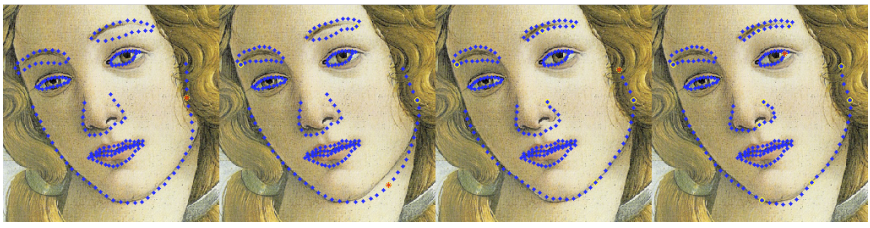
Interactive Facial Feature Localization

Vuong Le[1], Jonathan Brandt[2], Zhe Lin[2],
Lubomir Bourdev[3], and Thomas S. Huang[1]

[1] University of Illinois at Urbana Champaign, Urbana, IL 61801, USA
[2] Adobe Systems Inc., San Jose, CA 95110, USA
[3] Facebook Inc., Menlo Park, CA 94025, USA

Abstract. We address the problem of interactive facial feature localization from a single image. Our goal is to obtain an accurate segmentation of facial features on high-resolution images under a variety of pose, expression, and lighting conditions. Although there has been significant work in facial feature localization, we are addressing a new application area, namely to facilitate intelligent high-quality editing of portraits, that brings requirements not met by existing methods. We propose an improvement to the Active Shape Model that allows for greater independence among the facial components and improves on the appearance fitting step by introducing a Viterbi optimization process that operates along the facial contours. Despite the improvements, we do not expect perfect results in all cases. We therefore introduce an interaction model whereby a user can efficiently guide the algorithm towards a precise solution. We introduce the Helen Facial Feature Dataset consisting of annotated portrait images gathered from Flickr that are more diverse and challenging than currently existing datasets. We present experiments that compare our automatic method to published results, and also a quantitative evaluation of the effectiveness of our interactive method.



1 Introduction

Accurate facial component localization is useful for intelligent editing of pictures of faces, such as opening the eyes, red eye removal, highlighting the lips, or making a smile. In addition, it is an essential component for face recognition, tracking and expression analysis. The problem remains challenging due to pose and viewpoint variation, expression or occlusion (e.g. with sunglasses, hair or a beard). While the techniques have improved a lot over the past five years [1–5],

A. Fitzgibbon et al. (Eds.): ECCV 2012, Part III, LNCS 7574, pp. 679–692, 2012.

we are still far from a reliable and fully-automated facial component localization system. For all practical purposes, an interactive, user-assisted face localization would be necessary.

Our goal is a facial component localization system that achieves high fidelity results with minimal user interaction. Our contributions are as follows:

1. A novel facial component localization algorithm designed to accommodate partial labels. We use a part-based formulation which allows us to edit one part without affecting the correct localization of other face parts. The landmark predictions of each face part are jointly fit by finding the global minimum of the energy using dynamic programming.
2. An extension of the feature localization algorithm to accommodate partial observations. We allow the user to specify landmarks and adjust the remaining ones to minimize the residual error.
3. A new challenging dataset for facial component localization which contains 2330 high-resolution, accurately labeled face images and has larger degree of out-of-plane orientation and occlusion typical of real-world scenarios.

We show that our system in a fully automated mode outperforms other state of the art systems on this challenging dataset. Furthermore we show that the error decreases significantly after only a minimal number of user corrections.

2 Related Work

Active shape model (ASM) [6] and Active appearance model (AAM) [7] form classic families of methods for facial feature point detection. Comparing these two models, ASM has the advantages of being more accurate in point (or contour) localization, less sensitive to lighting variations and more efficient, hence is more suitable for applications requiring accurate contour fitting.

Since the first introduction of the traditional ASM, there have been many extensions for improving its robustness, performance and efficiency. For example, [8] employs mixtures of Gaussians for representing the shape, [9] uses Kernel PCA and SVM, and models nonlinear shape changes by 3D rotation, [10] apply robust least squares algorithms to match the shapes to observations, [11] uses more robust texture descriptors to replace the 1D profile model and used k nearest neighbor search for profile searching, and [12] relies on Bayesian inference.

Among those extensions to classical ASM, the recent work of Milborrow and Nicolls [1] with the introduction of the 2D profile model and denser point set obtained promising results through quantitative evaluation. The recent work [4] combines global and local models based on MRF and an iterative fitting scheme, but this approach is mainly focused on localizing very sparse set of landmark points.

Other notable recent works exploring alternatives to ASM include [2]. In [2], a robust approach for facial feature localization is proposed by a discriminative search approach combining component detectors with learned directional classifiers. In [3], a generative model with shape regularization prior is learned and

Fig. 1. Sample images from the Helen Facial Feature Dataset

used for face alignment to robustly deal with challenging cases such as expression, occlusion and noise. As an alternative to the parametric model approaches, a principled optimization strategy with nonparametric representations for deformable shapes is recently proposed in [13].

A Component based ASM is introduced in [14] which implements a set of ASMs for facial features and stacks up the PCA parameters into one long vector, then fits those vectors in a single global model using a Gaussian Process Latent Variable Model to handle nonlinear distribution. In comparison with our approach, they do not model the global spatial configuration of features of faces therefore may generate invalid configurations. In [2] the face components are found by an AAM style fitting method and a discriminative classification is used for predicting the movement of components. In another effort to explore the configuration of face parts, Bayesian objective function are combined with local detectors in [15] for localization of facial parts.

Motivated by the pictorial structure model in [16], we aim to decompose a face shape into components, and model variations of individual components as well as the spatial configuration between components. In [16], the face/human pose inference problem is modeled as a part-based problem formulated as energy minimization, where the energy (or cost) is expressed as the linear combination of a unary fitting term and a pair-wise potential term. Following this work, we represent the global face shape model as a set of inter-related components, and model their spatial relationships. Unlike [16] where the unknown parameter space is low dimensional, in our approach we need to estimate component locations as well as shapes, which is a higher dimensional space. Therefore, we adopt PCA to model both the component shape variation and the relative spatial configuration. Due to the component decomposition, our approach is more flexible in modeling face shapes for handling larger pose changes and has better generalization capabilities than the standard ASM. Our approach uses PCA to model the relative positions between components, hence it is simpler and more efficient than [4] which relies on expensive MRF inference.

3 Helen Facial Feature Dataset

Our goal is a facial feature localization algorithm that can operate reliably and accurately under a broad range of appearance variation, including pose, lighting, expression, occlusion, and individual differences. In particular, it is necessary that the training set include high resolution examples so that, at test time, a high resolution test image can be fit accurately. Although a number face databases exist, we found none that meet our requirements, particularly the resolution requirement. Consequently, we constructed a new dataset using annotated Flickr images.

Specifically, the dataset was constructed as follows: First, a large set of candidate photos was gathered using a variety of keyword searches on Flickr. In all cases the query included the keyword "portrait" and was augmented with different terms such as "family", "outdoor", "studio", "boy", "wedding", etc. (An attempt was made to avoid cultural bias by repeating the queries in several different languages.) A face detector was run on the resulting candidate set to identify a subset of images that contain sufficiently large faces (greater than 500 pixels in width). The subset was further filtered by hand to remove false positives, profile views, as well as low quality images. For each accepted face, we generated a cropped version of the original image that includes the face and a proportional amount of background. In some cases, the face is very close or in contact with the edge of the original image and is consequently not centered in the cropped image. Also, the cropped image can contain other face instances since many photos contain more than one person in close proximity.

Finally, the images were hand-annotated using Amazon Mechanical Turk to precisely locate the eyes, nose, mouth, eyebrows, and jawline. (We adopted the same annotation convention as the PUT Face Database [17].) To assist the Turk worker in this task, we initialized the point locations to be the result of the STASM [1] algorithm that had been trained on the PUT database. However, since the Helen Dataset is much more diverse than PUT, the automatically initialized points were often far from the correct locations.

In any case, we found that this particular annotation task required an unusual amount of review and post-processing of the data in order to ensure high quality results. Ultimately this is attributable to the high number of degrees of freedom involved. For example, it frequently happened that a Turk worker would permute components (swap eyes and brows or inner lip for outer lip), or alternatively shift the positions of the points sufficiently that their roles would be changed (such as selecting a different vertex to serve as an eye or mouth corner). Graphical cues in the interface, as well as a training video and qualifying test were employed to assist with the process. Also, automated processes were developed to enforce consistency and uniformity in the dataset. In addition to the above, the faces were reviewed at the component level manually by the authors to identify errors in the annotations. Components with unacceptable error were resubmitted to the Turk for correction.

The resulting dataset consists of 2000 training and 330 test images with highly accurate, detailed, and consistent annotations of the primary facial components.

A sampling of the dataset is depicted in Fig. 1. The full dataset is publicly available at `http://www.ifp.illinois.edu/~vuongle2/helen`.

4 Facial Feature Localization

We first briefly review the global ASM, and then derive our new components-based model and the facial feature localization algorithm.

4.1 Classic ASM

In the classical global ASM, the shape is holistically represented as a set of pre-defined points, called *landmarks*, along the shape contour (Fig. 1 left), and described by a vector concatenating all the x coordinates of the ordered landmark points followed by all the y coordinates. We relate one shape with another using similarity transformation.

ASM is composed of two submodels: the *profile model* and the *shape model*. At each landmark, the profile model is the normalized gradient vector in the direction orthogonal to the shape boundaries. The profile distance is computed as the Mahalanobis distance over the training set. The global shape model is the statistics of the global shape vector for the training set, and is represented as a PCA linear subspace.

Given a face rectangle obtained by a generic face detector, ASM fitting first initializes the landmark points by placing the mean shape[1] to the detected face frame, and then repeats the following steps: (i) adjust each landmark points independently by profile template matching; (ii) fit a global shape model to the adjusted point set. Due to the greedy scheme, many of those individually adjusted points might be incorrect, and the point set will be regularized based on the global shape model.

4.2 Component-Based ASM

Shape Model. The classical global ASM relies on a dense correlation matrix of all the member landmark locations in a single Gaussian model. As a consequence, it imposes strong spatial constraints for every pair of landmarks in the face. These constraints make the model work fairly well on studio data where the variances are small and test images are similar to training samples. However, when applied to natural photos with large variations in face postures and occlusions, the full correlation matrix of the global ASM tends to be too strict and cannot be simply generalized for wide variety of face configurations. An example of a failure case for the global ASM is shown in Fig.2(a).

Inspired by this observation and the pictorial structure work [16], we introduce a new component-based model referred to as the *Component-based ASM (CompASM)* for handling the wide variety of variations in natural face images. In our

[1] The mean shape refers to the average of aligned shapes for training faces, where alignment is done by fitting with similarity transformation.

model, shape variation of each facial component is modeled independently, up to a similarity transformation, and the relative positions of facial components are encoded by a configuration model.

Fig. 2. (a) A failure example of the classic ASM on a face image by STASM [1]. The bigger left eye constrains the right eye and the tilted right brow pulled the left brow off the correct location. (b) A better fitting result using our CompASM.

We specifically consider a facial landmark set as a union of seven components: 'jawline', 'nose', 'lips', 'left eye', 'right eye', 'left brow', 'eye brow'. Each component has its own coordinate frame called *local frame*, centered at the center of mass of the component. Those centers are successively represented in a higher level canonical face frame called the *configuration frame*. An illustration of our component-based model is shown in Fig.3.

In this model, there are three coordinate frames: global, configuration and local frames. The j-th landmark point of component i has its global coordinates p_{ij} on the global coordinate frame, and local coordinates q_{ij} on the local coordinate frame.

In the local frame, the coordinates of landmark points for a component are represented as a single shape model centered at the component's centroid. It is a combination of principal vectors by PCA:

$$q_i = \bar{q_i} + \Phi_i b_i \tag{1}$$

where q_i is the concatenation of q_{ij}, $\bar{q_i}$ and Φ_i are the mean vector and the shape basis learnt from the training data respectively, and b_i denotes the set of linear coefficients of the fitting.

In the configuration frame, the location of each component's centroid is represented by a displacement vector t_i from the face center. The linear subspace representation of these vectors is learnt to be:

$$t = \bar{t} + \Psi r, \tag{2}$$

where Ψ is configuration bases, and r denotes the set of PCA coefficients.

The global coordinates of the landmark points can be obtained from local coordinate and configuration coordinates by a similarity transform:

$$p_{ij} = sR(q_{ij} + t_i) + t_0, \tag{3}$$

where R, s and t_0 denote the rotation, scale and translation that align a face shape from the global frame with the configuration frame; t_i denotes the location

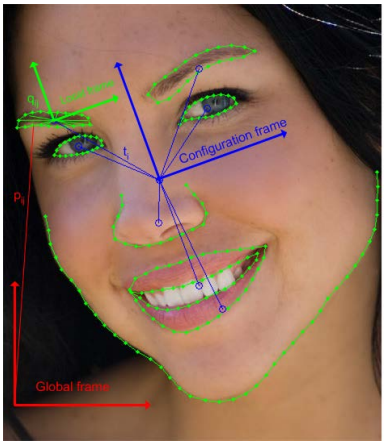


Fig. 3. Illustration of our component-based model

coordinates of component i in configuration frame. While t_i are different from component to component, the three similarity transform parameters in s and R are shared among the components.

Taking the mean of Eq.3 for each component, we obtain: $\bar{p_i} = sR(\bar{q_i} + t_i) + t_0$, where $\bar{q_i}$ is the local coordinate of centroid of component i and is effectively zero. And we can re-write it as $\bar{p_i} = sRt_i + t_0$.

Further combining the equation for all components, the model can be written as concatenated form as:

$$\bar{p} = sRt + t_0. \tag{4}$$

Eq. 4 and 2 are directly used in the configuration fitting step in the shape search algorithm. See Section 4.3 for the detailed algorithm.

In CompASM, the configuration model constrains relative locations between face components and is responsible for finding the orientation and scale of the face and estimating optimal locations of components, while the local model for each component is responsible for optimally fitting the component's shape model to the observation. In other words, each local model is fitted independently, hence can handle larger global shape variations.

Furthermore, the fact that each component shape is a single connected contour opens the possibility of jointly searching for multiple landmark points in the profile model. The new profile model adopting joint landmark optimization is introduced in the following section.

Profile Model. In the classic ASM [6, 1], during the profile fitting step, at each of the N landmarks of a given component we consider M candidate locations along a line orthogonal to the shape boundary. We evaluate the profile model at each location and pick the one with the highest score. The problem with this greedy algorithm is that each landmark location is chosen independently of its neighbors and neighboring landmarks could end up far from each other.

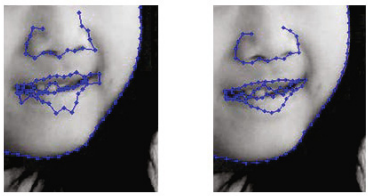


Fig. 4. An example of profile search by greedy method of classic ASM (a) and by our Viterbi global optimization (b)

To address this problem we introduce a binary constraint for each pair of adjacent landmarks to ensure that they are at an appropriate distance from each other. We jointly fit all the landmark locations by finding the sequence of candidate locations with a maximum sum of unary scores (at each candidate location) and binary scores (at each pair of adjacent locations). We can find the globally optimal such sequence in $O(NM^2)$ time using dynamic programming (the Viterbi algorithm [18]).

Our unary score is the probability of candidate location i of a given landmark, defined as:

$$p_i = (1 - d_i)/(M - 1) \tag{5}$$

where d_i is the Mahalanobis distance at the candidate location, normalized so that $\sum_i^M (d_i) = 1$ which ensures that $\sum_i^M (p_i) = 1$.

Our binary score is the probability that two adjacent landmarks are a given distance x apart. We model it using a continuous Poisson:

$$p(x) = \frac{\lambda^x e^{-\lambda}}{\Gamma(x+1)} \tag{6}$$

where x is the Euclidean distance between the two locations and Γ is the gamma function. We chose the Poisson distribution because it has a suitable PDF and a single parameter λ which we fit separately for each component.

To find the optimal set of candidate locations we run Viterbi, which maximizes the sum of the log probabilities. In our experiment section we show that our joint optimization of the landmarks outperforms the greedy approach in the traditional ASM algorithm. In addition, our model easily accommodates user interaction as discussed in Section 5. An example of the profile search results of two methods is depicted in Fig.4.

4.3 Our Algorithm

Here we describe how we combine our new shape model and profile model into the facial feature localization algorithm. Given an input image, after initialization by face detection, the shape model and profile model will be applied in an interleaved, iterative way in a coarse-to-fine manner in the pyramid.

In each particular iteration step, profile search is applied first, which gives the suggested candidate locations of the landmarks. The landmark candidates are then regularized by the local shape model. In each pyramid level, the fitting process is performed by repeating configuration model fitting and local model fitting until convergence.

The details of our facial feature localization method using CompASM is illustrated in Algorithm 1.

Algorithm 1. Facial Feature Localization Algorithm

Detect faces, initialize the shape model based on the face rectangle
for each resolution level **do**
  repeat
    a. Do profile search for suggesting new landmark locations
      a1. Collect unary scores by profile matching
      a2. Jointly find optimal path using the Viterbi algorithm
    b. Update the landmark localtions with local shape and conf. model
      b1. Find the centroid of suggested landmarks for each component
      b2. Fit the centroids to the configuration model using Eq.2 and 4
      b3. Apply the new configuration to the suggested landmarks using Eq.3
      b4. Fit the local shape model PCA subspace to landmarks using Eq.1
    c. Form a new global shape by applying the inverse similarity transformation
  until Number of points moved between two consecutive rounds $< 20\%$
  Map the localized result to the next resolution level
end for
Return the result from the highest resolution lvel

5 User Interaction Model

In this section we describe our algorithm for user-assisted facial feature localization. Once the automatic fitting is performed, the user is instructed to pick the landmark with the largest error and move it to the correct location after which we adjust the locations of the remaining landmarks to take into account the user input. This interaction step is called an *interaction round*. This process is repeated until the user is satisfied with the results. Our goal is to adjust the locations of the remaining landmarks as to minimize the number of interaction rounds.

Our update procedure consists of two main steps: *Linearly scaled movement* followed by a *Model fitting update*.

In the *Linearly scaled movement* step, when the user moves a landmark p_i, we move the neighboring landmarks on the same contour as p_i along the same direction. The amount of movement for neighboring landmarks is proportional to their proximity to p_i. Specifically, we identify landmarks p_a and p_b on both sides of p_i which define the span of landmarks that will be affected. Each landmark with index $j \in (a, i]$ moves by $\frac{a-j}{a-i} * d$ where d is the user-specified displacement

vector of p_i[2]. The landmark p_a is specified as the closest along the contour to p_i of the following three landmarks: (1) the one $\lfloor n/4 \rfloor$ landmarks away, where n is the number of landmarks in the component, (2) the nearest user-corrected landmark and (3) the end landmark if the component contour is open. The landmark p_b on the other side of the contour is determined analogously.

After the first step, in most cases, the neighboring points of p_i have moved close enough to the real curve that their profile models can snap to the right locations. That enables the second step of *Model fitting update*. In this step, we do several component based ASM fitting iterations at the highest resolution. Unlike regular ASM fitting, in these iterations we update only the current component. We also adjust the fitting so that the user-corrected landmarks don't move. Specifically, as we describe in Section 4, our fitting step iterates between shape model and profile model fitting. During the shape model fitting, we utilize constrained linear least squares to find the best location of the component given that it must go through the fixed landmarks. In our profile fitting, as we describe in Section 4.2 we specify candidate locations for each landmark and use the Viterbi algorithm to find the optimal contour. For user-specified landmarks, we choose the user-specified location as the only candidate location, thereby forcing Viterbi to find a path through the user-specified landmarks.

6 Experiments

6.1 Comparison to STASM

Our algorithm is designed to allow for high independence and freedom in component shape so that it can deal with natural photos with higher variations than the studio data. Nevertheless, we needs to assure that it can work adequately well on standard studio data as well. Therefore, in our first experiment, we compare our algorithm with STASM [1, 19] on standard datasets that they have chosen in their work. In this experiment, following [19] we train we train our algorithm on MUCT dataset [19] and test on BioID dataset [20]. MUCT contains 3755 images of 276 subjects taken in studio environment. Bioid consists of 1521 frontal images of 23 different test persons taken at the same lighting.

On the BioID test set, the fitting performance of STASM which features the stacked model and 2D profile trained on MUCT is reported to be the best among current algorithms [19]. The MUCT and BioID point sets includes five standalone points; therefore their shapes cannot be separated into single contours. Consequently, we are not able to use Viterbi optimized profile searching in this experiment. Following STASM, we evaluate the performance using the me-17 measure. This measure is the mean of distance between fitting result of 17 points in the face to the manually marked ground truth divided by the distance between two eye pupils. The comparison of the two algorithms' performance is shown in Table 1. The result shows that both of the algorithms' results on BioID are very close to perfect and STASM made a slightly lower error.

[2] Note that neighboring landmarks in our dataset are equidistant.

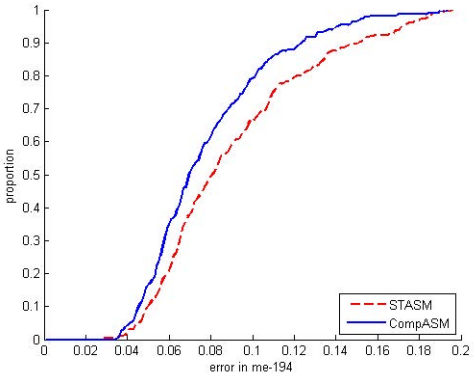


Fig. 5. CDF of me-194 error for CompASM and STASM on the Helen dataset

Table 1. Comparison of STASM and CompASM on the two test sets. For MUCT/BioID we use the me-17 measure. For Helen, we use the me-194 measure.

Dataset	Algorithm	Mean	Median	Min	Max
MUCT/BioID	STASM	0.043	0.040	0.020	0.19
MUCT/BioID	CompASM/Greedy	0.045	0.043	0.021	0.23
Helen	STASM	0.111	0.094	0.037	0.411
Helen	CompASM/Greedy	0.097	0.080	0.035	0.440
Helen	CompASM/Viterbi	0.091	0.073	0.035	0.402

While BioID is an easy dataset for fitting algorithms to deal with, the Helen dataset is a more challenging one. The Helen data consists of natural images which not only vary in pose and lighting but are also more diverse in subject identities and expression. In this dataset, the face features are highly uncorrelated with non-gaussian distribution which our model can exploit better. To measure fitting performance on Helen, we use me-194, similar to the me-17 measure, which calculates the mean deviation of 194 points from ground truth normalized by distance between two eye centroids. We divided Helen images into training set of 2000 images and testing set of 330 images. The comparison of STASM and CompASM is shown in Table 1 and Figure 5. Table 1 shows that CompASM outperforms STASM by 16%. The result proves that CompASM is more robust over diverse face images. Some examples of fitting results of STASM and CompASM are shown in Fig. 6.

6.2 Evaluation of User Interaction

Here we evaluate the effectiveness of our method described in Section 5 to refine the automatically determined facial component locations through user interaction. Qualitatively, we wish to minimize the user's effort in obtaining a

Fig. 6. Some examples of fitting results on the Helen test set of CompASM (first row) and STASM (second row)

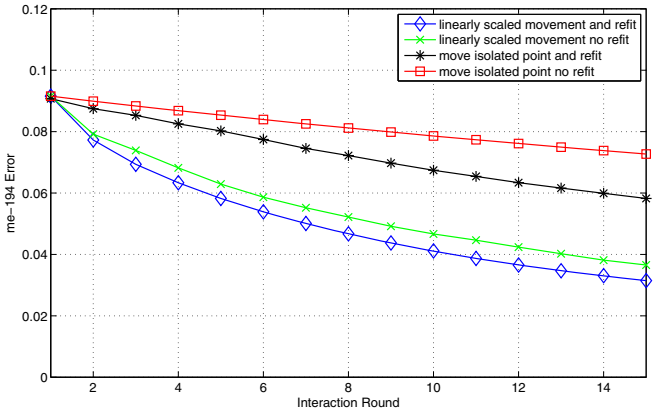


Fig. 7. The mean me-194 error measure across the Helen test set as a function of interaction round. One interaction round constitutes a single user action (see text for details). Each curve is the mean value across the test set under varying experimental conditions. Using both the linearly scaled movement and constrained refitting steps results in the most effective reduction in error.

satisfactory fitting result. More concretely, we wish to observe how the fitting error decays as a function of the number of interaction steps.

To this end, we simulate the user's typical behavior when interacting with the system, namely to choose the landmark point located farthest from its true position and move it to its true position. After being moved, it becomes a constraint for subsequent fitting rounds as described in Sec. 5.

We repeat this procedure for a number of rounds and record the me-194 error measure at each round. Fig. 7 depicts the distribution of the error across our test set over 15 interactivity rounds. To evaluate the effectiveness of the two steps of interactivity described in Sec. 5, we compare the performance of the case

Fig. 8. Examples of a sequence of editing steps where the most erroneous point is selected at each step. The red filled point denotes the point selected for editing. The open yellow circles denote points that have already been edited and fixed.

where both of the update steps are done with that of the case where only one of the steps is done, and also against a baseline where only the most erroneous point is moved. The comparative effectiveness of these strategies is apparent in Fig 7, from which we can conclude that both steps used in conjunction are most effective at reducing the overall error most quickly. Fig. 8 depicts some example runs of the interaction process. More examples with animation are available at the website `http://www.ifp.uiuc.edu/~vuongle2/helen/interactive`

7 Conclusion

In this paper, we proposed a new component-based ASM model (CompASM) and an interactive refinement algorithm for practical facial feature localization on real-world face images. CompASM extends the classical, global ASM by decomposing global shape fitting into two modules: component shape fitting and configuration model fitting. The model provides more flexibility for individual component's shape variation and relative configuration between components, hence is more suitable for handling images with large variation. We improve profile matching by introducing a new joint landmark optimization scheme using the Viterbi algorithm which outperforms the standard greedy-based approach. We also propose an interactive refinement algorithm for facial feature localization to minimize fitting errors for bad initial fitting. Furthermore, a new challenging facial feature localization dataset is introduced which contains more than 2000 high-resolution fully annotated real-world face images. Experimental results show that our approach outperforms the state of the art on the new, real-world facial feature localization dataset. Our interactive refinement algorithm is capable of reducing errors quickly but there is space to improve it further by learning from user's behavior or by guiding the interaction through statistics of landmark uncertainties, which we leave as our future work.

References

1. Milborrow, S., Nicolls, F.: Locating Facial Features with an Extended Active Shape Model. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 504–513. Springer, Heidelberg (2008)
2. Liang, L., Xiao, R., Wen, F., Sun, J.: Face Alignment Via Component-Based Discriminative Search. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 72–85. Springer, Heidelberg (2008)
3. Gu, L., Kanade, T.: A Generative Shape Regularization Model for Robust Face Alignment. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 413–426. Springer, Heidelberg (2008)
4. Tresadern, P., Bhaskar, H., Adeshina, S., Taylor, C., Cootes, T.: Combining local and global shape models for deformable object matching. In: BMVC (2009)
5. Zhu, X., Ramanan, D.: Face Detection, Pose Estimation, and Landmark Localization in the Wild. In: CVPR (2012)
6. Cootes, T., Taylor, C.J., Cooper, D., Graham, J.: Active shape models - their training and application. Computer Vision and Image Understanding 61, 38–59 (1995)
7. Cootes, T., Edwards, G.J., Taylor, C.J.: Active appearance models. IEEE Trans. PAMI 23, 681–685 (2001)
8. Cootes, T., Taylor, C.: A mixture model for representing shape variation. Image and Vision Computing 17, 567–574 (1999)
9. Romdhani, S., Gong, S., Psarrou, A.: A multi-view non-linear active shape model using kernel pca. In: BMVC (1999)
10. Rogers, M., Graham, J.: Robust Active Shape Model Search. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part IV. LNCS, vol. 2353, pp. 517–530. Springer, Heidelberg (2002)
11. Van Ginneken, B., Frangi, A., Staal, J., Ter Haar Romeny, B., Viergever, M.: Active shape model segmentation with optimal features. IEEE Trans. Medical Imaging 21, 924–933 (2002)
12. Zhou, Y., Gu, L., Zhang, H.: Bayesian tangent shape model: Estimating shape and pose parameters via bayesian inference. In: CVPR (2003)
13. Saragih, J.M., Lucey, S., Cohn, J.F.: Deformable model fitting by regularized landmark mean-shift. Int. J. Comput. Vision 91, 200–215 (2011)
14. Huang, Y., Liu, Q., Metaxas, D.: A component based deformable model for generalized face alignment. In: ICCV (2007)
15. Belhumeur, P.N., Jacobs, D.W., Kriegman, D.J., Kumar, N.: Localizing parts of faces using a consensus of exemplars. In: CVPR 2011, pp. 545–552 (2011)
16. Felzenszwalb, P.F., Huttenlocher, D.P.: Pictorial structures for object recognition. International Journal of Computer Vision 61, 55–79 (2005)
17. Kasinski, A., Florek, A., Schmidt, A.: The PUT face database. Image Processing and Communications 13, 59–64 (2008)
18. Forney Jr., G.D.: The Viterbi algorithm. Proceedings of IEEE 61, 268–278 (1973)
19. Milborrow, S., Morkel, J., Nicolls, F.: The MUCT Landmarked Face Database. Pattern Recognition Association of South Africa (2010)
20. Jesorsky, O., Kirchberg, K.J., Frischholz, R.: Robust Face Detection Using the Hausdorff Distance. In: Bigun, J., Smeraldi, F. (eds.) AVBPA 2001. LNCS, vol. 2091, pp. 90–95. Springer, Heidelberg (2001)