3D Face Model Fitting for Recognition

Frank B. ter Haar and Remco C. Veltkamp

Department of Information and Computing Sciences, Utrecht University, the Netherlands

Abstract. This paper presents an automatic efficient method to fit a statistical deformation model of the human face to 3D scan data. In a global to local fitting scheme, the shape parameters of this model are optimized such that the produced instance of the model accurately fits the 3D scan data of the input face. To increase the expressiveness of the model and to produce a tighter fit of the model, our method fits a set of predefined face components and blends these components afterwards. Quantitative evaluation shows an improvement of the fitting results when multiple components are used instead of one. Compared to existing methods, our fully automatic method achieves a higher accuracy of the fitting results. The accurately generated face instances are manifold meshes without noise and holes, and can be effectively used for 3D face recognition: We achieve 97.5% correct identification for 876 queries in the UND face set with 3D faces. Our results show that contour curve based face matching outperforms landmark based face matching.

1 Introduction

The use of 3D scan data for face recognition purposes has become a popular research area. With high recognition rates reported for several large sets of 3D face scans, the 3D shape information of the face proved to be a useful contribution to person identification. The major advantage of 3D scan data over 2D color data, is that variations in scaling and illumination have less influence on the appearance of the acquired face data. However, scan data suffers from noise and missing data due to self-occlusion. To deal with these problems, 3D face recognition methods should be invariant to noise and missing data, or the noise has to be removed and the holes interpolated. Alternatively, data could be captured from multiple sides, but this requires complex data acquisition. In this work we propose a method that produces an accurate fit of a statistical 3D shape model of the face to the scan data. The 3D geometry of the generated face instances, which are without noise and holes, are effectively used for 3D face recognition.

Related work. The task to recognize 3D faces has been approached with many different techniques as described in surveys of Bowyer et al. [1] and Scheenstra et al. [2]. Several of these 3D face recognition techniques are based on 3D geodesic surface information, such as the methods of Bronstein et al. [3] and Berretti et al. [4]. The geodesic distance between two points on a surface is the length of the shortest path between two points. To compute accurate 3D geodesic distances for face recognition purposes, a 3D face without noise and without holes is desired. Since this is typically not the case with laser range scans, the noise has to be removed and the holes in the 3D surface interpolated. However, the success of basic noise removal techniques, such as Laplacian smoothing is

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part IV, LNCS 5305, pp. 652–664, 2008.

very much dependent on the resolution of the scan data. Straightforward techniques to interpolate holes using curvature information or flat triangles often fail in case of complex holes, as pointed out in [5]. The use of a deformation model to approximate new scan data and interpolate missing data is a gentle way to regulate flaws in scan data.

A well known *statistical deformation model* specifically designed for surface meshes of 3D faces, is the 3D morphable face model of Blanz and Vetter [6]. This statistical model was built from 3D face scans with dense correspondences to which Principal Component Analysis (PCA) was applied. In their early work, Blanz and Vetter [6] fit this 3D morphable face model to 2D color images and cylindrical depth images from the $Cyberware^{TM}$ scanner. In each iteration of their fitting procedure, the model parameters are adjusted to obtain a new 3D face instance, which is projected to 2D cylindrical image space allowing the comparison of its color values (or depth values) to the input image. The parameters are optimized using a stochastic Newton algorithm. More recently, Blanz et al. [7] proposed a method to fit their 3D morphable face model to more common textured depth images. The fitting process is similar to their previous algorithm, but now the cost function is minimized using both color and depth values after the projection of the 3D model to 2D cylindrical image space. To initialize their fitting process, they manually select seven corresponding face features on their model and in the depth scan. A morphable model of expressions was proposed by Lu et al. [8]. Starting from an existing neutral scan, they use their expression model to adjust the vertices in a small region around the nose to obtain a better fit of the neutral scan to a scan with a certain expression.

Non-statistical deformation models were proposed as well. Huang et al. [9] proposed a global to local deformation framework to deform a shape with an arbitrary dimension (2D, 3D or higher) to a new shape of the same class. They show their framework's applicability to 3D faces, for which they deform an incomplete source face to a target face. Kakadiaris et al. [10] deform an annotated face model to scan data. Their deformation is driven by triangles of the scan data attracting the vertices of the model. The deformation is restrained by a stiffness, mass and damping matrix, which control the resistance, velocity and acceleration of the model's vertices. The advantage of such deformable faces is that they are not limited to the statistical changes of the input shapes, so the deformation has less restrictions. However, this is also their disadvantage, because these models cannot rely on statistics in case of noise and missing data.

Contribution. First, we propose a fully automatic algorithm to efficiently optimize the parameters of the morphable face model, creating a new face instance that accurately fits the 3D geometry of the scan data. Unlike other methods, ours needs no manual initialization, so that batch processing of large data sets has become feasible. Second, we quantitatively evaluate our fitted face models and show that the use of multiple components improves the fitting process. Thirdly, we show that our model fitting method is more accurate than existing methods. Fourthly, we show that the accurately generated face instances can be effectively used for 3D face recognition.

2 Morphable Face Model

In this work we fit the morphable face model of the USF Human ID 3D Database [11] to 3D scan data to obtain a clean model of the face scan, that we use to identify

3D faces. This statistical point distribution model (PDM) was built from 100 cylindrical 3D face scans with neutral expressions from which n=75,972 correspondences were selected using an optic flow algorithm. Each face shape S_i was described using the set of correspondences $S = (x_1, y_1, z_1, ..., x_n, y_n, z_n)^T \in \Re^{3n}$ and a mean face $\bar{S}$ was determined. PCA was applied to these 100 sets S_i to obtain the m=99 most important eigenvectors of the PDM. The mean face $\bar{S}$, the eigenvectors $s_i = (\Delta x_1, \Delta y_1, \Delta z_1, ..., \Delta x_n, \Delta y_n, \Delta z_n)^T$, the eigenvalues λ_i ($\sigma_i^2 = \lambda_i$) and weights w_i are used to model new face instances according to $S_{inst} = \bar{S} + \sum_{i=1}^{m} w_i \sigma_i s_i$. Weight w_i represents the number of standard deviations a face instance morphs along eigenvector ev_i. Since the connectivity of the n correspondences in the PDM is known, each instance is a triangular mesh with proper topology and without holes.

3 Face Scans

We fit the morphable face model to the 3D frontal face scans of the University of Notre Dame (UND) Biometrics Database [12]. This set contains 953 range scans and a corresponding 2D color texture from 277 different subjects. All except ten scans were used in the Face Recognition Grand Challenge (FRGC v.1). Because the currently used morphable model is based on faces with neutral expressions only, it makes no sense to use collections containing many non-neutral scans such as the FRGC v.2. Nevertheless, our proposed method performs well for the small expression variations of the UND set. Throughout this work, we have only used the 3D scan data and neglected the available 2D color information.

We aim at 3D face recognition, so we need to segment the face from each scan. For that, we employ our pose normalization method [13] that normalizes the pose of the face and localizes the tip of the nose. Before pose normalization was applied to the UND scan data, we applied a few basic preprocessing steps to the scan data: the 2D depth images were converted to triangle meshes by connecting the adjacent depth samples with triangles, slender triangles and singularities were removed, and only considerably large components were retained.

The cleaned surface meshes were randomly sampled, such that every $\approx$2.0 mm^2 of the surface is approximately sampled once. The pose normalization method uses these locations in combination with their surface normal as initial placements for a nose tip template. To locations where this template fits well, a second template of global face

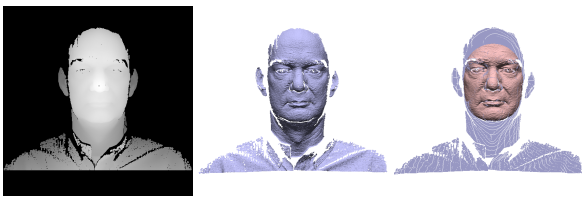


Fig. 1. Face segmentation. The depth image (left) is converted to a surface mesh (middle). The surface mesh is cleaned, the tip of the nose is detected and the face segmented (right, in pink).

features is fitted to normalize the face's pose and to select the tip of the nose. The face is then segmented by removing the scan data with a Euclidean distance larger than 100 mm from the nose tip. These face segmentation steps are visualized in Fig. 1.

4 Face Model Fitting

In general, 3D range scans suffer from noise, outliers, and missing data and their resolution may vary. The problem with single face scans, such as the UND face scans, is that large areas of the face are missing, which cannot be filled by simple hole filling techniques. When the morphable face model is fitted to a 3D face scan, a model is obtained that has no holes, has a proper topology, and has an assured resolution. By adjusting the m=99 weights w_i for the eigenvectors, the morphable model creates a new face instance. To fit the morphable model to 3D scan data, we need to find the optimal set of m weights w_i. In this section, we describe a fully automatic method that efficiently finds a proper model of the face scan in the m-dimensional space.

4.1 Distance Measure

To evaluate if an instance of the morphable face model is a good approximation of the 3D face scan, we use the Root Mean Square (RMS) distance of the instance's vertices to their closest points in the face scan. For each vertex point (p) from the instance (M_1), we find the vertex point (p') in the scan data (M_2) with the minimal Euclidean distance

$$e_{min}(p, M_2) = min_{p' \in M_2} d(p, p') \ , \tag{1}$$

using a kD-tree. The RMS distance is then measured between M_1 and M_2 as:

$$d_{rms}(M_1, M_2) = \sqrt{\frac{1}{n} \sum_{i=1}^{n} e_{min}(p_i, M_2)^2} \ , \tag{2}$$

using n vertices from M_1. Closest point pairs (p,p') for which p' belongs to the boundary of the face scan, are not used in the distance measure.

The morphable face model has n=75,972 vertices that cover the face, neck and ear regions and its resolution in the upward direction is three times higher than in its sideways direction. Because the running time of our measure is dependent on the number of vertices, we recreated the morphable face model such that it contains only the face (data within 110 mm from the tip of the nose) and not the neck and ears. To obtain a more uniform resolution of for the model, we reduced the upward resolution to one third of the original model. The number of vertices of this adjusted morphable mean face is now n=12,964 vertices, a sample every $\approx$2.6 mm^2 of the face area.

4.2 Iterative Face Fitting

With the defined distance measure for an instance of our compressed morphable face model, the m-dimensional space can be searched for the optimal instance. The fitting is done by choosing a set of m weights w_i, adjusting the position of the instance's

vertices according to $S_{inst} = \bar{S} + \sum_{i=1}^{m} w_i \sigma_i s_i$, measuring the RMS-distance of the new instance to the scan data, selecting new weights and continue until the optimal instance is found. Knowing that each instance is evaluated using a large number of vertices, an exhaustive search for the optimal set of m weights is too computationally expensive.

A common method to solve large combinatorial optimization problems is *simulated annealing* (SA) [14]. In our case, random m-dimensional vectors could be generated which represent different *morphs* for a current face instance. A morph that brings the current instance closer to the scan data is accepted (downhill), and otherwise it is either accepted (uphill to avoid local minima) or rejected with a certain probability. In each iteration, the length of the m-dimensional morph vector can be reduced as implementation of the "temperature" scheme. The problem with such a naive SA approach is that most random m-dimensional morph vectors are uphill. In particular close to the optimal solution, a morph vector is often rejected, which makes it hard to produce an accurate fit. Besides this inefficiency, it doesn't take the eigensystem of the morphable face model into account.

Instead, we propose an *iterative downhill walk* along the consecutive eigenvectors from a current instance towards the optimal solution. Starting from the mean face $\bar{S}$ ($\forall_{i=1}^{m} w_i = 0$), try new values for w_1 and keep the best fit, then try new values for w_2 and keep the best fit, and continue until the face is morphed downhill along all m eigenvectors. Then iterate this process with a *smaller search space* for w_i. The advantage in computation costs of this method is twofold. First, the discrete number of morphs in the selected search space directly defines the number of rejected morphs per iteration. Second, optimizing one w_i at a time means only a one (instead of m) dimensional modification of the current face instance $S_{new} = S_{prev} + (w_{new} - w_{prev})\sigma_i s_i$.

Because the first eigenvectors induce the fitting of global face properties (e.g. face height and width) and the last eigenvectors change local face properties (e.g. nose length and width), each iteration follows a global to local fitting scheme (see Fig. 2). To avoid local minima, two strategies are applied. (1) The selected w_i in one iteration is not evaluated in the next iteration, forcing a new (similar) path through the m-dimensional space. (2) The vertices of the morphable face model are uniformly divided over three

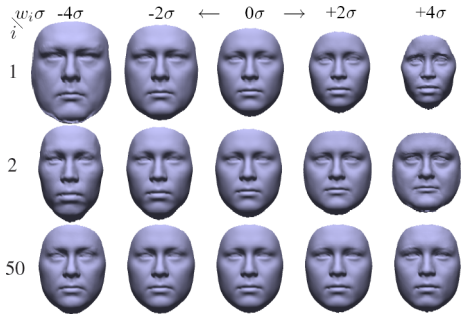


Fig. 2. Face morphing along eigenvectors starting from the mean face (center column). Different weights for the principal eigenvectors (e.g. i=1,2) changes the global face shape. For latter eigenvectors the shape changes locally (e.g. i=50).

sets and in each iteration a different set is modified and evaluated. Only in the first and last iteration all vertices are evaluated. Notice that this also reduces the number of vertices to fit and thus the computation costs.

The fitting process starts with the mean face and morphs in place towards the scan data, which means that the scan data should be well aligned to the mean face. To do so, the segmented and pose normalized face is placed with its center of mass on the center of mass of the mean face, and finely aligned using the Iterative Closest Point (ICP) algorithm [15]. The ICP algorithm iteratively minimizes the RMS distance between vertices. To further improve the effectiveness of the fitting process, our approach is applied in a *coarse fitting* and a *fine fitting* step.

4.3 Coarse Fitting

The more the face scan differs from the mean face $\bar{S}$, the less reliable the initial alignment of the scan data to the mean face is. Therefore, the mean face is coarsely fitted to the scan data by adjusting the weights of the first ten principal eigenvectors (m_{max}=10) in a single iteration (k_{max}=1) with 10 different values for w_{new}=[-1.35, -1.05, ..., 1.05, 1.35] as in Algorithm `ModelFitting`($\bar{S}$,$scan$). Fitting the model by optimizing the first ten eigenvectors results in the face instance S_{coarse}, with global face properties similar to those of the scan data. After that, the alignment of the scan to S_{coarse} is further improved with the ICP algorithm.

4.4 Fine Fitting

Starting with the improved alignment, we again fit the model to the scan data. This time the model fitting algorithm is applied using all eigenvectors (m_{max}=m) and multiple iterations (k_{max}=9). In the first iteration of Algorithm `ModelFitting`($\bar{S}$,$scan$), 10 new weight values w_{new} are tried for each eigenvector, to cover a large range of facial variety. The best w_{new} for every sequential eigenvector is used to morph the instance closer to the face scan. In the following k_{max}-1 iterations only four new weight values w_{new} are tried around w_i with a range w_{range} equal to w_{incr} of the previous iteration. By iteratively searching for a better w_i in a smaller range, the weights are continuously optimized. Local minima are avoided as described in Sect. 4.2. The range of the first iteration and the number of new weights tried in each next iteration were empirically selected as good settings.

4.5 Multiple Components

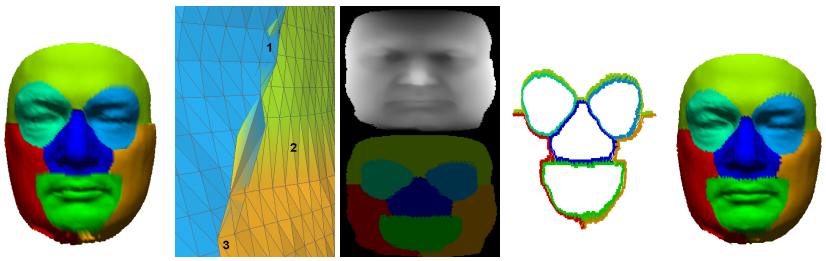
Knowing that the morphable model was generated from 100 3D face scans, an increase of its expressiveness is most likely necessary to cover a large population. To increase the expressiveness, also Blanz and Vetter [6] proposed to independently fit different components of the face, namely the eyes, nose, mouth, and the surrounding region. Because each component is defined by its own linear combination of shape parameters, a larger variety of faces can be generated with the same model. The fine fitting scheme from the previous section was developed to be applicable to either the morphable face model as a whole, but also to individual components of this model.

Algorithm 1. ModelFitting(S_{inst} to $scan$)

```
1: w_range = 1.5, w_incr = 0.3
2: for k ← 1 to k_max do
3:     select vertices (uniform subset of component)
4:     for i ← 1 to m_max do
5:         w_min = w_i − w_range + ½ w_incr
6:         w_max = w_i + w_range − ½ w_incr
7:         for w_new ← w_min to w_max do
8:             morph S_inst with w_new
9:             d_rms(S_inst, scan) smaller → keep w_new
10:            undo morph
11:            w_new = w_new + w_incr
12:        morph S_inst with w_i ← best w_new
13:    w_range = w_incr, w_incr = ½ w_incr
14: return S_inst
```

Component selection. All face instances generated with the morphable model are assumed to be in correspondence, so a component is simply a subset of vertices in the mean shape $\bar{S}$ (or any other instance). We define seven components in our adjusted morphable face model (see Fig. 3). Starting with the improved alignment, we can individually fit each of the components to the scan data using the fine fitting scheme, obtaining a higher precision of the fitting process (as shown in Sect. 6.1). Individual components for the left and right eyes and cheeks were selected, so that our method applies to non-symmetric faces as well. The use of multiple components has no influence on the fitting time, because the total number of vertices remains the same and only the selected vertices are modified and evaluated.

Component blending. A drawback of fitting each component separately is that inconsistencies may appear at the borders of the components. During the fine fitting, the border triangles of two components may start to intersect, move apart, or move across (Fig. 3). The connectivity of the complete mesh remains the same, so two components moving apart remain connected with elongated triangles at their borders. We solve these inconsistencies by means of a post-processing step, as described in more detail below.



(a) (b) (c) (d) (e)

Fig. 3. Multiple components (a) may intersect (b1), move apart (b2), or move across (b3). Simulating a cylindrical scan (c) and smoothing the new border vertices (d) solves these problems (e).

Knowing that the morphable face model is created from cylindrical range scans and that the position of the face instance doesn't change, it is easy to synthetically rescan the generated face instance. Each triangle of the generated face instance S_{fine} is assigned to a component (Fig. 3a). A cylindrical scanner is simulated, obtaining a cylindrical depth image $d(\theta, y)$ with a surface sample for angle θ, height y with radius distance d from the y-axis through the center of mass of $\bar{S}$ (Fig. 3c). Basically, each sample is the intersection point of a horizontal ray with its closest triangle, so we still know to which component it belongs. The cylindrical depth image is converted to a 3D triangle mesh by connecting the adjacent samples and projecting the cylindrical coordinates to 3D. This new mesh S'_{fine} has a guaranteed resolution depending on the step sizes of θ and y, and the sampling solves the problem of intersecting and stretching triangles. However, ridges may still appear at borders where components moved across. Therefore, Laplacian smoothing is applied to the border vertices and their neighbors (Fig. 3d). Finally, data further then 110 mm from the tip of the nose is removed to have the final model S_{final} (Fig. 3e) correspond to the segmented face. In Sect. 6.1, we evaluate both the single and multiple component fits.

5 Face Recognition

Our model fitting algorithm provides a clean model of a 3D face scan. In this section, we use this newly created 3D geometry as input for two 3D face matching methods. One compares facial landmarks and the other compares extracted contour curves.

Landmarks. All vertices of two different instances of the morphable model are assumed to have a one-to-one correspondence. Assuming that facial landmarks such as the tip of the nose, corners of the eyes, etc. are morphed towards the correct position in the scan data, we can use them to match two 3D faces. So, we assigned 15 anthropomorphic landmarks to the mean face and obtain their new locations by fitting the model to the scan data. To match two faces $\mathbf{A}$ and $\mathbf{B}$ we use the sets of c=15 corresponding landmark locations:

$$d_{corr}(\mathbf{A}, \mathbf{B}) = \sum_{i=1}^{c} d_p(a_i, b_i) \ , \tag{3}$$

where distance d_p between two correspondences a_i and b_i is the squared difference in Euclidean distance e to the nose tip landmark p_{nt}:

$$d_p(a_i, b_i) = (e(a_i, p_{nt}) - e(b_i, p_{nt}))^2 \ . \tag{4}$$

Contour curves. Another approach is to fit the model to scans $\mathbf{A}$ and $\mathbf{B}$ and use the new clean geometry as input for a more complex 3D face recognition method. To perform 3D face recognition, we extract from each fitted face instance three 3D facial contour curves, and match only these curves to find similar faces. The three curves were extracted and matched as described by ter Haar and Veltkamp [13].

In more detail, after pose normalization and the alignment of the face scan to both $\bar{S}$ and S_{coarse}, a correct pose of the face scan is assumed and thus a correct pose of the final face instance S_{final}. Starting from the nose tip landmark p_{nt}, 3D profile curves can

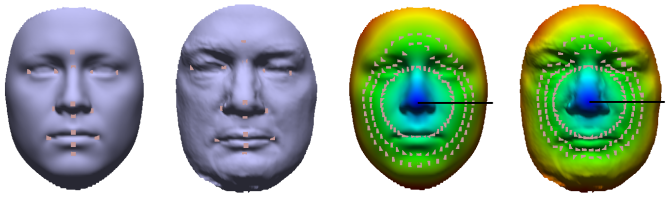


Fig. 4. The similarity of two 3D faces is determined using one-to-one correspondences, with on the left 15 corresponding landmarks and on the right 135 corresponding contour samples. The optimal XY-, C-, and G-contour curves (inner to outer) were extracted, for which the G-contour uses the (colored) geodesic distances. The line shown in black is one of the N_p profiles.

be extracted by walking the 3D surface in different directions (radii in the XY-plane). Samples along a profile from one face should correspond to samples along the same profile on another face. In case two faces are identical, these samples have the same Euclidean distance to the tip of the nose p_{nt}. For different faces, these samples cause a dissimilarity. The 3D face matching algorithm extracts N_p=45 profiles curves and extracts from each profile curve:

- One XY-sample, the location where the distance $\sqrt{(x^2+y^2)}$ to p_{nt} equals r.
- One C-sample, the location where the curve length of the profile to p_{nt} equals r.
- One G-sample, the location on the profile where the length of the shortest geodesic path over the entire surface to p_{nt} equals r.

The shortest geodesic paths were computed using the fast marching method [16]. The combination of N_p=45 XY-samples at the same distance r builds a XY-contour, similarly a C-contour and a G-contour are constructed. Based on a training set of morphable face instances, the curves that were found most distinctive were selected, namely the XY-contour at r=34 mm, the C-contour at r=68 mm, and the G-contour at r=77 mm. The information of each 3D face instance is now reduced to a set of 135 ($3 \times N_p$) 3D sample points, with one-to-one correspondence to the same set of 135 samples in a different face instance. The similarity of faces $\mathbf{A}$ and $\mathbf{B}$ is again defined by d_{corr}, with c=135 correspondences.

6 Results

The results described in this section are based on the UND face scans. For each of the 953 scans we applied our face segmentation method (Sect. 3). Our face segmentation method correctly normalized the pose of all face scans and adequately extracted the tip of the nose in each of them. The average distance and standard deviation of the 953 automatically selected nose tips to our manually selected nose tips was 2.3 ±1.2 mm.

Model fitting was applied to the segmented faces, once using only a single component and once using multiple components. Both instances are quantitatively evaluated in Sect. 6.1, and both instances were used for 3D face recognition in Sect. 6.2.

6.1 Face Model Fitting

In this section we evaluate the face model fitting as follows. Each segmented face was aligned to $\bar{S}$ and the coarse fitting method of Sect. 4.3 was applied. After the improved alignment of the scan data to S_{coarse}, the fine fitting method of Sect. 4.4 was applied to either the entire face (one component) or to each of the individual components (multiple components). For a fair comparison the same post-processing steps (Sect. 4.5) were applied to both S_{fine} instances. Fig. 5 shows qualitative better fits when multiple components are used instead of a single component. Globally, by looking at the more frequent surface interpenetration of the fitted model and face scan, which means a tighter fit. Locally, by looking at facial features, such as the nose, lips and eyes. Note that our fitting method correctly neglects facial hair, which is often a problem for 3D face recognition methods.

To quantitatively evaluate the produced fits, we determined the RMS distance (Eq. 2) for each of the fitted models to their face scan $d_{rms}(S_{final}, scan)$ and for the scan data to the face instance $d_{rms}(scan, S_{final})$. Points paired with boundary points are not included, so that results report merely the measurements in overlapping face regions. Results are reported over all 953 scans in Table 1. They show that our face morphing method provides accurate alignments of the morphable face model to the scan data for both the single component and multiple components. All results are in favor of multiple component morphs. Important to know is that the segmented UND faces have approximately twice the number of vertices compared to the fitted face model. Therefore, the closest point distances are higher for the scan to model case.

Fig. 5. Fitted face models S_{final} based on a single component (1st and 3rd column) and multiple components (2nd and 4rd column) to scan data in blue. Results from the front and side view, show a qualitative better fit of the multiple components to the scan data. The last two subjects on the right were also used in [7].

Table 1. The quantitative evaluation (in mm) of our face fitting method

measure	outliers	$M_1 \rightarrow M_2$	min	max	mean	sd
d_{rms}	yes	1 component $\rightarrow$ scan	0.478	3.479	0.776	0.176
d_{rms}	yes	7 components $\rightarrow$ scan	0.374	2.076	0.608	0.123
d_{rms}	yes	scan $\rightarrow$ 1 component	0.787	7.236	1.115	0.561
d_{rms}	yes	scan $\rightarrow$ 7 components	0.696	6.269	0.935	0.503
$d_{avr.depth}$	yes	scan $\leftrightarrow$ 1 component	0.393	4.704	0.692	0.290
$d_{avr.depth}$	yes	scan $\leftrightarrow$ 7 components	0.254	2.542	0.444	0.197
$d_{avr.depth}$	no	scan $\leftrightarrow$ 1 component	0.393	2.379	0.656	0.183
$d_{avr.depth}$	no	scan $\leftrightarrow$ 7 components	0.254	1.818	0.423	0.120

Table 2. Recognition rates and mean average precisions based on landmarks and contour curves for single and multiple component fits.

features	model fit	RR	MAP
landmarks	1 component	85.8%	0.872
landmarks	7 components	85.2%	0.862
contours	1 component	96.3%	0.952
contours	7 components	97.5%	0.967

Comparison. Blanz et al. [7] reported the accuracy of their model fitting method using the average depth error between the cylindrical depth images of the input scan and the output model. The mean depth error over 300 FRGC v.1 scans was 1.02 mm when they neglected outliers (distance $>$ 10 mm) and 2.74 mm otherwise. To compare the accuracy of our method with their accuracy, we produced cylindrical depth images (as in Fig. 3c) for both the segmented face scan and the fitted model and computed the average depth error $|d_{scan}(\theta, y) - d_{final}(\theta, y)|$ without and with the outliers. For the fitted single component these errors $d_{avr.depth}$ are 0.656 mm and 0.692 mm, respectively. For the fitted multiple components these errors are 0.423 mm and 0.444 mm, respectively. So even our single component fits are more accurate then those of Blanz et al.

Our time to process a raw scan requires $\approx$3 seconds for the face segmentation, $\approx$1 second for the coarse fitting, and $\approx$30 seconds for the fine fitting on a Pentium IV 2.8 GHz. Blanz method reported $\approx$4 minutes on a 3.4 GHz Xeon processor, but includes texture fitting as well. Huang et al. [9] report for their deformation model a matching error of 1.2 mm after a processing time of 4.6 minutes.

6.2 Face Recognition

As described in Sect. 5, we can use the 953 morphed face instances to perform 3D face recognition. For this experiment, we computed the 953 $\times$ 953 dissimilarity matrix and generated for each of the 953 queries a ranked list of face models sorted in decreasing similarity. From these ranked lists, we computed the recognition rate (RR) and the mean average precision (MAP). A person is recognized (or identified) when the face retrieved on top of the ranked list (excluding the query) belongs to the same subject as the query. For 77 subjects only a single face instance is available which cannot be identified, so the RR is based on the remaining 876 queries. The mean average precision (MAP) of

the ranked lists are reported, to elaborate on the retrieval of all relevant faces, i.e. all faces from the same subject.

Four 3D face recognition experiments were conducted, namely face recognition based on landmark locations from the fitted single component and the fitted multiple components, and based on contour curves from the fitted single component and the fitted multiple components. Results in Table 2 show that the automatically selected anthropomorphic landmarks are not reliable enough for effective 3D face recognition with 85.8% and 85.2% recognition rates (RR). Notice that the landmarks obtained from the single component fit perform better than those from the multiple component fit. This is probably caused by three landmarks (outer eye corners and Sellion) lying close to component boundaries, where the fitting can be less reliable.

The fitted face model is an accurate representation of the 3D scan data. This accuracy allows the contour based method to achieve high recognition rates (see Table 2). For the single component fits, the contour matching achieves a RR of 96.3% and for multiple component fits even 97.5%. For a high recognition rate, only one of the relevant faces in the dataset is required on top of each ranked list. The reported MAPs show that most of the other relevant faces are retrieved before the irrelevant ones. Some of the queries that were not identified, have a non-neutral expression (happy, angry, biting lips, etc.) while its relevant faces have a neutral expression. A face recognition method invariant to facial expressions, will most likely increase the performance even further.

Comparison. Blanz et al. [7] achieved a 96% RR for 150 queries in a set of 150 faces (from the FRGC v.1). To determine the similarity of two face instances, they computed the scalar product of the 1000 obtained model coefficients. Using a set of facial depth curves, Samir et al. [17] reported a 90.4% RR for 270 queries in a set of 470 UND scans. Mian et al. [18] reported a 86.4% RR for 277 queries in a set of 277 UND scans.

7 Concluding Remarks

Where other methods need manual initialization, we presented a fully automatic 3D face morphing method that produces a fast and accurate fit for the morphable face model to 3D scan data. Based on a global to local fitting scheme the face model is coarsely fitted to the automatically segmented 3D face scan. After the coarse fitting, the face model is either finely fitted as a single component or as a set of individual components. Inconsistencies at the borders are resolved using an easy to implement post-processing method. Our results show that the use of multiple components produces a tighter fit of the face model to the face scan, but assigned anthropomorphic landmarks may lose their reliability for 3D face identification. Face matching using facial contours, shows higher recognition rates based on the multiple component fits then for the single component fits. This means that the obtained 3D geometry after fitting multiple components has a higher accuracy. With a recognition rate of 97.5% for a large dataset of 3D faces, our model fitting method proves to produce highly accurate fits usable for 3D face recognition.

Acknowledgements

This research was supported by the FP6 IST Network of Excellence 506766 AIM@-SHAPE and partially supported by FOCUS-K3D FP7-ICT-2007-214993. The authors thank the University of South Florida for providing the USF Human ID 3D Database.

References

1. Bowyer, K.W., Chang, K., Flynn, P.: A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition. CVIU 101(1), 1–15 (2006)
2. Scheenstra, A., Ruifrok, A., Veltkamp, R.C.: A Survey of 3D Face Recognition Methods. In: Kanade, T., Jain, A., Ratha, N.K. (eds.) AVBPA 2005. LNCS, vol. 3546, pp. 891–899. Springer, Heidelberg (2005)
3. Bronstein, A.M., Bronstein, M.M., Kimmel, R.: Three-dimensional face recognition. IJCV 64(1), 5–30 (2005)
4. Berretti, S., Del Bimbo, A., Pala, P., Silva Mata, F.: Face Recognition by Matching 2D and 3D Geodesic Distances. In: Sebe, N., Liu, Y., Zhuang, Y.-t., Huang, T.S. (eds.) MCAM 2007. LNCS, vol. 4577, pp. 444–453. Springer, Heidelberg (2007)
5. Davis, J., Marschner, S.R., Garr, M., Levoy, M.: Filling holes in complex surfaces using volumetric diffusion. 3DPVT, 428–861 (2002)
6. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3D faces. SIGGRAPH, 187–194 (1999)
7. Blanz, V., Scherbaum, K., Seidel, H.P.: Fitting a Morphable Model to 3D Scans of Faces. In: ICCV, pp. 1–8 (2007)
8. Lu, X., Jain, A.: Deformation Modeling for Robust 3D Face Matching. PAMI 30(8), 1346–1356 (2008)
9. Huang, X., Paragios, N., Metaxas, D.N.: Shape Registration in Implicit Spaces Using Information Theory and Free Form Deformations. PAMI 28(8), 1303–1318 (2006)
10. Kakadiaris, I., Passalis, G., Toderici, G., Murtuza, N., Theoharis, T.: 3D Face Recognition. In: BMVC, pp. 869–878 (2006)
11. Sarkar.S.: USF HumanID 3D Face Database. University of South Florida
12. Chang, K.I., Bowyer, K.W., Flynn, P.J.: An Evaluation of Multimodal 2D+3D Face Biometrics. PAMI 27(4), 619–624 (2005)
13. ter Haar, F.B., Veltkamp, R.C.: A 3D Face Matching Framework. In: Proc. Shape Modeling International (SMI 2008), pp. 103–110 (2008)
14. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by Simulated Annealing. Science 220, 4598, 671–680 (1983)
15. Besl, P.J., McKay, N.D.: A method for registration of 3D shapes. PAMI 14(2), 239–256 (1992)
16. Kimmel, R., Sethian, J.: Computing geodesic paths on manifolds. Proc. of National Academy of Sciences 95(15), 8431–8435 (1998)
17. Samir, C., Srivastava, A., Daoudi, M.: Three-Dimensional Face Recognition Using Shapes of Facial Curves. PAMI 28(11), 1858–1863 (2006)
18. Mian, A.S., Bennamoun, M., Owens, R.: Matching Tensors for Pose Invariant Automatic 3D Face Recognition. IEEE A3DISS (2005)