An Analysis of the Sensitivity of Active Shape Models to Initialization when Applied to Automatic Facial Landmarking

Keshav Seshadri, Student Member, IEEE and Marios Savvides, Member, IEEE

Abstract—Active Shape Models (ASMs) have recently gained popularity for performing automatic facial landmark fitting. Their demonstrated ability to generalize and fit unseen faces make them ideal candidates for this task unlike traditional Active Appearance Model (AAM) based approaches, which have difficulty in accurately landmarking unseen images. Given a test image, a face detector is used to determine the locations, orientations and sizes of faces in the image. Facial landmarking algorithms, such as ASMs, are initialized based on these parameters. In this paper, we conduct a series of experiments to exhaustively evaluate the tolerance of three popular ASMs to initialization perturbations (translation, rotation, and scaling in size) of the face detected, a topic that has not been analyzed in depth to-date. Our results are consistent across different databases, provide an understanding of the role initialization plays in the landmark fitting process and serve as a performance gauge that could be considered when comparing facial landmarking algorithms.

Index Terms—Sensitivity analysis, Automatic facial landmarking, Active Shape Models (ASMs), Active Appearance Models (AAMs).

I. INTRODUCTION

The automatic localization of facial landmarks is a key preprocessing step that can aid in boosting facial recognition rates [1], [2], [3], generation of 3D facial models [4], [5], [6], expression analysis [7], [8] and pose estimation [9], [10]. Active Shape Models (ASMs) [11] and Active Appearance Models (AAMs) [12] can be used for locating these facial landmarks and have been well researched over the past few years. For the purpose of solely performing automatic facial landmarking (and not facial texture reconstruction), ASMs, in general, are easier to train than AAMs and have also been shown to be more accurate in fitting unseen faces by Cootes et al. in [13]. This is because they build local texture models which are less susceptible to illumination variations than the holistic global texture models that are employed by AAMs. For this reason, we focus our work in this paper on ASM based methods of facial landmarking.

A key requirement to ensure accurate ASM or AAM fitting is that the object’s location in the image must be roughly known in order to accurately initialize them. Thus, for the application of locating facial landmarks, such as the corners of the eyes or the tip of the nose, an ASM or AAM requires the use of a face detector before the fitting process can begin. Though numerous studies have been carried out in order to improve the overall fitting accuracy of ASMs [14], [15], [16], [17], the effects of initialization on their performance has been largely ignored. Cootes, Edwards and Taylor have initialized an ASM using different perturbations and observed the results on ASM fitting using a small dataset in [18] while Lui et al. [19] have underscored the importance of initialization to AAM fitting, however, till date, to the best of our knowledge, there has not been a large scale quantitative evaluation of the role of scale, translation and rotation effects in ASM fitting. Hence, the aim of this paper is to provide a large scale study into understanding the effects of initialization and how sensitive or robust certain ASMs are to such initialization perturbations when applied to the task of automatic landmarking of frontal faces.

We focus our experiments on three popular and commonly used ASM algorithms. The first, is the classical ASM that is outlined by Cootes, Taylor and Lanitis in [18] and described in greater detail by Cootes and Taylor in [20]. The second and third methods are more recent ones that are respectively described by Milborrow and Nicolls in [21] and by Seshadri and Savvides in [22]. We also benchmark the performance of these ASMs against the classical AAM approach described in [12] in order to emphasize why ASMs are more suited to the task of facial landmarking. In order to quantitatively measure the influence of initialization on these algorithms, we carry out four rigorous experiments. The first experiment compares the fitting accuracy of the methods on large sets of images drawn from the NIST Multiple Biometric Grand Challenge - 2008 (MBGC-2008) still face challenge database [23], [24], the CMU Multi-PIE (MPIE) database [25], and the Japanese Female Facial Expression database (JAFFE) [26]. In this baseline experiment we measure the fitting accuracy of the ASMs and the classical AAM approach when no errors in initialization parameters are present. Our subsequent experiments examine the influence of initialization errors (translation, rotation, and scaling effects) on the algorithms using test sets which are subsets of the previously mentioned databases. Our results are consistent across the different test sets and not only enable a fair comparison of the fitting accuracy of the afore mentioned landmarking methods but also serve in understanding the effects of initialization errors that are inherently added by face detection modules and the sensitivity of these algorithms towards such initialization errors.

The remainder of this paper is organized as follows. Section
II describes and contrasts how automatic facial landmarking is carried out using traditional ASM and AAM approaches while section III provides a description of the three ASM algorithms under evaluation. Section IV overviews the experiments that were carried out on these ASMs in order to understand the effects of initialization, analyzes the results obtained, and also benchmarks their performance against that of the classical AAM approach. Finally, section V presents our conclusions on this study.

II. ACTIVE SHAPE MODELS AND ACTIVE APPEARANCE MODELS

In this section we provide an overview of how traditional Active Shape Models (ASMs) and Active Appearance Models (AAMs) work when they are applied to the problem of automatic facial landmarking.

A. Traditional Active Shape Model

1) ASM Training Stage: Building an ASM to automatically locate facial features in an image first requires an off-line training stage. In this training stage, a facial shape model (or a Point Distribution Model (PDM)) and a local texture model are built. Each face (or shape) can be described by the coordinates of \( N \) landmark points by which it is annotated. We have developed a landmarking scheme to ensure that faces can be easily hand-labeled by human operators in order to provide consistent ground truth data. The scheme consists of 79 densely placed points as shown in Fig. 1. Fig. 1 also shows how landmarks (such as 2,3,4,5,6, and 7) whose positions appear slightly ambiguous can be placed using their relationship to more clearly identifiable landmarks such as the corners of the eyes (landmarks 16,20,24, and 28) or tip of the nose (landmark 79). Due to the large number of images that had to be manually annotated, it was not possible to get each image annotated by more than one person and so a thorough analysis of the variance or standard deviation in human landmarking across multiple images could not be made. However, we were able to conduct a small test to approximate this standard deviation (such as provided by Jepson et al. in [27]) using a few images from the MBGC database that were landmarked by ten different people. The estimated average error in human landmarking across all 79 landmarks and all images was calculated to be 1.77 pixels with a standard deviation of 1.44 pixels when all Euclidean distances were normalized to correspond to a face with 50 pixels between the eyes. These values will put the mean and standard deviation of fitting errors obtained by the automatic facial landmarking algorithms in section IV into better context.

The \( x \) and \( y \) coordinates of each landmark are stacked to form a shape vector \( \mathbf{x} \) as depicted in (1) in which \( x_i \) and \( y_i \) are the \( x \) and \( y \) coordinates of the \( i^{th} \) landmark.

\[
\mathbf{x} = [x_1 \ x_2 \ldots \ x_N \ y_1 \ y_2 \ldots \ y_N]^T \tag{1}
\]

The \( n \) shapes \((x_1, x_2 \ldots x_n)\) in the training set are now aligned using Generalized Procrustes Analysis [28] to minimize the mean squared error between their landmark locations. This is the first step in determining a canonical facial shape representation. The mean shape \( \mathbf{X} \) of the aligned shapes is then computed using (2) and is subtracted from each of the \( n \) shapes. The resultant vectors are stored in a matrix \( \mathbf{X} \).

\[
\mathbf{x} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \tag{2}
\]

Principal Component Analysis (PCA) is now used to determine a matrix \( \mathbf{P}_s \) of orthonormal basis vectors to best represent the training set facial shape variation. The objective function it seeks to minimize is given in (3).

\[
\arg\min_{\mathbf{P}_s} \| \mathbf{X} - \mathbf{P}_s \mathbf{P}_s^T \mathbf{X} \|^2 \tag{3}
\]

The solution determined by PCA is that the matrix \( \mathbf{P}_s \) consists of the eigenvectors corresponding to the dominant eigenvalues of the covariance matrix \( \mathbf{S}_s \) of the aligned shapes. Thus, in order to determine a linear subspace that describes facial shape variation in the training set, we first compute the covariance matrix \( \mathbf{S}_s \) of the aligned shapes using (4), as well as the eigenvalues and eigenvectors of \( \mathbf{S}_s \).

\[
\mathbf{S}_s = \frac{1}{n-1} \sum_{i=1}^{n} (\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^T \tag{4}
\]
The $t$ orthonormal eigenvectors corresponding to the eigenvalues that model $v\%$ of the total face shape variation are stored along the columns of $P_s$. A shape $x$, drawn from the training set or from outside it, can now be approximated using the shape model equation given in (5), in which $b_s$ is a vector of the projection coefficients of the basis vectors such that a linear combination of these vectors closely approximates $x$ in a least squares sense.

$$ x \approx \bar{x} + P_s b_s \quad (5) $$

The next phase of training involves building statistical models of the local appearance (grayscale pixel intensities) of regions surrounding each landmark. A 1D profile is built by obtaining the pixel intensities along a line through a landmark, sometimes referred to as a whisker. This line is of length $2k + 1$ with $k$ pixels sampled on either side of a landmark and is perpendicular to the direction of the shape boundary at the landmark. The derivative of this vector is computed and it is then normalized, thus giving rise to a profile vector $g$. The mean profile vector $\bar{g}$ of all such profile vectors (averaged over the entire training set) is computed for each landmark location and the corresponding covariance matrix $S_g$ of the profiles at each landmark location is also computed. The process is repeated to generate these statistics for $L_{max}$ multiple resolutions of an image pyramid where images at level $j$ have half the height and width of images at level $j - 1$ with $j$ being varied from 1 to $L_{max}$. It is to be noted that profiles are built without first performing illumination normalization of the images using techniques such as histogram equalization etc. as the extracted profiles are separately normalized to compensate for local illumination variation.

2D profiles capture more information than their 1D counterparts and hence most recent ASM algorithms build 2D profiles by sampling the image gradients in $m \times m$ square regions around each landmark. These local image sections are then vectorized and each element $g_i$ of the profile $g$ is transformed, using a constant parameter $u$, into $g_i'$ as shown in (6).

$$ g_i' = g_i / (|g_i| + u) \quad i = 1, 2 \ldots m^2 \quad (6) $$

The multi-resolution framework and the generation of mean profile vectors and covariance matrices for each landmark remain the same as for the 1D profiles. Fig. 2 illustrates how 1D and 2D profiles are constructed.

2) ASM Testing Stage: A trained ASM can be used to fit the predefined landmarks onto any image containing a face once the location of the face has been determined using a face detector. We use the OpenCV implementation [29] of the Viola-Jones face detector [30] for this purpose, which is a widely used face detector in research literature.

In the first stage of the fitting process, the mean shape is aligned over the detected face using a similarity transform (alignment using only translation, rotation, and scaling). In our work, this process is carried out by determining the similarity transform that aligns landmarks 3 and 13 of the mean face with the mid points on opposite sides of the facial bounding box and applying the same transform to all points in the mean shape. After this alignment step, profiles are constructed for several candidate points around each landmark location. A 1D search evaluates candidate locations at $n_p$ locations on either side of each landmark along the whisker through it. A 2D search evaluates locations in a $n_p \times n_p$ region around a landmark and is hence slower due to more computations but more accurate and thorough than a 1D search. In both cases however, the best candidate is the one whose surrounding profile $g$ has the highest probability $P(g)$ of being drawn from a multivariate Gaussian distribution with mean $\bar{g}$ of dimensionality $d = 2k + 1$ for 1D profiles or $d = m \times m$ for 2D profiles and covariance matrix $S_g$ of size $d \times d$, since the profiles surrounding each landmark are modeled by such a distribution. This is shown in (7).

$$ P(g) = \frac{1}{(2\pi)^{d/2}|S_g|^{1/2}} \exp\left\{-\frac{1}{2}(g - \bar{g})^T S_g^{-1} (g - \bar{g})\right\} \quad (7) $$

Maximizing $P(g)$ is equivalent to minimizing the squared Mahalanobis distance between $g$ and $\bar{g}$, which is a part of the exponential term in (7). Hence, the objective function to be minimized when determining the best candidate location for a landmark is the squared Mahalanobis distance $f(g)$ between the profile $g$ surrounding the candidate and the mean profile $\bar{g}$ for the landmark in question and is given by (8).

$$ f(g) = (g - \bar{g})^T S_g^{-1} (g - \bar{g}) \quad (8) $$

All landmarks are moved into the optimal candidate locations for them to obtain a new shape vector $x_{ini}$ (also referred to as the image shape). The vector of shape coefficients $b_s$ for $x_{ini}$ is calculated by iteratively minimizing the objective function given in (9), in which $t_x, t_y, s,$ and $\theta$ are respectively the $x$-axis translation, $y$-axis translation, scaling parameter, and rotation angle of the similarity transform $T$ that minimizes the Euclidean distance between $x_{ini}$ and the reconstructed shape $x_{recon} = \bar{x} + P_s b_s$ (also referred to as the model shape).

$$ |x_{ini} - T_{t_x, t_y, s, \theta}(\bar{x} + P_s b_s)|^2 \quad (9) $$
The iterative process followed to jointly determine $b_s$ and $T$ can be found in [20] and is reproduced below for convenience.

1) Set all values in $b_s$ to zero
2) Generate the model shape: $x_{recon} = x + P_s b_s$
3) Determine the parameters $t_x, t_y, s, \theta$ of the similarity transform $T$ that best maps $x_{recon}$ to $x_{ini}$
4) Invert the parameters and use the inverted transform to project $x_{ini}$ into the model coordinate frame: $x_{mod} = T^{-1}(x_{ini})$
5) Update the model parameters to match $x_{mod}$: $b_s = P_s^T (x_{mod} - \bar{x})$
6) Constrain the elements of $b_s$
7) Return to step 2 if convergence has not been reached

The constraining of shape coefficients (referred to in the process for calculating $b_s$) is carried out so that the shape generated by them is representative of shapes in the training set (i.e. the shape generated is a legal shape). This is done by either one of two methods. The first method saturates the values of the shape coefficients ($b_s$) to lie within $b_{max}$ standard deviations of their zero mean values, i.e. the range $\pm b_{max} \sqrt{\lambda_i}$, where $\lambda_i$ is the $i^{th}$ eigenvalue retained by PCA and $b_{max}$ is usually set to 2 or 3 while the second method ensures that $b_s$ lies inside a hyper-ellipsoid using (10) and (11).

$$b_{new} = \sum_{i=1}^{t} \frac{b_{s_i}^2}{\lambda_i}$$

$$b_{s_i} = \frac{b_{s_i} \cdot b_{max}}{b_{new}} \quad i = 1, 2 \ldots t$$

A new shape is generated using the final set of constrained shape coefficients and is compared to the previous shape. The process is repeated until the change in locations of most landmarks is below a certain threshold or the maximum number of iterations $N_{max}$ is exceeded. Once convergence is reached at the lowest resolution stage, the landmark coordinates are scaled to correspond to the dimensions of the image at the next level of the image pyramid and the entire process is repeated until convergence at the highest resolution image at which stage the final landmark coordinates are ready. The process is outlined in Fig. 3.

B. Traditional Active Appearance Model

In this section we describe the basic Active Appearance Model (AAM) algorithm outlined by Cootes et al. in [12]. This approach is referred to as the classical AAM approach throughout this paper.

1) AAM Training Stage: An AAM has the same shape model equations and builds a shape subspace in identical fashion to an ASM. After this step is completed, it models texture by using a global approach rather than the local one adopted by ASMs. Yet again, no form of image pre-processing is carried out before obtaining the texture models. Shape free patches are generated for each training image by warping them so that their control points (landmarks) exactly match the mean shape $\bar{x}$. The grayscale pixel intensities from these shape free patches are now sampled and normalized. The normalized texture vectors are henceforth denoted by $t$. PCA is now applied on these texture vectors and as was the case with the shape model, a texture model (given in (12)) can be used to represent a normalized texture vector using the mean normalized texture vector $\bar{t}$ and an orthogonal basis of eigenvectors stacked along the columns of a matrix $P_t$.

$$t \approx \bar{t} + P_t b_t$$

The next step in constructing an AAM is to build a combined appearance model of shape and texture variation. This is carried out by extracting the shape coefficients vector $b_s$ and the texture coefficients vector $b_t$ and concatenating them to obtain a vector $b$ using (13). In (13), $W_s$ is a diagonal matrix of weights that compensates for the difference in units between shape and texture and allows their parameters to be successfully concatenated without one set of parameters dominating the other.

$$b = \begin{bmatrix} W_s b_s \\ b_t \end{bmatrix} = \begin{bmatrix} W_s P_t^T (x - \bar{x}) \\ P_t^T (t - \bar{t}) \end{bmatrix}$$

Once PCA is applied on the concatenated vectors $b$, they can be represented by a combined appearance model as shown in (14), in which $P_c$ is a matrix of the orthonormal basis of eigenvectors and $\bar{c}$ is the zero mean vector of the combined appearance coefficients that controls both the shape and texture of the model.

$$b \approx P_c c = \begin{bmatrix} P_{cs} \\ P_{ct} \end{bmatrix} c$$

2) AAM Testing Stage: An AAM is initialized in the same way as an ASM, by detecting a face and roughly aligning the mean shape $\bar{x}$ on top of it. From here on however, the AAM search is driven by a different approach than that of an ASM. The objective of an AAM is to minimize the L2-norm of the error vector $\delta t$ between the true shape free texture vector $t_i$ currently enclosed by the landmarks and the reconstructed
version of this texture vector $t_{mn}$ obtained by using the AAM combined appearance coefficients in $c$. The iterative search continues and updates $c$ until convergence, i.e. the global texture reconstruction error is minimized. The updates to $c$ at each step of the iteration are provided by (15).

$$c = R(t_i - t_{mn}) = R\delta t$$  \hfill (15)

In (15), the matrix $R$ is calculated during the training stage by systematically displacing each of the appearance parameters, observing the resultant change in texture and averaging the results over all training images and several iterations. The process by which $R$ is calculated and the assumptions made on the update rule are fully described in [12] and [20]. The vector $c$ is updated until convergence and similar to the case with ASMs, a multi-resolution approach with shape and texture models for each resolution can be used to increase the speed of convergence of the AAM as well as enhance its accuracy. The final set of landmarks as well as the reconstructed texture vector can be obtained as a function of the combined appearance coefficients $c$ using the linear relationships listed in (16) and (17) respectively.

$$x = x + P_sW_s^{-1}P cs c$$  \hfill (16)

$$t = t + P_tP_{st}c$$  \hfill (17)

The process by which an AAM fits a test image is shown in Fig. 4.

III. ACTIVE SHAPE MODEL ALGORITHMS COMPARED

We compare the effects of initialization on three Active Shape Model algorithms using experiments described in section IV. These are the classical ASM approach discussed in [18], [20], the Stacked ASM (Stasm) approach that is proposed in [21], and the Modified Active Shape Model (MASM) method that is described in [22].

A. Classical ASM Approach

The classical ASM approach is the most basic ASM implementation and uses most of the principles that are outlined in section II. It adopts a multi-resolution approach and thus obtains better fitting results than a model run on a single resolution, however, its use of 1D profiles is a fundamental limitation. 1D profiles do not model local texture accurately enough to distinguish the region around one landmark from the region around another and at the test stage this leads to the occurrence of local minima (false positives) when trying to determine the best location for a particular landmark. Additionally, classical ASM only performs a 1D search when evaluating potential landmark positions. This leads to good fitting speed but limits the amount landmarks can move from their initial locations. Thus, this method produces poor fitting results unless the landmarks are initialized extremely well.

B. Stacked ASM (Stasm) Approach

The Stacked ASM (Stasm) approach preserves most of the conventional ASM framework but incorporates certain changes that lead to better fitting performance. The most significant of these changes is the use of 2D profiles to better model local appearance as well as the use of a 2D search region around the initial locations of landmarks in order to move them into correct positions. Both these improvements lead to better fitting results, albeit at the cost of speed due to increased computations. Stasm also uses two ASM models that are stacked in series. The first model, that uses only 1D profiling and searching, is used to better initialize a second model which uses 2D profiles and a 2D search space. Other slight improvements are used to improve the speed of fitting such as the trimming of the 2D profile covariance matrix to make it sparse enough to speed up the computation of Mahalanobis distances between profiles as well as running the 2D search model for a lower number of image resolutions than the 1D search model.

The use of all these improvements helps Stasm achieve fairly accurate fitting results but its still suffers from problems due to local minima. Stasm also does not accurately fit landmarks along the lips and does not handle faces exhibiting different expressions or slight pose variation very well. To summarize, it is an implementation that combines several standard techniques to good effect but could use more modifications to the overall ASM structure in order to achieve higher precision.

C. Modified ASM (MASM) Method

The Modified Active Shape Model (MASM) method is a robust method that proposes theoretical changes to the traditional ASM framework. These changes help MASM achieve more accurate results than the previously described methods.

MASM uses only 2D profiles and 2D searches and completely does away with 1D profiles which are present in both classical ASM and Stasm. It also goes one step further than the previous algorithms and builds a subspace to model pixel variation in a square region around each landmark. This
subspace is spanned by the eigenvectors corresponding to the dominant eigenvalues, that model 97% of the texture variation, of the profile covariance matrix $S_g$ which are stacked along the columns of a matrix $P_g$. At the testing stage, a 2D profile $g$ around a candidate location is projected onto this subspace using (18) to obtain a vector of projection coefficients $b_g$ and is then reconstructed using these coefficients to obtain $g_r$ using (19).

$$b_g = P_g^T(g - g)$$

$$g_r = g + P_g b_g$$

The weighted reconstruction error $w(g)$ between this reconstructed profile and the original profile is calculated using (20) and it is this metric which is used to determine the best location for a landmark.

$$w(g) = (g_r - g)^T S_g^{-1} (g_r - g)$$

The candidate point deemed to be the most suitable location for the landmark is the one whose surrounding profile has the lowest weighted reconstruction error as such a profile is the one that can be best modeled by the subspace. The use of this local subspace mitigates the local minima problem to some extent and contributes to accurate fitting results. The process followed by MASM in moving a landmark is shown in Fig. 5.

Another feature of MASM is that edge detection is used to better fit points that lie along the facial outline (landmarks...
Initialization of an ASM or AAM is vital to ensuring high fitting accuracy. This initialization is usually provided by the bounding box around a face that is returned by a face detector. However, face detectors, such as Viola-Jones, are prone to errors that result in scale and translation effects in the bounding box. Since the bounding box is also generally upright, in-plane rotation of a face cannot be accounted for leading to rotation based initialization errors. In order to demonstrate the typical errors in Viola-Jones face detection, we carried out a simple test. We used frontal images of 249 subjects with a neutral expression across 20 illuminations in the MPIE database and ran the Viola-Jones face detector on these 4980 images. For each subject, only the illumination varied across the face (in less than a second using controlled flashes) and the subject was perfectly still (motionless). Thus, the face detection results should ideally be identical across all these 20 images of the subject and the same bounding box should be returned. However, this is not the case. For each subject, the mean $x$ and $y$ coordinates of the top left corner of the bounding box were computed. The Euclidean distance of the top left corner of the bounding box in all 20 images of that subject from this mean position were then obtained. The mean and standard deviation of these distances across all 4980 images of the 249 subjects are presented in the first two columns of Table I and are an indicator of the translation errors that creep in during face detection. In similar fashion, we computed the mean size (width or height) of the face for each subject, the mean number of pixels of the top left corner of the bounding box in all 20 images of that subject and the same bounding box should be returned. However, this is not the case. For each subject, the mean $x$ and $y$ coordinates of the top left corner of the bounding box were computed. The Euclidean distance of the top left corner of the bounding box in all 20 images of that subject from this mean position were then obtained. The mean and standard deviation of these distances across all 4980 images of the 249 subjects are presented in the first two columns of Table I and are an indicator of the translation errors that creep in during face detection. In similar fashion, we computed the mean size (width or height) of the face for each subject. The absolute difference of the facial size returned for the 20 images of that subject from the respective mean for that subject were then obtained and the third and fourth columns of Table I respectively show the mean and standard deviation of these values over all images. These values serve as an indicator of the scaling errors that can be introduced during face detection.

Currently, no existing ASM implementation can recover from extremely poor initialization. The question that arises is how sensitive they are are to initialization errors. In order to measure this, we conducted four experiments on multiple databases in which we artificially introduced initialization

<table>
<thead>
<tr>
<th>Method</th>
<th>MBGC</th>
<th>MPIE</th>
<th>JAFFE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Std. Dev.</td>
<td>Mean</td>
</tr>
<tr>
<td>Classical AAM (9 key landmarks)</td>
<td>3.35</td>
<td>2.49</td>
<td>7.67</td>
</tr>
<tr>
<td>Classical AAM (remaining 70 landmarks)</td>
<td>9.86</td>
<td>6.36</td>
<td>-</td>
</tr>
<tr>
<td>Classical AAM (all 79 landmarks)</td>
<td>9.61</td>
<td>6.21</td>
<td>-</td>
</tr>
<tr>
<td>Classical ASM (9 key landmarks)</td>
<td>4.43</td>
<td>2.85</td>
<td>3.84</td>
</tr>
<tr>
<td>Classical ASM (remaining 70 landmarks)</td>
<td>4.88</td>
<td>3.35</td>
<td>4.38</td>
</tr>
<tr>
<td>Classical ASM (all 79 landmarks)</td>
<td>4.83</td>
<td>3.40</td>
<td>4.32</td>
</tr>
<tr>
<td>Stasm (9 key landmarks)</td>
<td>2.86</td>
<td>2.17</td>
<td>3.33</td>
</tr>
<tr>
<td>Stasm (remaining 70 landmarks)</td>
<td>3.07</td>
<td>2.54</td>
<td>3.67</td>
</tr>
<tr>
<td>Stasm (all 79 landmarks)</td>
<td>3.00</td>
<td>2.49</td>
<td>3.62</td>
</tr>
<tr>
<td>MASM (9 key landmarks)</td>
<td>1.96</td>
<td>1.32</td>
<td>2.94</td>
</tr>
<tr>
<td>MASM (remaining 70 landmarks)</td>
<td>2.86</td>
<td>2.17</td>
<td>3.33</td>
</tr>
<tr>
<td>MASM (all 79 landmarks)</td>
<td>2.76</td>
<td>2.11</td>
<td>3.29</td>
</tr>
</tbody>
</table>

1-15 in Fig. 1). This is implemented using a simple weighting function [31] that is given by $q(g)$ in (21).

$$q(g) = (c - I)(g_r - g)^T S_g^{-1}(g_r - g) \quad (21)$$

In (21), $I$ is the Sobel edge intensity at the candidate location and can only take on the values 1 or 0, for points that lie along an edge and those that do not respectively, while $c$ is a constant whose value must be greater than 1 (we set this value to 2). The use of this weighting function ensures that the best location for a landmark along the facial outline not only bears a similar profile to the training data but also lies along a strong edge.

The final stage of MASM involves using a goodness of fit criteria for all the landmarks. Based on the reconstruction error metric, an empirically determined threshold is set and all points with a reconstruction error that is higher than this threshold are refitted at the finest resolution of the image pyramid. This refitting stage contributes to slight refinement in landmark positions and enhances fitting accuracy.

The use of all of the above modifications allow MASM to accurately fit facial landmarks on frontal faces. MASM also handles faces with slight pose variation and varied expressions satisfactorily as will be shown in section IV. From a computational standpoint though, classical ASM is much faster than Stasm and MASM as it only samples pixels along a whisker (and not in a square region) through a landmark and also evaluates fewer candidates in order to update landmark positions. Stasm uses a 1D search followed by a 2D search for some pyramid levels and is hence more computationally expensive than classical ASM. MASM is the most computationally expensive as it uses only 2D profiles and searches and also projects the extracted profile onto a PCA subspace, reconstructs it and evaluates the reconstruction error. Typically the time required by a MATLAB implementation of classical ASM to fit an image on a 32 bit Intel Core 2 Quad CPU Q6600 @ 2.4 GHz with 2 GB RAM is around 7 secs. The corresponding times for Stasm and MASM are 16 secs and 25 secs respectively.

IV. EXPERIMENTS AND RESULTS
errors and observed how this impacted the fitting accuracy of the ASMs. In each of the experiments we also compared the performance of the ASMs to that of the classical AAM approach in order to emphasize why ASMs are more suited to the task of automatic facial landmarking. For all of our experiments, the training set we used remained the same - a set of 4000 manually annotated images drawn from the MBGC-2008 still face challenge database which contains a total of 34,729 frontal images of 810 subjects. This ensured that no algorithm had an advantage over the others simply due
to training data. The values of the parameters used by us when comparing the three ASMs are listed in Table II. We used the same shape parameters for the classical AAM as the classical ASM approach.

A. Experiment 1 - Comparison of Fitting Accuracies in the Absence of Initialization Errors

Our first experiment compared the fitting accuracy of the three ASMs when initialization errors were not present i.e. when they were initialized under the assumption that the face detector results were free of translation, rotation, and scaling effects. For this experiment, we used three test sets: the first consisted of 30,729 images from the MBGC database (left over from the training set) while the second and third consisted of 1000 images (of 249 subjects with varying illumination) from the frontal view set of session 1 of the CMU Multi-PIE (MPIE) database and the entire set of 213 images in the JAFFE database respectively. Details on the databases can be found in Table III. The MPIE dataset was used by us to test the ASMs on images with illumination variations not modeled in their training set while the JAFFE dataset was used in order to observe their ability to fit unseen expressions.

The three ASMs were initialized using face detector results and allowed to fit the 79 facial landmarks onto all images in the three test sets. Images where the face was incorrectly detected (wrong position or scale or not detected at all) were removed from the dataset and not considered in our final analysis. The Euclidean distances between the coordinates of landmarks that were fitted automatically and those that were

Fig. 9. Error surfaces produced in experiment 2 by the different landmarking algorithms on the (a) MBGC test set (b) JAFFE test set (c) MPIE test set.

<table>
<thead>
<tr>
<th>Method</th>
<th>Test Dataset</th>
<th>MBGC</th>
<th>JAFFE</th>
<th>MPIE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classical AAM</td>
<td></td>
<td>$-10 \leq x \leq +10$</td>
<td>$-10 \leq x \leq +10$</td>
<td>$-10 \leq x \leq +10$</td>
</tr>
<tr>
<td>Classical ASM</td>
<td>$-10 \leq y \leq +20$</td>
<td>$0 \leq y \leq +30$</td>
<td>$-10 \leq y \leq +10$</td>
<td></td>
</tr>
<tr>
<td>Stasm</td>
<td>$-20 \leq x \leq +40$</td>
<td>$0 \leq y \leq +10$</td>
<td>$-20 \leq x \leq +20$</td>
<td></td>
</tr>
<tr>
<td>MASM</td>
<td>$-40 \leq x \leq +40$</td>
<td>$-20 \leq y \leq +20$</td>
<td>$-30 \leq y \leq +30$</td>
<td></td>
</tr>
</tbody>
</table>
Fig. 10. Fitting results obtained by the three ASMs after carrying out experiment 2 on some images from the MBGC test set (a) Classical ASM (b) Stasm (c) MASM. In (a), (b), and (c) the green dots and red lines indicate the final fitting results, the blue squares correspond to the true faces detected, and the green squares represent the displaced faces which were used to initialize the ASMs with the displacements corresponding to $-30$, $0$, and $+30$ pixels along the $y$-axis from top to bottom (rows of (a), (b), and (c)) and $-30$, $0$, and $+30$ pixels along the $x$-axis from left to right (columns of (a), (b), and (c)).
manually annotated were normalized to an interocular distance of 50 pixels based on the ground truths and served as our performance metric.

Fig. 6(a) compares the performance of the three ASMs on the MBGC test set. Each curve in Fig. 6(a) shows normalized fitting error values on the $x$-axis and the corresponding percentage of landmarks that were found to lie within the respective fitting error value along the $y$-axis. The curves are monotonically increasing and their steepness is an indicator of performance. Fig. 6(b) and Fig. 6(c) are similar to Fig. 6(a) and compare the performance of the ASMs on the MPIE and JAFFE test sets respectively. In all three cases, MASM outperforms Stasm and classical ASM by ensuring that a greater percentage of landmarks are fitted within a low range of fitting error values. The gap in performance between MASM and the other methods is highest on the JAFFE test set where it is able to fit the unseen expressions better. These results are also summarized in Table IV, which compares the mean and standard deviation of the normalized fitting errors across all landmarks obtained by the three ASMs on the three test sets.

Fig. 7(a), (b), and (c) are similar to those shown in Fig. 6 but separately consider the fitting error over 9 key fiducial landmarks (the corners of the eyes - landmarks 16, 20, 24, and 28, the corners of the mouth - landmarks 32, 38, 42, and 46, and the tip of the nose - landmark 79) and the remaining 70 landmarks and plot these the percentage of these values found to lie within specific normalized fitting error values. The curves clearly demonstrate that the 9 key fiducial points are localized with higher accuracy than the remaining 70 points. Table IV also shows the mean and standard deviation of the fitting errors across these 9 and 70 landmarks respectively to compare them against the values obtained when all 79 landmarks are considered.

The performance of the classical AAM approach was compared to the ASMs only on the MPIE test set. The ASMs outperformed the AAM and the latter showed a much higher mean fitting error and an inability to accurately fit images with illumination variations not exhibited by its training set. We performed this test in order to demonstrate that ASMs are better suited to the task of facial landmarking than AAMs. In future experiments, we limit tests carried out on the classical AAM to one dataset while the ASMs are tested on three different datasets.

B. Experiment 2 - Study of the Effects of Translation

In our second experiment, we translated the face detected by the Viola-Jones face detector along the $x$ and $y$ axes in steps of 10 pixels between -50 and +50 pixels. Fig. 8(a) illustrates the process followed for a particular combination of $x$ and $y$ translations. Each ASM algorithm was run under these conditions on a subset of images from the MBGC database. For every test image, the Euclidean distance between landmark coordinates obtained by the ASM and those that
were manually annotated (ground truths) was determined and normalized to ensure that it corresponded to an image in which the interocular distance (based on the ground truths) was 50 pixels. These distances were averaged over all images and all 79 landmarks to obtain the mean normalized fitting error NE. Calculating NE for every displacement scenario made it possible to plot a 3D error surface with NE plotted on the x-axis corresponding to displacements of the detected face which were plotted on the x-y plane.

The error surfaces produced by the three ASMs on the MBGC test set are shown in Fig. 9(a) from which it is clear that classical ASM is extremely dependent on accurate initialization and its performance gets worse as the face detected is displaced from its true location. This gives rise to a paraboloid like error surface with the global minimum value at the origin of the x-y plane, corresponding to the case when the face is not displaced. This is expected since it employs only a 1D search and hence can not recover from poor initialization to converge to the right locations for the landmarks. The error surface produced by Stasm also has a paraboloid like appearance but with a flatter base, indicating that for most displacements it is able to recover satisfactorily and produce accurate results. MASM outperforms the other two ASMs as its error surface is almost uniform with low fitting error values, except for large displacements. It also produces a lower value of NE for every displacement scenario than the other two methods. Images showing the fitting results produced by the three ASM algorithms on a few images from the MBGC test set under different initialization conditions are shown in Fig. 10. Fig. 10 also shows how MASM is able to recover from poor initializations and yet converge to the correct locations for most landmarks while Stasm and classical ASM are not as robust and do not fit the landmarks as accurately.

The same experiment was repeated on the entire set of images in the JAFFE database and on a subset of images with frontal faces from session 1 of the MPIE database. Error surfaces that were obtained for the JAFFE and MPIE test sets are shown in Fig. 9(b) and Fig. 9(c) respectively. The values of the fitting errors obtained for the JAFFE and MPIE images are higher than those obtained for the MBGC images as they are quite different from the images used to train the ASMs. However, the overall shape of their error surfaces resembles those obtained using the MBGC test set. Yet again, MASM was able to recover quite well from poor initializations, compared to the other methods. Table V lists the range of x and y translations for each method and each test dataset within which NE remained under 10 pixels. This range is an indicator of how wide the base of the error surface is and clearly shows that MASM exhibits the most tolerance to poor initialization. Beyond this range, all methods performed quite poorly as they could not recover from the excessive translation effects. For all three databases, the range of tolerance along the x and y axis is approximately 1/6th the width of the face for the ASMs. The classical AAM approach was only tested on the MPIE dataset and its error surface is shown in Fig. 9(c). It’s performance is not as robust as the ASMs and it remains accurate for smaller translation errors along the x and y axes, as can be seen from Fig. 9(c) and Table V.

C. Experiment 3 - Study of the Effects of In-Plane Rotation and Translation

The third experiment we carried out was aimed at understanding the role that in-plane rotation along with translation along the direction of rotation can play. For this experiment, the face detected was rotated by an angle θ about its base with θ varied between −30° and +30° in steps of 5° and was then translated along the direction of rotation by r pixels with r varied between 0 and 30 pixels in steps of 5 pixels. The process followed is shown in Fig. 8(b) for a particular value of r and θ. An error surface was obtained by plotting NE along the z-axis and the values of r and θ along the x-axis and y-axis respectively. The error surfaces generated by the ASMs on the same three test sets that were used in experiment 2 are shown in Fig. 11. Fitting results obtained by the ASMs on some images from the MPIE database are shown in Fig. 12.

From Fig. 11 it is apparent that the shape of all error surfaces is the same with the minimum error values lying along the θ = 0° line. A clear trend is also evident as each ASM has an error surface of unique shape which is consistent across the three test sets. MASM and Stasm have error surfaces with wider bases than classical ASM, again indicating their ability to compensate for translation and some amount of rotation of the face detected, however, they too could not deal with clockwise or anti-clockwise rotation upwards of 15°. From Table VI, which shows the region of the error surfaces that correspond to a value of NE lower than 10 pixels for the different methods and test datasets, it is clear that MASM is able to produce lower fitting errors for a larger range of r and θ than Stasm and classical ASM and is hence more robust to the rotation and translation effects that we introduced.
Fig. 12. Fitting results obtained by the three ASMs after carrying out experiment 3 on a few images from the MPIE database (a) Classical AAM (b) Classical ASM (c) Stasm (d) MASM. In (a), (b), (c), and (d) the green dots and red lines indicate the final fitting results, the blue squares correspond to the true faces detected, and the green squares represent the rotated and displaced faces used to initialize the ASMs with the rotations ($\theta$) and translations ($r$) corresponding to $\theta = -15^\circ$, $0^\circ$, and $+15^\circ$ from top to bottom (along the rows of (a), (b), and (c)) and $r = 0$, $15$, and $30$ pixels from left to right (along the columns of (a), (b), and (c)).

Fig. 13. Error plots produced by the different landmarking methods when experiment 4 was conducted on the (a) MBGC test set (b) JAFFE test set (c) MPIE test set.

Fig. 11 and Fig. 12 both show that the three ASMs are more sensitive to rotation than to translation as they are able to obtain higher fitting accuracy when only translation is present than when rotation is present and translation is absent. This
is because the faces used to train the methods were mostly upright and thus when dealing with an unseen face exhibiting a large degree of in-plane rotation, the local profiles around landmarks were not accurately modeled leading to inaccurate fitting. The error surface of the classical AAM approach on the MPIE test set is shown in Fig. 11(c) while sample fitting results produced by it are shown in Fig. 12(a). Both these figures demonstrate the low tolerance of the AAM to rotation and translation effects in initialization. This is also corroborated by the fact that it shows a much smaller area, in Table VI, for which NE remains under 10 pixels than Stasm or MASM.

D. Experiment 4 - Study of Scaling Effects

Our fourth and final experiment addressed the effects of scaling the size of the detected face on ASM performance by observing the fitting errors that were obtained when a scaling parameter $s$ was varied between 0.5 and 1.5 in steps of 0.05. The process is illustrated in Fig. 8(c). We again carried out our tests on the same datasets that were used in experiments 2 and 3. Since we varied only one parameter in this experiment, we obtained a 2D plot of NE vs $s$. The error plots obtained for the three test datasets are shown in Fig. 13. Sample fitting results obtained by the three ASMs on a few images from the MPIE database for different values of $s$ are shown in Fig. 14.

The conclusion we can draw from this experiment is that since all three ASMs rely on face detection results to estimate the size of a face in image, they are extremely susceptible to scaling effects. They all show a minimum fitting error when $s$ is equal to 1 or close to it, corresponding to error-free initialization, but degrade in performance on either side of this value. However, the error curve is not symmetric about $s = 1$, showing that the ASMs are able to recover better when the face is scaled up rather than when it is scaled down. This is explained by the fact that profiles constructed for landmarks on a bigger face can still allow an ASM to differentiate between different regions of the face unlike profiles built for a shrunken face, where all the landmarks are relatively close to each other and hence differentiating regions of the face is more difficult.

The error curves in Fig. 13 show that both Stasm and MASM perform similarly across all databases with MASM showing marginally lower fitting errors for some points along the error curves, especially on the MPIE test set. Both have error curves that flatten out as $s$ approaches 1 and remain that way even when $s$ exceeds 1, thus proving their resilience to oversizing of the face. Classical ASM again suffers because of its limited 1D search space and hence only provides accurate
results when the scaling factor is close to 1. For values of $s$ between 0.5 and 0.7, all three methods perform almost equally poorly and thus their error curves overlap. Classical AAM was once again tested only on the MPIE database and as can be seen from its error curve (shown in Fig. 13(b)) and sample fitting results (shown in Fig. 14(a)), it does not handle scaling effects as robustly as the ASMs.

V. Conclusions

Initialization plays an extremely important role in ensuring accurate automatic facial landmarking. Though some prior work has been carried out in order to observe the effects of initialization, to the best of our knowledge, ours is the first large scale quantitative study of these effects on several popularly used facial landmarking algorithms across multiple databases. Our experiments aid in understanding the impact of initialization on the performance of three different Active Shape Models, namely classical ASM, Stasm and MASM, when used for this application under identical training and testing conditions.

Our first experiment compared the fitting accuracies of the three ASMs when initialization errors were absent. Subsequently, we performed experiments to determine how these methods fared under poor initialization conditions. The second experiment we carried out showed how the methods were impacted if the face detection was erroneous in location and translated along the $x$ and $y$ axes. Using this experiment, we established that a robust ASM method should have as flat an error surface as possible to be robust to translation effects. Our third experiment rotated the face detected through varying angles and then translated it along the direction of rotation. The error surfaces obtained for the different ASM algorithms all had a similar shape. The area of the base of the error surface is again a measure of how sensitive an ASM is to these rotation and translation effects. It was also observed that rotation affects fitting accuracy more than translation. Our final experiment measured robustness to scaling of the face detected and established that oversizing leads to less performance degradation than shrinking of the face detection window. During all four experiments, we also benchmarked the performance of three ASMs against that of the classical AAM approach and demonstrated why ASMs are more applicable to the task of automatic facial landmarking.

The susceptibility of a facial landmarking technique to initialization effects should be a parameter for gauging performance as face detection is prone to translation, rotation and scaling errors and our experiments help in visualizing the effects of such initialization errors. Our results held across three test datasets and showed that our MASM approach was the most robust, among the algorithms we evaluated, to the initialization effects that can creep in during face detection. Since MASM is able to cope with large initialization variations, it could allow us to employ simpler and more computationally efficient face detectors (for example, based on skin color), which are not as robust as the Viola-Jones face detector to still be able to perform accurate facial landmarking.

ACKNOWLEDGMENTS

This research was supported in part by CyLab at Carnegie Mellon under grants DAAD19-02-1-0389 and W911NF-09-1-0273 from the Army Research Office and by cooperative agreement W911NF-10-2-0028 from the Army Research Laboratory.

REFERENCES


Keshav Seshadri is a student member of the IEEE. He received the B.Tech. degree in Electronics and Communication Engineering from Visvesvaraya National Institute of Technology (VNIT), Nagpur, India in 2007 and the M.S. degree in Electrical and Computer Engineering (ECE) from Carnegie Mellon University in 2008. He is currently a PhD student at Carnegie Mellon’s ECE department. His research interests lie in machine learning and pattern recognition with a focus on automatic facial landmarking and facial recognition.

Marios Savvides is a Research Associate Professor at Carnegie Mellon University in the Electrical and Computer Engineering Department and is the Director of CMU CyLab Biometrics Center. He is one of the four researchers chosen to form the Office of the Director of National Intelligence (ODNI) Center of Academic Excellence in Identity Sciences (CASIS). His research is mainly focused on developing algorithms for robust face and iris recognition as well as in using pattern recognition, machine vision, and computer image understanding for enhancing the performance of real time biometric systems. He is a program committee member on several biometric conferences such as IEEE BTAS, SPIE Biometric Identification, and IEEE AutoID. He has organized and co-chaired the Robust Biometrics Understanding the Science and Technology (ROBUST) 2008 conference. He has authored and co-authored over 120 journal and conference publications, including several book chapters, on the field of biometrics and is an area editor of Springer’s Encyclopedia of Biometrics.