Face Alignment Using View-Based Direct Appearance Models

Shuicheng Yan†, Xinwen Hou‡, Stan Z. Li*, Hongjiang Zhang*, Qiansheng Cheng†
† Department of Info. Sci., School of Math. Sci., Peking University, 100871, China
‡Mathematical School of NanKai University, Tianjin, 300071, China
* Microsoft Research Asia, Beijing Sigma Center, Beijing 100080, China

Abstract—
Accurate face alignment is the prerequisite for many computer vision problems, such as face recognition, synthesis and 3-D face modeling. In this paper, a novel appearance model, called Direct Appearance Model (DAM), is proposed and its extended view-based models are applied for multi-view face alignment. Similar to the active appearance model (AAM), DAM also makes ingenious use of both shape and texture constraints; however, it doesn't combine them as in AAM, texture information is used directly to predict the shape and estimate the position and appearance (hence the name DAM). The way that DAM models shapes and textures has the following advantages as compared with AAM: (1) DAM subspaces include admissible appearances previously unseen in AAM, (2) It can converge more quickly and has higher accuracy, and (3) the memory requirement is cut down to a large extent. Extensive experiments are presented to evaluate the DAM alignment in comparison with AAM.

I. Introduction

The appearance based approaches [Sirovich 87], [Turk 91], [Beymer 93], [Murase 95] avoid the difficulties in 3D modeling by using images of example appearances. It has become a dominant approach in face analysis and many other applications. The appearance of a face in an image is initially represented as a patch of image intensities enclosed by the facial outline (namely, shape). In this paper, the intensity patch contained in the shape after warping to the mean shape [Cootes 01] is called *texture*. Both shape and texture provide important clues useful for characterizing the face appearance [Beymer 95]. The task of face alignment is to accurately locate the representative points of the facial outline and extract the confident face texture, which is crucial for high accuracy face recognition, synthesis [Moghaddam 97], [Torre 98], [Edwards 98], [Blanz 99], [Gong 00], and tracking [Sclaroff 98], [Hager 98], [Cascia 00], [Ahlberg 01].

The Active Appearance Model (AAM), proposed by Cootes et. al. [Cootes 01], is a powerful tool for face alignment, recognition [Edwards 98] , synthesis [Blanz 99] and widely used in medical imaging processing. It makes ingenious use of subspace analysis techniques, PCA in particular, to model both shape variation and texture variation, and the correlations between them. Another feature is that AAM uses a smart search strategy: It assumes linear relationships between appearance variation and texture variation and between texture variation and position variation; and learns the two linear regression models from training data. The two models facilitate the minimizations in high dimensional space. This strategy is also developed in the active blob model of Sclaroff and Isidoro [Sclaroff 98]. The AAM has been extended to multi-view faces by using piecewise linear modeling [Cootes 00].

More recent progress has been made in this area. [Li 01] present a method for learning 3D face shape model from 2D images based on a shape-and-pose-free texture model. Cootes and Taylor show that imposing extra constraints such as fixing eye locations can improve AAM search result. [Baker 01] propose an efficient algorithm (inverse compositional algorithm) for alignment. [Hou 01] present a novel formulation of the relationship between texture and shape by using texture to directory predict shape. [Yan 02] propose a strategy to combine local texture and global texture for shape localization.

However, there are still two problems in the conventional AAM. 1) Our analysis on mutual dependencies of shape, texture and appearance parameters in the AAM subspace models shows that there exist admissible appearances that are not modeled and hence cannot be reached by AAM search processing. 2) Another problem with AAM is that the training of the two prediction models is based on texture difference vectors and is therefore very memory consuming because the training data for the two models are generated in a rapidly multiplicative way. The memory explosion makes AAM training very difficult even with a moderate number of images.

We proposed a new appearance model, called Direct Appearance Model (DAM) [Li 02], to give a solution to above two problems in multi-view face alignment. To solve the first problem, the DAM model provides a proper subspace modeling approach based on our findings: the mapping from the texture subspace to the shape subspace is many-to-one and therefore a shape can be determined entirely by the texture enclosed by itself. From these relationships, the DAM model considers an appearance, which is composed of both shape and texture, to be determinable by using just the corresponding texture. DAM uses the texture information *directly* to predict the shape and to update the estimates of position and appearance (hence the name DAM); in contrast to AAM's crucial idea of modeling the AAM appearance subspace from combining shape and texture. Thus, DAM includes the admissible appearances

The work presented in the paper was carried out at Microsoft Research Asia.

previously unseen by AAM, and improves the convergence rate and accuracy.

To avoid the second problem, DAM predicts the new face position and appearance based on the principal components of texture difference vectors, instead of the raw vectors themselves as in AAM. This cuts down the memory requirement to a large extent, and further improves the convergence rate and accuracy. The claimed advantages of DAM are substantiated by comparative experimental results.

In multi-view face alignment, the whole range of views from frontal to side views are partitioned into several sub-ranges, and one DAM model is trained to represent the shape and texture for each sub-range. Which view DAM model to use may be decided by using some pose estimate for static images. In the case of face alignment from video, the previous view plus the two neighboring view DAM models may be attempted, and then the final result is chosen to be the one with the minimum texture residual error.

The rest of the paper is organized as follows: In Section 2, we analyze the AAM model and point out its shortcomings after a brief introduction of AAM. Then we propose the DAM model and search algorithm. In Section 3, DAM is used for multi-view face alignment. This is followed by extensive experimental results in Section 4. In Section 5, we conclude this paper.

II. Direct Appearance Model

Assume that a training set be given as $\mathbf{A} = \{(S_i, T_i^o)\}$ where a shape $S_i = ((x_1^i, y_1^i), \ldots, (x_K^i, y_K^i)) \in \mathbb{R}^{2K}$ is a sequence of K points in the 2D image plane, and a texture T_i^o is the patch of image pixels enclosed by S_i. Let $\overline{S}$ be the mean shape of all the training shapes. $\overline{S}$ is calculated after the shapes are aligned to the tangent space of the mean shape $\overline{S}$, which can be implemented as an iterative procedure [Cootes 98]. After the shape warping, the texture T_i^o is warped correspondingly to $T_i \in \mathbb{R}^L$, where L is the number of pixels in the mean shape $\overline{S}$, by pixel value interpolation *e.g.* using a triangulation or thin plate spline method.

A. Introduction to AAM

In the conventional AAM, the subspace analysis techniques are efficiently used for modeling the variable variations and correlations. The shape is modeled by k ($< 2K$) principal modes learned from the training shapes using PCA. By this, a shape, which is originally in $\mathbb{R}^{2K}$, is represented as a point or vector s in the low dimensional shape subspace in $\mathbb{R}^k$

$$S = \overline{S} + \mathbf{U}s \tag{1}$$

where $\mathbf{U}$ is the matrix consisting of k principal orthogonal modes of variation in $\{S_i\}$. Because the training shapes have been aligned to the tangent space of $\overline{S}$, the eigenvectors in $\mathbf{U}$ is orthogonal to the mean shape $\overline{S}$, i.e. $\mathbf{U}^T\overline{S} = 0$, and the projection from S to s is

$$s = \mathbf{U}^T(S - \overline{S}) = \mathbf{U}^T S \tag{2}$$

The above defines AAM's shape subspace $\mathbb{S}_s$.

After deforming each training shape S_i to the mean shape, the corresponding texture T_i^o is warped to T_i. All the warped textures are aligned to the tangent space of the mean texture $\overline{T}$ by using an iterative approach as described in [Cootes 98]. The PCA texture model is obtained as

$$T = \overline{T} + \mathbf{V}t \tag{3}$$

where $\mathbf{V}$ is the matrix consisting of ℓ principal orthogonal modes of variation in $\{T_i\}$, t is the vector of texture parameters. The projection from T to t is

$$t = \mathbf{V}^T(T - \overline{T}) = \mathbf{V}^T T \tag{4}$$

By this, the L pixel values in the mean shape is represented as a point in the ℓ dimensional texture subspace $\mathbb{S}_t$.

Since there may be correlations between the shape and texture variations, a further appearance model is built from $\{s\}$ and $\{t\}$. The appearance of each example is a concatenated vector

$$A = \begin{pmatrix} \mathbf{\Lambda} s \\ t \end{pmatrix} \tag{5}$$

where $\mathbf{\Lambda}$ is a diagonal matrix of weights for the shape parameters allowing for the difference in units between the shape and texture variation. One may simply set $\mathbf{\Lambda} = r\mathbf{I}$ where r^2 is the ratio of the total intensity variation to the total shape variation. Again, by applying PCA on the set $\{A\}$, one gets

$$A = \mathbf{W}a \tag{6}$$

where $\mathbf{W}$ is the matrix consisting of principal orthogonal modes of variation in $\{A\}$. By projecting from A to a, AAM models its appearance subspace $\mathbb{S}_a$ by

$$a = \mathbf{W}^T A \tag{7}$$

Consider the difference between the texture T_{im} in the image patch and the texture T_a reconstructed from the current appearance parameters

$$\delta T = T_{im} - T_a \tag{8}$$

In AAM, the search for a face in an image is guided by minimizing the norm $||\delta T||$. The AAM assumes that the appearance displacement δa and the position (including translations (x, y), scale s and rotation parameter θ) displacement δp are linearly correlated to δT. It predicts the displacements as

$$\delta a = \mathbf{A}_a \delta T \tag{9}$$

$$\delta p = \mathbf{A}_p \delta T \tag{10}$$

where the prediction matrices $\mathbf{A}_a, \mathbf{A}_p$ are to be learned from the training data by using linear regression. In order to estimate $\mathbf{A}_a$, we need to systematically displace a to get δa and the induced δT for each training image.

B. Motivations for DAM

The conventional AAM is widely applied in different fields. However, the following analysis of relationships between the shape, texture and appearance subspaces in AAM shows defects of the AAM models. Thereby we suggest a property that an ideal appearance model should have, which motivates us to propose the DAM.

First, let us look into relationship between shape and texture from an intuitive viewpoint. A texture (*i.e.* the patch of intensities) is enclosed by a shape (before aligning to the mean shape); the same shape can enclose different textures (*i.e.* configurations of pixel values). However, the reverse is not true: different shapes can not enclose the same texture. So the mapping from the texture space to the shape space is many-to-one. The shape parameters should be determined completely by texture parameters but not vice versa.

Then, let us look further into the correlations or constraints between the linear subspaces $\mathbb{S}_s$, $\mathbb{S}_t$ and $\mathbb{S}_a$ in terms of their dimensionalities or ranks. Let denote the rank of space $\mathbb{S}$ by $\dim(\mathbb{S})$. We have the following analysis:

1. When $\dim(\mathbb{S}_a)=\dim(\mathbb{S}_t)+\dim(\mathbb{S}_s)$, the shape and texture parameters are independent of each other, and there exist no mutual constraints between the parameters s and t.

2. When $\dim(\mathbb{S}_t)<\dim(\mathbb{S}_a)<\dim(\mathbb{S}_t)+\dim(\mathbb{S}_s)$, not all the shape parameters are independent of the texture parameters. That is, one shape can correspond to more than one texture configuration in it, which conforms our intuition.

3. One can also derive the relationship $\dim(\mathbb{S}_t)<\dim(\mathbb{S}_a)$ from Eq.(5) and (6) the formula

$$\mathbf{W}a = \begin{pmatrix} \mathbf{\Lambda}s \\ t \end{pmatrix} \tag{11}$$

when that s contains some components which are independent of t.

4. However, in AAM, it is often the case where $\dim(\mathbb{S}_a)<\dim(\mathbb{S}_t)$ if the dimensionalities of $\mathbb{S}_a$ and $\mathbb{S}_t$ are chosen to retain, say 98%, of the total variations, which is reported by Cootes [Cootes 98] and also observed by us. The consequence is that some admissible texture configurations cannot been seen in the appearance subspace because $\dim(\mathbb{S}_a)<\dim(\mathbb{S}_t)$, and therefore cannot be reached by the AAM search. We consider this a flaw of AAM's modeling of its appearance subspace.

From the above analysis, we conclude that the ideal model should be such that $\dim(\mathbb{S}_a)=\dim(\mathbb{S}_t)$ and hence that s completely linearly determinable by t. In other words, the shape should be linearly dependent on the texture so that $\dim(\mathbb{S}_t \cup \mathbb{S}_s)=\dim(\mathbb{S}_t)$. The DAM model is proposed mainly for this purpose.

Another motivation of DAM is the space consumption: the regression of $\mathbf{A}_a$ in AAM is very memory consuming. AAM prediction needs to model linear relationship between appearance and texture difference according to (9). However, both a and δT are high dimensional vectors, and therefore the storage size of training data generated for learning (9) increases very rapidly as the dimensions increase. It is very difficult to train AAM for $\mathbf{A}_a$ even with a moderate number of images. Learning in a low dimensional space will relieve the burden.

C. DAM Modeling and Training

Our proposed DAM proves a solution to the problems in AAM. It consists of a shape model, two texture (original and residual) model and two prediction (position and shape prediction) model. The shape, texture models and the position prediction model (10) are built in the same way as in AAM. The residual texture model is built using the subspace analysis technique PCA. Abandoning AAM's crucial idea of combining shape and texture parameters into an appearance model, it predicts the shape parameters directly from the texture parameters. In the following, the last two models are demonstrated in detail.

Recall the conclusions we made earlier: (1) an ideal appearance model should have $\dim(\mathbb{S}_a)=\dim(\mathbb{S}_t)$ and (2) shape should be computable uniquely from texture but not vice versa. Therefore we propose the following regression model by assuming a linear relationship between shape and texture

$$s = \mathbf{R}t + \varepsilon \tag{12}$$

where $\varepsilon = s - \mathbf{R}t$ is noise and $\mathbf{R}$ is a $k \times l$ projection matrix. Denoting the expectation by $E(\cdot)$, if all the elements in the variance matrix $E(\varepsilon\varepsilon^T)$ are small enough, the linear assumption made in Eq.(12) is approximately correct. This is true as will be verified later by experiments. Define the objective cost function

$$C(\mathbf{R}) = E(\varepsilon^T\varepsilon) = \mathrm{trace}[E(\varepsilon\varepsilon^T)] \tag{13}$$

$\mathbf{R}$ is learned from training example pairs $\{(s,t)\}$ by minimizing the above cost function. The the optimal solution is

$$\mathbf{R}^* = E(st^T)[E(tt^T)]^{-1} \tag{14}$$

The minimized cost is the trace of the following

$$E(\varepsilon\varepsilon^T) = E(ss^T) - \mathbf{R}^*E(tt^T)\mathbf{R}^{*T} \tag{15}$$

Even in the assumption (12), AAM will still miss some admissible texture if only retains 98% of the total variations in (6); As all the example shape and texture are modeled in (12), the admissible appearance can be seen in the subspace modeled by DAM.

Another motivation of DAM is the huge memory consumption for the modeling of the regression matrix $\mathbf{A}_a$ in AAM. Instead of using δT directly as in the AAM search (*cf.* Eq.(10), we use principal components of it, $\delta T'$, to predict the position displacement

$$\delta p = \mathbf{R}_p \delta T' \tag{16}$$

where $\mathbf{R}_p$ is the prediction matrix learned by using linear regression. To do this, we collect texture differences induced by small position displacements in each training image, and perform PCA on this data to get the projection matrix $\mathbf{H}^T$. A texture difference is projected onto this subspace as

$$\delta T' = \mathbf{H}^T \delta T \tag{17}$$

$\delta T'$ is normally about 1/4 of δT in dimensionality. Results have shown that the use of δT instead of $\delta T'$ as in Eq.(16) makes the prediction more stable and more accurate.

The DAM learning consists of two parts: (1) learning $\mathbf{R}$, and (2) learning $\mathbf{H}$ and $\mathbf{R}_p$: (1) $\mathbf{R}$ is learned from the shape-texture pairs $\{s, t\}$ obtained from the landmarked images. (2) To learn $\mathbf{H}$ and $\mathbf{R}_p$, we generate artificial training data by perturbing the position parameters p around the landmark points to obtain $\{\delta p, \delta T\}$; then learn $\mathbf{H}$ from $\{\delta T\}$ using PCA; after that we compute $\delta T'$; and finally derive $\mathbf{R}_p$ from $\{\delta p, \delta T'\}$.

The DAM regression in Eq.(16) requires much less memory than the AAM regression in Eq.(9), typically DAM needs only about 1/20 of memory required by AAM. For DAM, there are 200 training images, 4 parameters for the position: $(x, y, \theta, scale)$, and 6 disturbances for each parameter to generate training data for the training $\mathbf{R}_p$. So, the size of training data for DAM is $200 \times 4 \times 6 = 4,800$. For AAM, there are 200 training images, 113 appearance parameters, and 4 disturbances for each parameter to generate training data for training $\mathbf{A}_a$. The size of training data for $\mathbf{A}_a$ is $200 \times 113 \times 4 = 90,400$. Therefore, the size of training data for AAM's prediction matrices is $90,400 + 4,800 = 95,200$, which is 19.83 times that for DAM. On a PC, for example, the memory capacity for AAM training with 200 images would allow DAM training with 3,966 images.

Note that there is a variant of basic AAM [Cootes 01], which uses texture difference to predict shape difference. The prediction of shape is done by $\delta s = \mathbf{B}\delta T$. However, this variant is not as good as the basic AAM [Cootes 01].

III. Multi-View DAM

The full range of face poses are divided into 5 view subranges: $[-90°, -55°]$, $[-55°, -15°]$, $[-15°, 15°]$, $[15°, 55°]$, and $[55°, 90°]$ with 0° being the frontal view. The landmarks for frontal, half-side and full-side view faces are illustrated in Fig.1. The dimensions of shape and texture vectors before and after the PCA dimension reductions are shown in Table I where the dimensions after PCA are chosen to be such that 98% of the corresponding total energies are retained. The texture appearances due to respective variations in the first three principal components of texture are demonstrated in Fig.2.

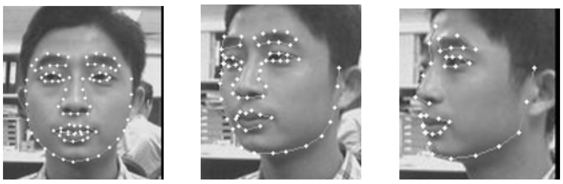


Fig. 1. Frontal, half-side, and full-side view faces and the labeled landmark points.

View	#1	#2	#3	#4	#5
Fontal	87	69	3185	144	878
Half-Side	65	42	3155	144	1108
Full-Side	38	38	2589	109	266

TABLE I

Dimensionalities of shape and texture variations for face data. #1 Number of landmark points. #2 Dimension of shape space $\mathbb{S}_s$. #3 Number of pixel points in the mean shape. #4 Dimension of texture space $\mathbb{S}_t$. #5 Dimension of texture variation space ($\delta T'$).

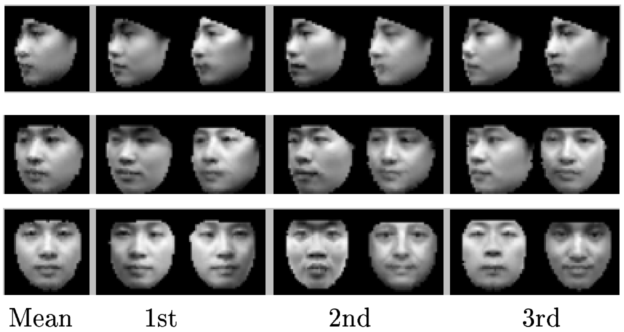


Fig. 2. Texture and shape variations due to variations in the first three principal components of the texture (The shapes change in accordance with $s = \mathbf{R}t$) for full-side ($\pm 1\sigma$), half-side ($\pm 2\sigma$), and frontal ($\pm 3\sigma$) views. .

The left side models and right side models are reflections of each other, so we only need to train one side of them. So we train $[-15°, 15°]$, $[15°, 55°]$, and $[55°, 90°]$ for the 5 models. We can find the corresponding model for all the face with view in $[-90°, 90°]$.

The novel DAM prediction models leads to the following search procedure: The DAM search starts with the mean shape and the texture of the input image enclosed by the mean shape, at a given initial position p_0. The texture difference δT is computed from the current shape patch at the current position, and its principal components are used to predict and update p and s using the DAM linear models described above. Note that the p can be computed from δT in one step as $\delta p = \mathbf{R}_T \delta T$, where $\mathbf{R}_T = \mathbf{R}_p \mathbf{H}^T$, instead of two steps as in Eqns.(16) and (17). If $||\delta T||$ calculated using the new appearance at the position is smaller than the old one, the new appearance and position are accepted; otherwise the position is updated by amount $\kappa \delta p$ with varied κ values. The search algorithm is summarized below:

1. Initialize the position parameters p_0, and determine view by which to select the DAM model to use; set shape parameters $s_0 = 0$;
2. Get texture T_{im} from the current position, project it into the texture subspace $\mathbb{S}_t$ as t, reconstruct the texture T_a, and compute texture difference $\delta T_0 = T_{im} - T_a$ and the energy $E_0 = ||\delta T_0||^2$;
3. Compute $\delta T' = \mathbf{H}^T \delta T$, get the position displacement $\delta p = \mathbf{R}_p \delta T'$;

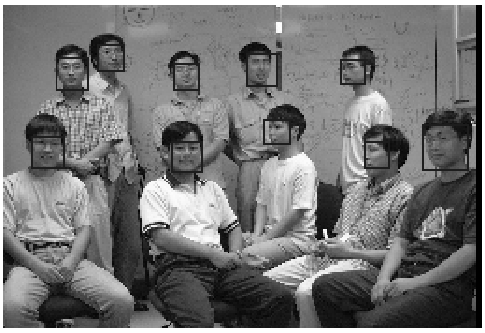


Fig. 3. Initial alignment provided by a multi-view face detector.

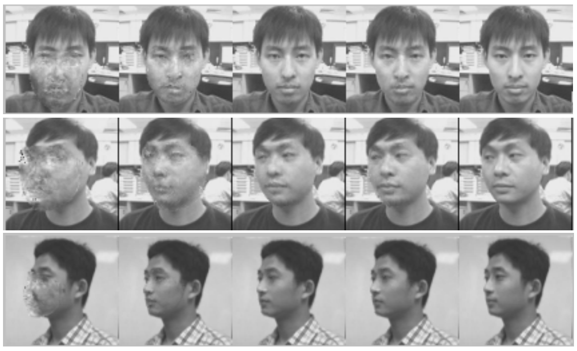
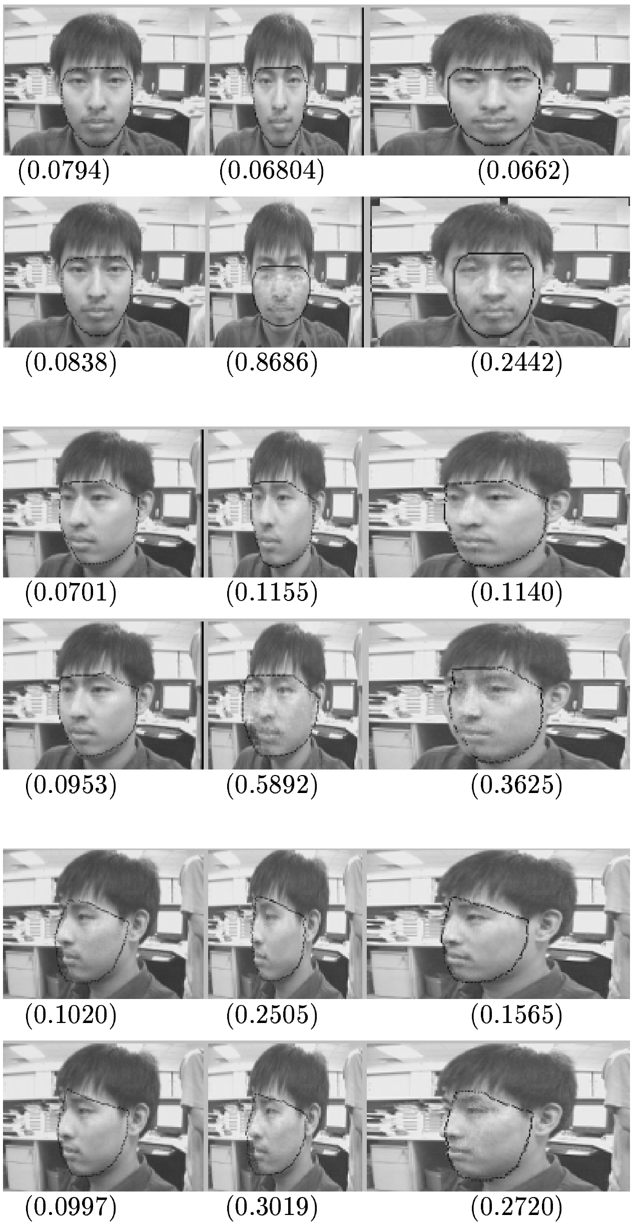


Fig. 4. DAM aligned faces (from left to right) at the 0-th, 5-th, 10-th, and 15-th iterations, and the original images for (top-bottom) frontal, half-side and full-side view faces.

```
4. Set step size $\kappa = 1$;
5. Update $p = p_0 - \kappa\delta p$, $s = \mathbf{R}t$;
6. Compute the difference texture $\delta T$ using the new shape at the new position, and its energy $E = ||\delta T||^2$;
7. If $|E - E_0| < \epsilon$, the algorithm is converged; exit;
8. If $E < E_0$, then let $p_0 = p, s_0 = s, \delta T_0 = \delta T, E_0 = E$, goto 3;
9. Change $\kappa$ to the next number in $\{1.5, 0.5, 0.25, 0.125, \ldots, \}$, goto 5;
```

In our implementation, the initialization and pose estimation are performed automatically by using a robust real-time multi-view face detector we have developed recently, as shown in Fig.3. A multi-resolution pyramid structure is used in search to improve the result. Fig.4 demonstrates scenarios of how DAM converges.

When the face is undergone large variation due to stretch in either the x or y direction, the model fitting can be improved by allowing different scales in the two directions. This is done by splitting the scale parameter into two: s_x and s_y. The improvement is demonstrated in Figs.5.



(0.0794) (0.06804) (0.0662)

(0.0838) (0.8686) (0.2442)

(0.0701) (0.1155) (0.1140)

(0.0953) (0.5892) (0.3625)

(0.1020) (0.2505) (0.1565)

(0.0997) (0.3019) (0.2720)

Fig. 5. Results of non-isometric (top of each of the three blocks) and isometric (bottom) search for frontal (top block), half-side (middle block) and full-side (bottom block) view faces. From left to right of each row are normal, and stretched faces. The number below each result is the corresponding residual error.

IV. Experimental Results

The training set contains 200 frontal, 200 half-side, and 170 full-side view faces whose sizes are of about 64x64 pixels, while the test set contains 80 images for each view group. The landmark points are labeled manually (see Fig.1 and Table I). They are used for the training and as ground-truth in the test stage.

To compare, we also implemented AAM using the same data in the frontal view. The shape and texture parameter vectors are 69+144 dimensional, respectively, where the weight parameter for the concatenation of the two parts is calculated as $r = 8.84$ for $\mathbf{\Lambda} = r\mathbf{I}$ in Eq.(5). The concatenated vector space is reduced to a 113 dimensional appearance subspace which retains 98% of the total variation of the concatenated features.

For DAM, the linearity assumption made for the model of Eq.(12) is well verified because all the elements in $E(\varepsilon\varepsilon^T)$ calculated over the training set are smaller than 10^{-4}.

Some results about DAM learning and search have been presented in Figs.2-5. Fig.6 compares the convergence rate and accuracy properties of DAM and AAM (for the frontal view) in terms of the error in δT (*cf.* Eq.(8)) as the algo-

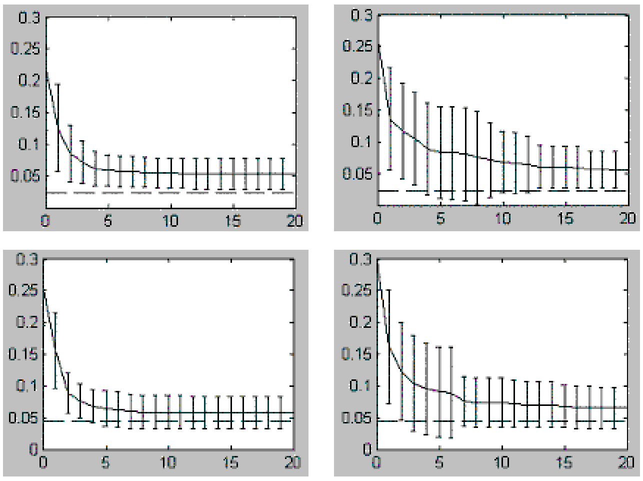


Fig. 6. Mean error (the curve) and standard deviation (the bars) in reconstructed texture $||\delta T||$ as a function of iteration number for the DAM (left) and AAM (right) methods with the training (top) and test (bottom) sets, for frontal face images. The horizontal dashed lines in the lower part of the figures indicate the average $||\delta T||$ for the manually labeled alignment.

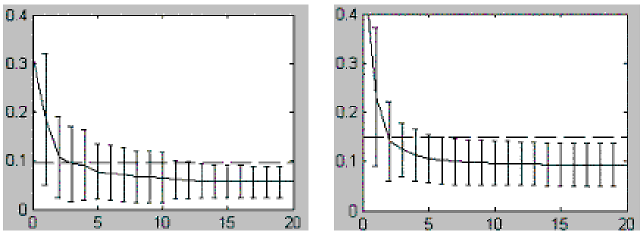


Fig. 7. Mean error in $||\delta T||$ and standard deviation of the DAM alignment for half- (left) and full- (right) side view face images from the test set. Note that the mean errors in the calculated solutions are smaller than obtained using the manually labeled alignment after a few iterations.

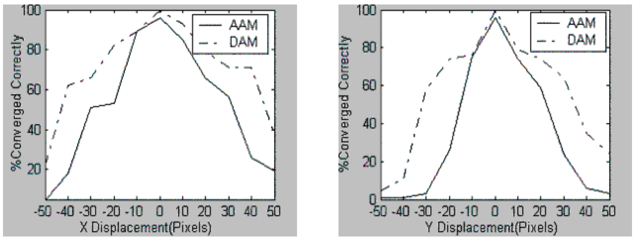


Fig. 8. Alignment accuracy of DAM (dashed) and AAM (solid) in terms of localization errors in the x (left) and y (right) directions.

rithms iterate. The statistics are calculated from 80 images randomly selected from the training set and the 80 images from test set. We see that DAM has faster convergence rate and smaller error than AAM. Fig.7 illustrates the error of DAM for non-frontal faces. Fig.8 compares the alignment accuracy of DAM and AAM (for frontal faces) in terms of the percentage of images whose texture reconstruction error δT is smaller than 0.2, where the statistics are obtained using another test set including the 80 test images mentioned above and additional 20 other test images. It shows again that DAM is more accurate than AAM.

The DAM search is fairly fast. It takes on average 39 ms per iteration for frontal and half-side view faces, and 24 ms for full-side view faces in an image of size 320x240 pixels. Every view model takes about 10 iterations to converge. If 3 view models are searched with per face, as is done with image sequences from video, the algorithm takes about 1 second to find the best face alignment.

V. Conclusion

In this paper, we have presented a method for multi-view face alignment based on our proposed Direct Appearance Models (DAM). DAM overcomes certain limitations of AAM in the subspace modeling. Unlike AAM, all admissible appearances can be seen in the subspaces modeled by DAM and thus reachable in DAM search. The DAM has faster convergence and solution accuracy. Also, DAM requires less memory than AAM and allows to learn prediction matrices from a large number of training images. The occlusions, facial expressions, and illumination are still hard conditions for accurate face alignment, we are planing to develop robust algorithm to deal with these conditions.

References

J. Ahlberg. "Using the active appearance algorithm for face and facial feature tracking". In *IEEE ICCV Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-time Systems*, pages 68–72, Vancouver, Canada, July 13 2001.

David Beymer. "Vectorizing face images by interleaving shape and texture computations". A.I.Memo 1537,MIT,1995.

S. Baker and I. Matthews. "Equivalence and efficiency of image alignment algorithms". In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Hawaii, December 11-13 2001.

David Beymer, Amnon Shashua, and Tomaso Poggio. "Example based image analysis and synthesis". A.I.Memo 1431,MIT,1993.

V. Blanz and T.Vetter. "A morphable model for the synthesis of 3d faces". In *SIGGRAPH'99 Conference Proceedings*, pages 187–194, 1999.

T. Cootes and C.Taylor. Statistical models of appearance for computer vision. *"http://www.isbe.man.ac.uk/ bim/refs.html"*,2001.

T. Cootes, G.Edwards, and C.Taylor. Active appearance models. In *ECCV98*, volume 2, pages 484–498.

T. Cootes, G.Wheeler, K.Walker, and C.Taylor. Coupled-view active appearance models. In *Proc. British Machine Vision Conference*, volume 1, pages 52–61, 2000.

T. Cootes, K.Walker, and C.Taylor. View-based active appearance models. In *Proc. Int. Conf. on Face and Gesture Recognition*, pages 227–232, 2000.

M. La Cascia, S. Sclaroff, and V. Athitsos. "Fast, reliable head tracking under varying illumination: An approach based on robust registration of texture-mapped 3d models". 22(4):569–579, 2000.

F. de la Torre, S. Gong, and S. McKenna. "View alignment with dynamically updated affine tracking". In *Proc. IEEE International Conference on Automatic Face and Gesture Recognition*, Nara, Japan, April 14-16 1998.

G. Edwards, T. Cootes, and C.Taylor. "Face recognition using active appearance models". In *Proceedings of the European Conference on Computer Vision*, volume 2, pages 581–695, 1998.

S. Gong, S. McKenna, and A. Psarrou. *Dynamic Vision: From Images to Face Recognition.* World Scientific Publishing and Imperial College Press, April 2000.

Gregory D. Hager and Peter N. Belhumeur. "Efficient region tracking with parametric models of geometry and illumination". IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(10):1025–1039, October 1998.

X.W. Hou, S.Z. Li, H.J. Zhang, Q.S. Cheng. Direct Appearance Models. in Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition. Hawaii. December, 2001.

Y. Li, S. Gong, and H. Liddell. Constructing facial identity surfaces in a nonlinear discriminating space. *int Proc. of the Nineteenth IEEE Computer Society Conference on Computer Vision* *and Pattern Recognition (CVPR2001), Kauai Marriott, Hawaii*, December 2001.

S.Z. Li, S.C. Yan, H.J. Zhang, Q.S. Cheng. Multi-View Face Alignment Using Direct Appearance Models. In Proceedings of the 5th International Conference on Automatic Face and Gesture Recognition. Washington, DC, USA. 20-21 May, 2002.

H. Murase and S. K. Nayar. “Visual learning and recognition of 3-D objects from appearance”. *International Journal of Computer Vision*, 14:5–24, 1995.

B. Moghaddam and A. Pentland. “Probabilistic visual learning for object representation”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7:696–710, July 1997.

S. Sclaroff and J. Isidoro. “Active blobs”. In *Proceedings of IEEE International Conference on Computer Vision*, Bombay, India, 1998.

L. Sirovich and M. Kirby. “Low-dimensional procedure for the characterization of human faces”. *Journal of the Optical Society of America A*, 4(3):519–524, March 1987.

Matthew A. Turk and Alex P. Pentland. “Face recognition using eigenfaces.”. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 586–591, Hawaii, June 1991.

S.C. Yan, C. Liu, S.Z. Li, L. Zhu, Z, H.J. Zhang, H. Shum, Q.S. Cheng. Texture-Constrained Active Shape Models. in Proceedings of the First International Workshop on Generative-Model-Based Vision. Copenhagen, Denmark. May, 2002.