339	<b>4.8610</b>
LFPW [3]	3.4083	<b>3.2720</b>

motivation behind the construction of this method was, after all, to improve upon the aesthetic quality of state-of-the-art face alignment results. Furthermore, the ground truth annotations provided by the datasets that we used were built for the purpose of evaluating face alignment performance, where sub-pixel accuracy for each and every landmark point is typically not to be expected, especially for very high resolution images. Therefore, our experiments require a qualitative evaluation to give a better picture of overall aesthetic improvement in face alignment results after refinement. For each test face image across all datasets, we generate a copy of the face image where the face alignment contour points before refinement are highlighted in red and the shifted contour points after refinement highlighted in green. See Figure 9. Table 2 summarizes the average results of our questionnaire, where 3 subjects are asked to step through all test face images of each dataset, and judge whether or not the contour improved after refinement. If no contour change was observed, or if there was some difficulty in discerning the state of contour improvement, the subjects were asked to mark ‘uncertain’ on the questionnaire.

Table 2: Contour Improvement: Qualitative Evaluation (%)

Dataset	Yes	No	Uncertain
Helen [11]	90.2516	5.3459	4.4025
AFW [16]	59.6667	6.3333	34.0000
LFPW [3]	95.8333	2.7778	1.3889

We see that for most test face images from the Helen and LFPW datasets, the subjects noticed an improvement in

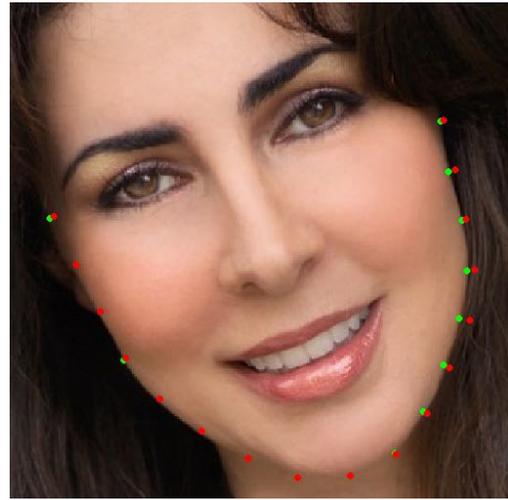


Figure 9: Sample refinement visualization (image from LFPW) shown to subjects during qualitative evaluation.

the accuracy of the facial contour. However, for the AFW dataset, the subjects had some difficulty in judging whether or not there was improvement - this is likely due to the fact that this dataset contains many images with faces that have a low resolution. Overall, the qualitative tests overwhelmingly suggest that our refinement approach facilitates an improvement in the aesthetic quality of face alignment results.

## 6. Discussion

We proposed an observation-based active adjustment algorithm to fix the inaccurate landmark points of a given contour from a face shape returned from face alignment. To reinforce the performance of this algorithm, we introduced a data-driven validation framework to learn the weaknesses of the algorithm and to minimize the number of erroneous adjustments from refinement. Our evaluation demonstrates that our approach is capable of consistently improving the

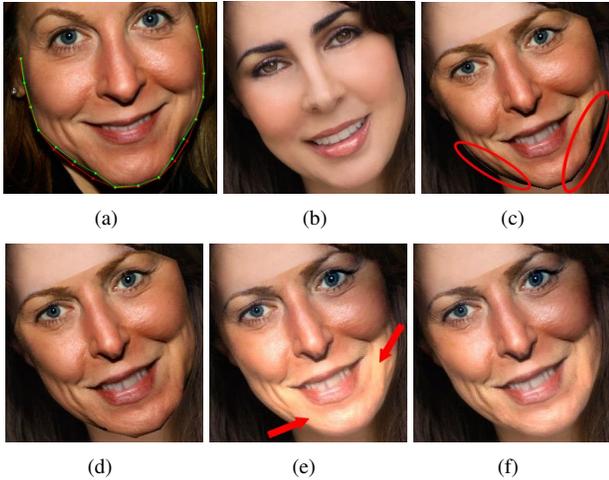


Figure 10: A face swapping example where the sample contour refinement result from Figure 1 (a) blends into background face (b). Raw face swapping using the face alignment without refinement (c) and with refinement (d). Poison blending over the face replacement results using face alignment without refinement (e) and with refinement (f). The dark regions of the background captured by the misaligned facial contour force the bottom half of the face to be discolored. Note the difference in illumination between (e) and (f). Using refinement to reduce facial contour misalignment can effectively reduce the facial boundary noise that affects face swapping results.

sub-pixel accuracy as well as the aesthetic quality of a given facial contour. The active adjustment algorithm can also be applied to other problems like object contour refinement and structure segmentation boundary refinement.

## References

- [1] Anonymous (under submission), details omitted for double-blind reviewing, 2014.
- [2] A. A. Amini, T. E. Weymouth, and R. C. Jain. Using dynamic programming for solving variational problems in vision. *Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence*, TPAMI 1990.
- [3] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR 2011.
- [4] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR 2012.
- [5] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685, 2001.
- [6] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models—their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.
- [7] D. Cristinacce and T. Cootes. Automatic feature localisation with constrained local models. *Pattern Recognition*, 41(10):3054–3067, 2008.
- [8] P. Dollár and C. L. Zitnick. Structured forests for fast edge detection. *Proceedings of the IEEE International Conference on Computer Vision*, ICCV 2011.
- [9] M. A. T. Figueiredo, J. M. N. Leitao, and A. K. Jain. Adaptive b-splines and boundary estimation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR 1997.
- [10] IBUG. <http://ibug.doc.ic.ac.uk/resources/300-W/>. <http://ibug.doc.ic.ac.uk/resources/300-W/>.
- [11] V. Le, J. Brandt, Z. Lin, L. Boudev, and T. S. Huang. Interactive facial feature localization. *Proceedings of the European Conference on Computer Vision*, ECCV 2011.
- [12] L. Liang, R. Xiao, F. Wen, and J. Sun. Face alignment via component-based discriminative search. *Proceedings of the European Conference on Computer Vision*, ECCV 2008.
- [13] D. Rueckert and P. Burger. Contour fitting using an adaptive spline model. *Proceedings of the British Machine Vision Conference*, BMVC 1995.
- [14] G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. Robust and efficient parametric face alignment. *Proceedings of the IEEE International Conference on Computer Vision*, ICCV 2011.
- [15] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *IEEE Conf. Computer Vision and Pattern Recognition*, 2013.
- [16] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *IEEE Conf. Computer Vision and Pattern Recognition*, 2012.