Silhouette-Based 3D Face Shape Recovery

Citation	Lee, Jinho, Baback Moghaddam, Hanspeter Pfister, and Raghu Machiraju. 2003. Silhouette-based 3D face shape recovery. In Proceedings: Graphics Interface 2003: June 11-13, 2003, Halifax, Nova Scotia, ed. T. Möller, and C. Ware, 21-30. Natick, M.A.: A K Peters.
Published Version	http://www.graphicsinterface.org/
Accessed	February 18, 2015 3:57:30 PM EST
Citable Link	http://nrs.harvard.edu/urn-3:HUL.InstRepos:4141477
Terms of Use	This article was downloaded from Harvard University's DASH repository, and is made available under the terms and conditions applicable to Other Posted Material, as set forth at http://nrs.harvard.edu/urn-3:HUL.InstRepos:dash.current.terms-of-use#LAA

Silhouette-Based 3D Face Shape Recovery

Jinho Lee* Baback Moghaddam† Hanspeter Pfister† Raghu Machiraju*

* The Ohio State University
† Mitsubishi Electric Research Laboratories

Abstract

The creation of realistic 3D face models is still a fundamental problem in computer graphics. In this paper we present a novel method to obtain the 3D shape of an arbitrary human face using a sequence of silhouette images as input. Our face model is a linear combination of *eigenheads*, which are obtained by a Principal Component Analysis (PCA) of laser-scanned 3D human faces. The coefficients of this linear decomposition are used as our model parameters. We introduce a near-automatic method for reconstructing a 3D face model whose silhouette images match closest to the set of input silhouettes.

Key words: Face model, eigenhead, principal component analysis, model fitting, silhouette images.

1 Introduction

Creating realistic 3D face models is an important problem in computer graphics and computer vision. Most existing methods either require a lot of manual labor by a skilled artist, expensive active light 3D scanners [4, 11], or the availability of high quality texture images as a substitute for exact face geometry [7, 12, 30]. More recent efforts have focused on the availability of an underlying model for human faces [1, 2, 19, 24]. These model-based approaches make use of the fact that human faces do not vary much in their general characteristics from person to person.

We follow the model-based approach to reconstruct accurate human face geometry from photographs. Our underlying face model is not synthetic but is based on real human faces measured by laser-based cylindrical scanners. This data-driven face model is limited in its expressive power by the number and variety of the faces in the training database. However, it can be easily expanded by incorporating new faces into the existing database.

Our approach is most similar to the method of Blanz and Vetter [2]. But instead of deriving an approximate textured 3D face from a single photograph, we focus on acquiring relatively accurate geometry of a face from multiple silhouette images at more affordable cost and with less user interaction.

Why silhouettes? Using silhouettes separates the geometric subtleties of the human head from the nuances of shading and texture. As a consequence we do not require knowledge of rendering parameters (e.g., light direction, intensity, etc.) which need to be specified by the user and adjusted by an optimization process as in [2].

The use of geometry for face reconstruction and synthesis is supported by the premise that for a demographically diverse dataset (across gender and race) anthropometric and hence structural variations best classify various groups and races. Texture often increases the uncertainty in the classification process. On the other hand, accurately measured reflectance values can increase the robustness of the methods. However, texture and reflectance measurements may be used to disambiguate and synthesize new faces after reconstructing the geometry.

Another motivation to use silhouette images rests on the assumption that a set of carefully chosen viewpoints would generate a unique sequence of face silhouettes for each individual. Therefore, the set of silhouette images would be sufficient to recover an optimal face in our face space. To verify this premise, we built a system for capturing silhouette images of a human face by eleven calibrated cameras.

Finally, to match silhouette images generated by our face model to the given silhouette images, we adopt an inverse design and optimization approach through an objective function which measures the error between two silhouette images. Our 3D model faces are not full in their extent; the models are deprived of hair and also do not include the back of the head. Whereas, the input silhouette images include the entire head. Thus, the input silhouette has always larger area than the synthesized silhouette. We address this problem of partial silhouette matching in a novel way through choice of appropriate error metrics. As we will show in this paper, silhouettes provide expedient and robust reconstruction.

We now enumerate the significant contributions of our paper:

- We report a robust and efficient method to reconstruct human faces from silhouettes.
- Few user-specified parameters are required making our method close to an automatic method.

- We report a novel algorithm for establishing correspondence between two faces.
- We use a novel and efficient error metric, *boundary weighted XOR* in our optimization procedures.
- The method is very robust even when presented with partial information of the human head.
- Our method is resolution-independent allowing for expedient reconstructions tailored for a given display.
- We report extensive experimental data and statistical analysis to support the efficacy of our methods.

In Section 2 we describe relevant previous work on face reconstruction. Then, in Section 3 we describe our face model. Section 4 formulates the inverse problem of reconstructing a 3D face from its silhouette images. In Section 5 we describe our results when we apply our technique to a face database. Section 6 provides a summary of our work and points to future research.

2 Background and Related Work

Principal Component Analysis (PCA) [9] is a statistical method to extract the most salient directions of data variation from large multidimensional datasets. Though low dimensional representation using the PCA is a popular method for the synthesis and recognition of 2D face images [3, 17, 18, 25, 31], its application to 3D face geometry is relatively rare and not well explored.

Atick *et al*. [1] proposed a method to use eigenheads to solve a *shape from shading* problem by leveraging the knowledge of object class, which was used to recover the shape of a 3D human face from single photograph. Jebara *et al*. [8] used modular eigenspaces for 3D facial features and their correlation with the texture to reconstruct the structure and pose of a human face in the live video sequences.

As pointed out earlier, Blanz and Vetter [2] formulated an optimization problem to reconstruct textured 3D face from one or more photographs in the context of inverse rendering. Our formulation is similar in essence. However, our implementation of various stages are more robust and amenable to efficient realizations.

For instance, let us consider the techniques used to derive correspondence between head models. A 3D face model is often obtained from a laser scanner which samples surface of a face uniformly in cylindrical coordinates. For a successful application of PCA, one needs the same number of 3D vertex positions among the various faces in the training set or the database. The easiest way to do so is to exploit the uniform cylindrical space in which the original laser-scanned data is stored [1]. This method, however, does not exploit the point-to-point correspondence across faces. Moreover, if the scale of the faces varies across samples (e.g. a young subject vs. fully grown male), only partial set of points on the larger object will be relevant.

Blanz and Vetter used a 3D variant of a gradient-based optical flow algorithm to derive the necessary point-to-point correspondence [32]. Their method also employs color and/or texture information acquired during the scanning process. This approach will not work well for faces of different races or in different illumination given the inherent problems of using static textures. We present a simpler method of determining correspondences that does not depend on the color or texture information.

Shape from silhouette techniques have been used to reconstruct three dimensional shapes from multiple silhouette images of an object [10, 14, 20, 27, 33]. The reconstructed 3D shape is called a visual hull, which is a maximal approximation of the object consistent with the object's silhouettes. The accuracy of this approximate visual hull depends on the number and location of the cameras used to generate the input silhouettes. In general, a complex object such as the human face does not yield a good shape when approximated by a visual hull using a small number of cameras. Moreover, human faces possess concavities (e.g. eye sockets and philtrum) which are impossible to reconstruct even in an exact visual hull due to its inherent limitation (See Figure 1).

However, using knowledge of the object to be reconstructed, silhouette information can be exploited as an important constraint for the exact shape of the object. We use the shape coefficients of an eigenhead model as the model parameters to be fit to a sequence of silhouette images.

There has been work reported on recovering other kinds of parameters using the knowledge of object class in the context of optimization by inverse rendering. In [26], a method was presented to search the optimal configuration of human motion parameters by applying a novel silhouette/contour likelihood term. Lensch *et al*. [13] recovered internal/external camera parameters using exact information of an object and its silhouette images. Our error metric is similar to the area-based difference measure used in [13] but provides more elaborate guidance for the convergence of an inverse method in the presence of noise and clutter.

3 Face Model: Eigenheads

In this section, we describe our face model in a low dimensional *3D face space* and a novel method to obtain

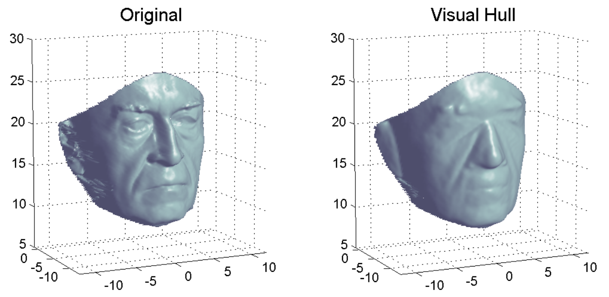

Figure 1: Original laser-scanned face vs. visual hull obtained using 50 viewpoints.

the point-to-point correspondence among 3D faces for the subsequent eigen-analysis.

3.1 Principal Component Analysis

We applied PCA to a database of 3D geometries of human faces. As a consequence, we can now define face geometries with *eigenheads* [1]. This decomposition can be used to reconstruct a new or existing face through the linear combination of these eigenheads. Therefore, a face model is given by

$$H(\boldsymbol{\alpha}) = \boldsymbol{h_0} + \sum_{m=1}^{M} \alpha_m \boldsymbol{h_m} \tag{1}$$

and the model parameter is $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \ldots, \alpha_M\}$, where $\boldsymbol{h_m}$ is the m^{th} eigenhead and $\boldsymbol{h_0}$ is the mean or average head.

Figure 2 illustrates how PCA captures the four largest variations of faces in the database. The first mode captures overall scale of faces which is correlated with gender information. The second mode depicts variations in the shape of chin. The third mode describes the overall length of faces and the fourth mode captures salient aspects of race.

Our face database comes from USF dataset [29] and consists of Cyberware scans of 97 male adult and 41 female adult faces with a mixture of race and age. All faces in the database were resampled to obtain point-to-point correspondence using the technique described in the following subsection and then aligned to a reference face to remove any contamination of the PCA caused by pose variation and/or misalignment.

3.2 Correspondence

Let each 3D face in a face database be $F_i, i = 1..N$. Since the number of vertices (M_i) in F_i varies, we resample all faces so that they have the same number of vertices all in mutual correspondence. This is required given the need to achieve correspondence in feature points across all F_i. In other words, the tip of the nose of F_i should be mapped to the tip of the nose of F_j, and so on. We define a reduced set of 26 landmark feature points in a face F_i as $Q_i = \{q_{i,1}, q_{i,2}, ..., q_{i,m}\}$, where m is the number of feature points and $q_{i,j}$ is the vertex index for a specific feature point. Let $\mathbf{q_{i,k}} = (x, y, z)$ be the location of the feature point k in Cartesian coordinate space. Then, the problem of deriving full correspondence for all models F_i is stated as: resample the M vertices for all F_i under the constraint $q_{i,k} = q_{j,k}, i \neq j$ for all i, j and k.

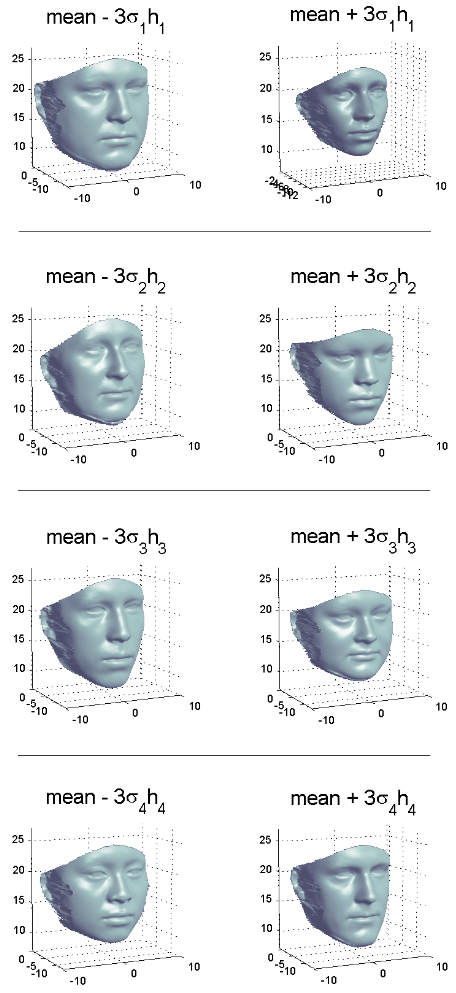

Figure 2: Visualization of the first four eigenheads. σ_i is the square root of i^{th} eigenvalue of the corresponding $\boldsymbol{h_i}$

Our method is composed of the following steps:

1. Select a reference face F_r, which is the closest face to the mean face in the database.
2. Determine locations of feature points and select m feature points from each F_i manually.

3. Deform F_r so that it fits the target face F_i. This requires the interpolation of all points in F_r under the constraint $\mathbf{q_{r,k}} = \mathbf{q_{i,k}}$. Let the deformed face be F_i^d. Now F_i^d has a shape similar to F_i since both have same locations for the all m feature points. Note that F_i^d has exactly the same number of points as F_r.

4. For each point in F_i^d, sample a point on the surface of F_i in the direction of underlying cylindrical projection (as defined by the scanner configuration). Let the resulting resampled point set be F_i^s which satisfies the constraints on the feature locations $q_{r,k} = q_{i,k}^s$ and $\mathbf{q_{i,k}} = \mathbf{q^s_{i,k}}$.

5. Repeat step 3 and step 4 for all F_i's $(i \neq r)$ in database.

For step 3, a standard model for scattered data interpolation can be exploited [16, 19]. Note that, at step 4, we cannot get corresponding samples on the surface of F_i for some points on the boundary of F_i^d. It is likely that the two faces under consideration do not match exactly on the boundary. We keep track of the indices of those void sample points and use only sample points whose indices are not void in any resampling of F_i in the database. Figure 3 depicts the process to establish the correspondence between reference and target faces.

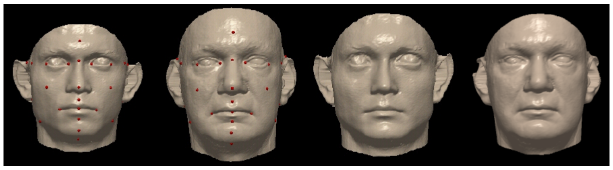

Figure 3: Getting correspondence between two faces. From left to right, reference face, target face, warped reference face, resampled target face. Note the void samples in the ears of the resampled target face.

4 Fitting Model Parameters to Silhouette Images

In this section, we describe our method for fitting model parameters to a set of input silhouette images. Generally, this fitting process does not require a specific face model. A novel weighted silhouette contour cost is presented in Section 4.3. The optimization strategy described in Section 4.4 depends on the underlying face model. We describe how our face model and database is adapted to a specific non-linear multidimensional optimization algorithm.

4.1 Problem Formulation

Let $M(\boldsymbol{\alpha})$ be any arbitrary face model which produces a polygon mesh given a vector parameter $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \cdots, \alpha_n\}$. Let $S^k, k = 1..K$ be an input silhouette image captured by camera C^k. Also, let T be a similarity transformation that aligns a reference model face to the real 3D face. Then, $S^k_m(\boldsymbol{\alpha})$ is a silhouette image rendered by projecting $T(M(\boldsymbol{\alpha}))$ onto an image plane using the calibration information of the given camera C^k. We discuss how we obtain this transformation in the next subsection.

Provided we define a cost function f that measures the difference of two silhouette images, our goal is to find $\boldsymbol{\alpha}$ that minimizes the total penalty

$$E(\boldsymbol{\alpha}) = \sum_{k=1}^{K} f(S^k, S^k_m(\boldsymbol{\alpha})) \tag{2}$$

for a suitable cost function f.

4.2 Solving The Alignment Problem

Finding the alignment transformation T is not trivial using only the silhouette information. The form of T depends on the pose and size of the face of a person to be captured. T can be defined as

$$T(\mathbf{x}) = s(\mathbf{R}\mathbf{x} + \mathbf{t}),$$

where s is a scale factor, $\mathbf{R}$ is a rotation matrix, $\mathbf{t}$ is a translation vector. The alignment problem is then one of minimization of the functional:

$$\sum_{j=1}^{L} \|\mathbf{x}_j - T(\mathbf{y}_j)\|^2, \tag{3}$$

in terms of s, $\mathbf{R}$ and $\mathbf{t}$. It should be noted that $\mathbf{x}_j$ is the j^{th} 3D feature point in real face, $\mathbf{y}_j$ is the j^{th} 3D feature point in a reference model face and L is the number of feature points to be used.

We already know $\mathbf{y}_j$. However, $\mathbf{x}_j$ is determined from a standard non-linear least square minimization technique [21, 28]. A Levenberg-Marquardt algorithm is applied to obtain the 3D point locations that correspond to L feature points selected manually in a small number of (3-4) texture images. We used $L = 7$ in our experiments. Once we determine $\mathbf{x}_j$, then, we compute the values of s, $\mathbf{R}$ and $\mathbf{t}$ such that Eq.(3) is minimized. The needed parameters are obtained from an application of the *full ordinary Procrustes analysis* [5].

4.3 Partial Silhouette Matching

Now, we discuss how we design the cost function f in Eq.(2). The easiest way to measure difference of two binary images is the number of 'on' pixels when pixelwise XOR operation is applied to the two images [13]. In this case,

$$f(S^k, S^k_m(\boldsymbol{\alpha})) = \sum_{i}^{H} \sum_{j}^{W} c(i,j) \tag{4}$$

$$c(i,j) = \begin{cases} 0 & \text{if } S^k(i,j) = S^k_m(\boldsymbol{\alpha})(i,j) \\ 1 & \text{otherwise.} \end{cases}$$

If our goal requires that $f = 0$, that is, if two silhouettes overlap exactly, the optimal solution will be unique in terms of $S^k_m(\boldsymbol{\alpha})$. However, if our objective function f cannot be reduced to zero given inherent characteristics of the problem, it is likely that there are multiple optimal solutions. Any preference among those multiple optimal solutions should be incorporated in the cost function.

In our case, the input silhouette area covers the full head including hair and the back, while our face model includes the front of the face delineated by the ears on the sides and lower part of the forehead from the top. Thus, our objective function, f, is often non-zero (or $f > 0$) since the silhouette generated by our model ($S^k_m(\boldsymbol{\alpha})$) considers only a partial area of the input silhouette (S^k) (see Figure 8 and Figure 10). If we use the objective function f in Eq.(4), we could have multiple set of $S^k_m(\boldsymbol{\alpha})$ that minimize f and we cannot guarantee that these solutions match the real boundary contours in the input silhouettes. Our goal is to match the real boundary contours between input and model silhouettes and f is required to be the global minimum. Accordingly, we impose higher penalty for the mismatch near the boundary pixels of input silhouettes.

Though a mismatch in the pseudo contour area contributes a higher cost to f, this contribution can be considered as a constant factor. Our new cost function replaces $c(i,j)$ in Eq.(4) with

$$c(i,j) = \begin{cases} 0 & \text{if } S^k(i,j) = S^k_m(\boldsymbol{\alpha})(i,j) \\ \frac{1}{d(i,j)^2} & \text{otherwise} \end{cases} \quad (5)$$

$$d(i,j) = D(S^k)(i,j) + D(\tilde{S}^k)(i,j),$$

where $D(S)$ is the Euclidean distance transform of binary image S and $\tilde{S}$ is the inverse image of S. Note that d represents a distance map from silhouette contour and can be computed once in a preprocessing step. We call this cost function *boundary-weighted XOR* , which provides a simple and effective alternative to precise contour matching schemes. As a result, there is no need for expensive operations of correspondence, edge-linking, curve-fitting, distance computations between boundary curves; all needed when precise contour matching schemes are used. Thus, our optimization algorithms are fast and robust.

4.4 Optimization

To minimize Eq.(2), we use a *downhill simplex method* which requires only function evaluation [13, 21]. The optimization parameter is the model parameter $\boldsymbol{\alpha}$. One function evaluation includes the following step:

1 Compute a mesh G from $M(\boldsymbol{\alpha})$.

2 Compute the aligned mesh $T(G)$.

3 For each input silhouette S^k,

- Project $T(G)$ into k^{th} image plane and generate silhouette image S^k_m.
- Compute *boundary-weighted XOR* between S^k and S^k_m and add it to the total cost.

This optimization process depends on the characteristics of the model parameter. Here, we discuss the optimization process based on our model parameter described on Section 3. Among the 137 eigenheads, we chose the first 60 eigenheads to reconstruct a 3D face. Furthermore, we found this number to be sufficient to capture most of the salient features in a human face. Thus, the corresponding coefficients serve as our multi-dimensional optimization parameter of dimensionality 60.

The simplex method can be easily adapted to our multi-dimensional face model. The initial simplex of 60 dimensions consists of 61 vertices. Let the coefficients $\boldsymbol{\alpha} = \{0, \cdots, 0\}$ (corresponding to the mean face) be one of the initial points $\mathbf{p}_0$ of the simplex. We can choose the other remaining 60 points to be

$$\mathbf{p}_i = \mathbf{p}_0 + \mu_i \mathbf{e}_i, \quad i = 1..60,$$

where $\mathbf{e}_i$'s are 60 unit vectors and μ_i can be defined by the characteristic length scale of each component of $\boldsymbol{\alpha}$. We set $\mu_i = 3\sqrt{\lambda_i}$, where λ_i is the i^{th} eigenvalue corresponding to i^{th} eigenhead in our face model. With this initial configuration, the movement of this 60 dimensional simplex is confined to be within our face space and there is no need to perform exhaustive searches in the exterior of the face space. Another noteworthy aspect of our optimization procedure in the chosen face space is that it is resolution-independent. This allows for very expedient reconstructions.

Although, the downhill simplex method has slow convergence properties, the choice of the error metric can improve it's efficiency significantly. The choice of our boundary-weighted XOR error metric has proven to be very beneficial given its low cost and simplicity. Our results reported in a later section bear testimony to this claim.

4.5 Texture Extraction

Our optimized 3D model matches all input silhouette images as close as possible. Since the input silhouette images are obtained from the corresponding texture images, we do not need any further registration process for texture extraction. We extract texture colors in object space rather than image space and do not create a single texture map image. That is, for each 3D vertex in the reconstructed 3D face, we assign a color value which is determined from multiple texture images. To do so, we proceed as follows.

Our approach is a view-independent texture extraction approach [19, 23, 30]. Each vertex is projected to all image planes and tested if the projected location is within the silhouette area and if the vertex is visible (not occluded) at each projection. For all valid projections, we compute the dot product between the vertex normal and the viewing direction, and use the dot product as a weight of the texture color sampled at the projected image location. The final color value at a vertex is computed by dividing the weighted sum of texture values of all valid projections by the sum of weights.

5 Experiments

In section 5.1, we discuss some implementation issues regarding the speed of optimization process. In the subsequent subsections, we provide experimental results for our silhouette fitting process described in Section 4 with several different camera settings.

5.1 Implementation Issues

One concern is the speed of the optimization process. The most time-consuming part in a function evaluation is the silhouette generation part (See Step 3 in Section 4.4). Since our face model is of very high resolution (approximately 48000 vertices and 96000 triangles), even rendering with flat shading takes considerable time when it should be repeated in an optimization process.

A simple remedy for this problem is to reduce the mesh resolution by vertex decimation. Also, if we reduce the mesh resolution, it is natural to reduce the resolution of silhouette images accordingly (originally 1024×768). The reduction in model and image resolution will accelerate the XOR computation process in Step 3. In our experiments, we determined that 95% decimation in the mesh and 50% reduction in image resolution resulted in a similar convergence rate and a lower (1/10) cost of that required for original resolution data. With this reduced resolution data, the total optimization expended only 3-4 minutes on an Intel Pentium IV, 2 GHz microprocessor. Note that this reduction in input data resolution does not affect the resolution of the final reconstruction. Once we estimate the optimal coefficients, we can reconstruct a 3D face in full resolution from the eigenheads using Eq.(1).

Another way to expedite the optimization process is to employ a hierarchical approach [13]. With more reduced resolution, we can obtain an approximation of the solution that can be achieved with original resolution data in even lesser time. For example, 99% decimation in mesh resolution and a 75% reduction in image resolution resulted in only 30-40 seconds until convergence. We can provide this solution obtained at a lower resolution as a initial guess of the optimization process at a higher resolution. As a result, it is likely better results can be obtained than those obtained using only high resolution data. All the results presented here were obtained from this hierarchical optimization technique.

Note that the shape parameters (α) are not directly dependent on the input silhouette image resolution and do not dictate the 3D output mesh resolution. The degree of resolution-independence built into our scheme is a very desirable feature. Our statistical shape model captures fine details (as being correlated with coarser ones) which allows us to use lower-resolution sensing in the input images and low-resolution XOR computations for shape recovery.

5.2 Synthetic Data

Synthetic data can be derived from our face model space directly. To show the robustness of our method, we chose 50 sample faces in the database and 50 faces reconstructed by randomly chosen parameters, $\alpha_i = (-0.8\sqrt{\lambda_i}, 0.8\sqrt{\lambda_i}), i = 1..60$, according to the Gaussian distribution. Eleven synthetic cameras were positioned in the front hemisphere around the object (Figure 7). The input silhouette images were acquired by rendering each of the sample faces in the eleven image planes. Besides the cost value, we measured L_2 and Hausdorff distance between each reconstructed face and corresponding original face in 3D.

Table 1 lists the various statistical estimators of the errors for all 100 samples. Table 2 demonstrates that our cost value based on the difference in 2D silhouette images has strong correlation with L_2 distance in 3D. Also, by comparing all 100 reconstructed 3D faces to the original faces visually, we could see the L_2 error has strong correlation with the visual similarity of two 3D faces. One important conclusion we can draw from this observation is that silhouette matching with sufficiently large number of viewpoints provides us with a very good estimate of the shape of a human face assuming that the target face is already in the 3D face space that is spanned by the eigenheads.

	min	max	mean	med.	std. dev.
XOR cost	1509	4104	2579	2527	600.8
L_2	12.59	115.3	45.44	39.90	20.40
Hausdorff	0.297	2.826	0.762	0.676	0.424

Table 1: Statistical estimators of errors

Figure 4 shows resulting reconstructions from our op-

	XOR cost	L_2	Hausdorff
XOR cost	1	0.89	0.70
L_2	0.89	1	0.79
Hausdorff	0.70	0.79	1

Table 2: Correlation coefficients between error types

timization process. The selected faces in the figure cause the minimum, average, and the maximum L_2 error among all the 100 samples. We observe that our silhouette matching algorithm captures the most important features of a face within our constructed face space. Figure 5 shows the difference between input silhouette images and the rendered silhouette images of the 3D face that results in an average L_2 error before and after optimization.

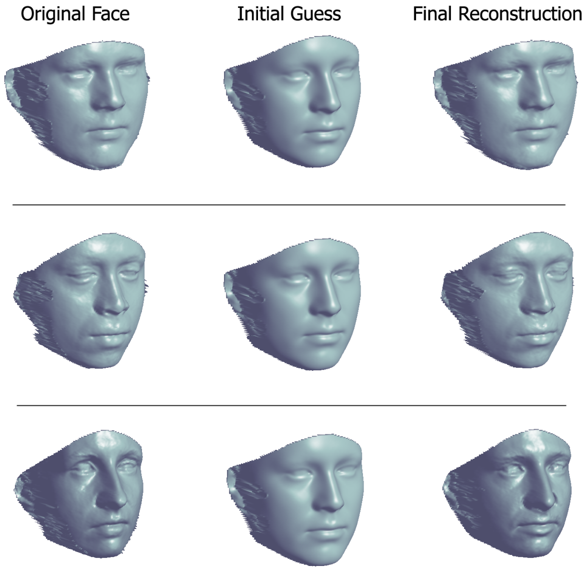

Figure 4: Reconstruction of synthetic faces: (top) minimum L_2 error, (middle) average L_2 error, (bottom) maximum L_2 error.

5.3 Camera Arrangement

Like the visual hull method, it is important to choose the viewpoints carefully to get maximal 3D information from a set of silhouette images. We repeated the experiment in the previous section with different camera arrangements. Eleven cameras were sampled on the front hemisphere around the object (see Figure 7). Figure 6 shows four different arrangements in 2D plots, which parameterize the shaded area in Figure 7 in spherical coordinates at a fixed radial distance; actual camera locations of circle marks are in the symmetric positions of ϕ-axis.

Table 3 compares the results for the four camera arrangements. Restricting the camera placement of the along vertical axis (Figure 6c) improved the fidelity of our cost function slightly. Note that denser sampling around the side area of a face (Figure 6d) did not improve the result in terms of both errors. All the experiments in Section 5.2 and Section 5.4 were performed with the arrangement depicted in Figure 6b.

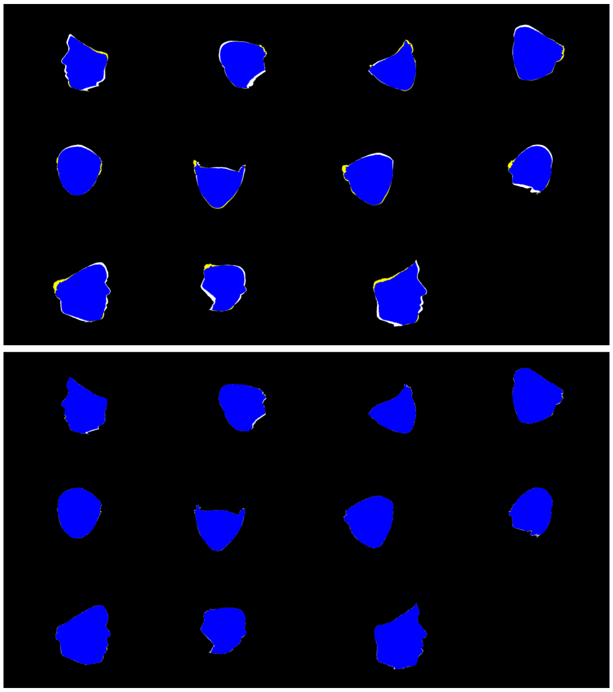

Figure 5: Silhouette difference of a synthetic face before (above) and after (below) optimization.

	avg. XOR cost	avg. L_2	corr. coef
(a)	2759	45.78	0.89
(b)	2579	45.44	0.89
(c)	2807	49.60	0.92
(d)	2634	46.73	0.89

Table 3: Errors obtained from different camera settings.

5.4 Real Data

The challenges in using pictures taken by real cameras include the issues of silhouette acquisition, accuracy of camera parameters, misalignment, and ‘clutter’ (excess head area beyond the face model). We assume that silhouette images can be easily acquired by a simple background subtraction technique. We calibrated the eleven static cameras (Figure 7) by a standard technique using a calibration object [28]. One could enhance this initial camera calibration by a technique that uses silhouette images [6, 22]. In Section 4.3 we describe how we avoid the

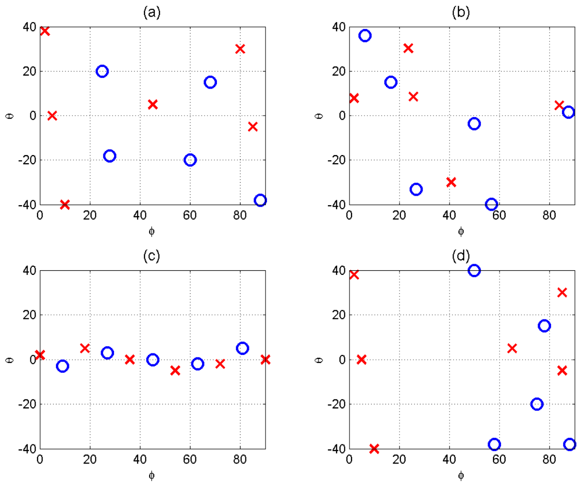

Figure 6: Different arrangements of eleven cameras. (a) evenly distributed set. (b) clustering at informative viewpoints. (c) restricting variation along θ-axis. (d) denser sampling near the side area of a face.

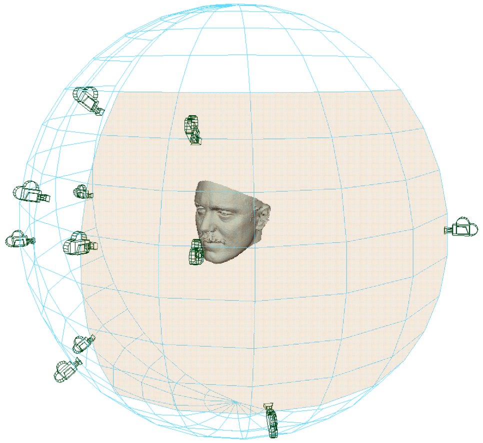

Figure 7: Arrangement of a set of 11 cameras in 3D; (b) in Figure 6. Shaded area indicates half the sampling range; $(0^\circ, 90^\circ)$ in azimuth and $(-40^\circ, 40^\circ)$ in elevation

effect of clutter through the design of a suitable cost function based on the boundary-weighted XOR error metric.

Figure 8 and Figure 10 show how our model face fits to real silhouette images of faces of Caucasian and Asian origin. With similar quality of alignment to the average synthetic case in Figure 5, these two diagrams indicate our boundary-weighted XOR cost function successfully attracts the model-generated silhouette contour to the boundary of input silhouette images. Note that this alignment cannot be achieved with a simple XOR-based cost function due to the lack of preference in matching direction.

Figure 9 and Figure 11 demonstrate the effectiveness of 3D reconstruction and subsequent texture mapping of the Caucasian and Asian model heads in Figure 8 and Figure 10 respectively. Note that the location of eyes and the shape of noses and lips in the texture mapped images agree well with the reconstructed 3D geometry. It is remarkable that the race information, which is expected to be coupled with silhouette contour, was successfully captured by our silhouette matching scheme.

6 Conclusion and Future Work

In this paper we present a method to reconstruct faces from silhouette projections. In experiments with synthetic and real data, we demonstrate that 2D silhouette matching in the various viewpoints captures the most important 3D features of a human face for reconstruction. The number and locations of cameras play an important role in the quality of silhouette-based shape recovery. We plan to devise a systematic way to obtain a set of maximally informative viewpoints for capturing geometry of a human face.

Our methods are almost automatic. Very little user interaction is required. User intervention is needed for picking feature points on laser-scanned face surfaces to obtain point correspondences for model building. Other interaction is needed for picking feature points in input photographs during the alignment stage. Both steps can be automated with robust feature point detection algorithms for color images, which will make the proposed system fully automatic.

Finally, we developed a formulation to find optimal model parameters which provide best fit to the given silhouette images. The advantage of this scheme is that the silhouette-based cost function is robust and easy to compute. In particular, our formulation works well in the case that the model matches only partial areas of the input silhouette images. The proposed cost function provides high fidelity in the reconstructed 3D faces but is not amenable to the computation of gradient information in terms of model parameters. Our work provides a robust and efficient solutions to reconstructing the human face. The separate treatment of geometry and texture will enable the pursuit of even more robust and efficient algorithms for the various stages of the reconstruction process.

In the future, we plan to use differentiable cost functions for better convergence rate. Additionally, it will be worthwhile to also consider methods based on Monte-Carlo Markov Chains for efficient implementation.

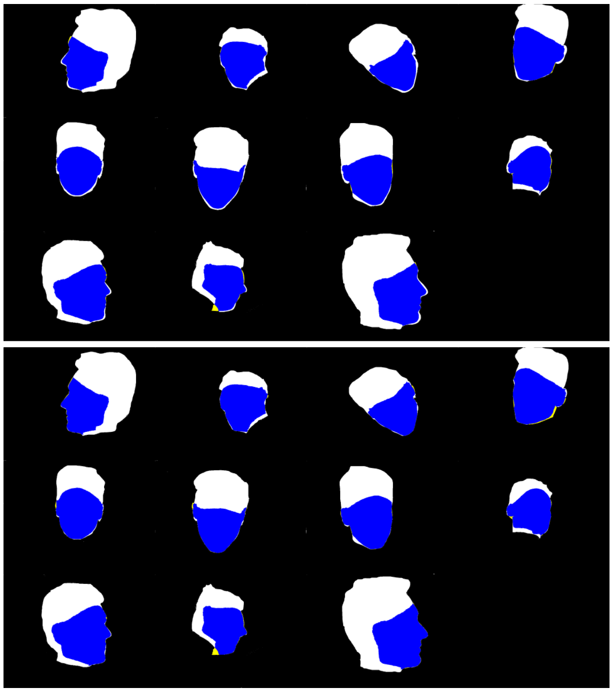

Figure 8: Difference in real silhouette images of a Caucasian head model before (above) and after (below) optimization.

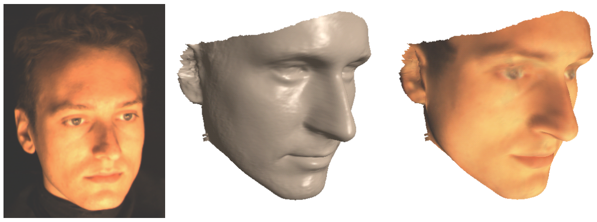

Figure 9: 3D reconstruction of the Caucasian model of Figure 8 shown in a novel viewpoint (left image is one of the 11 input (real) images).

Acknowledgements

We thank Tim Weyrich and Addy Ngan for setting up the cameras to acquire their silhouette and texture images. We are also grateful to Prof. Sudeep Sarkar, Univ. of South Florida, for allowing us to use the USF DARPA HumanID 3D Face Database for this research.

References

[1] J. J. Atick, P. A. Griffin, and N. Redlich. Statistical Approach to Shape from Shading: Reconstruction of 3D Face Surfaces from Single 2D Images. *Neural Computation*, Vol. 8, No. 6, pages 1321-1340, 1996.

[2] V. Blanz and T. Vetter. A Morphable Model for the Synthesis of 3D Faces. In *Proceedings of SIGGRAPH 99*, July 1999.

[3] I. Craw and P. Cameron. Face Recognition by Computer. In *British Machine Vision Conference 1992*, pages 498-507, 1992.

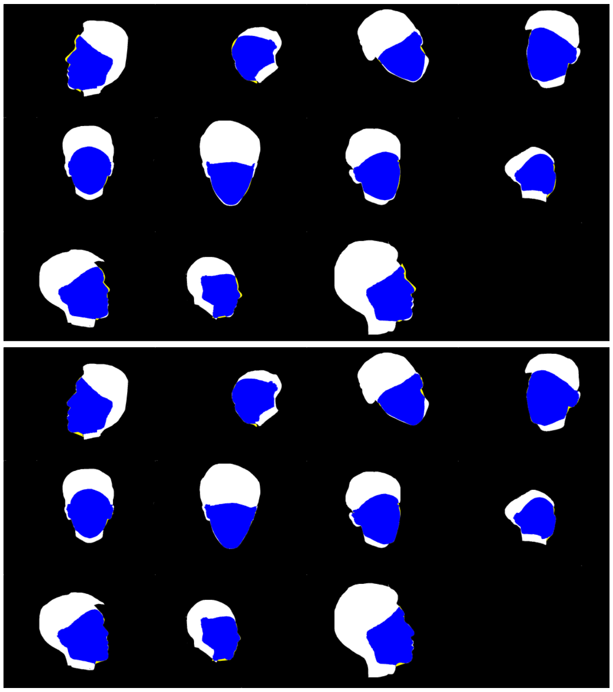

Figure 10: Difference in real silhouette images of an Asian model before (above) and after (below) optimization.

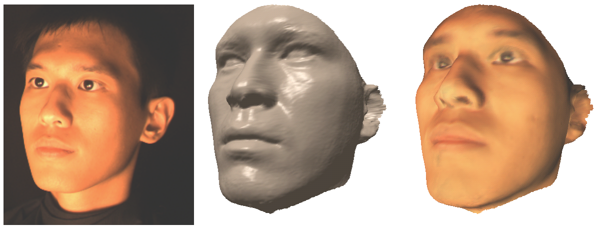

Figure 11: 3D reconstruction of the Asian model of Figure 10 shown in a novel viewpoint (left image is one of the 11 input (real) images).

[4] Cyberware, Inc., Monterey, CA. URL: http://www.cyberware.com/

[5] I. L. Dryden and K. V. Mardia. *Statistical Shape Analysis*. John Wiley & Sons, New York, 1998.

[6] A. A. Grattarola. Volumetric Reconstruction from Object Silhouettes: A Regularization Procedure. *Signal Processing*, Vol. 27, No. 1, pages 27-35, 1992.

[7] B. Guenter, C. Grimm, D. Wood, H. Malvar, and F. Pighin. Making Faces. In *Proceedings of SIGGRAPH 98*, pages 5566, July 1998.

[8] T. Jebara, K. Russell, and A. Pentland. Mixtures of Eigenfeatures for Real-Time Structure from Texture. In *Proceedings of ICCV '98*, Bombay, India, January, 1998.

[9] I. T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, New York, 1986.

[10] S. Lazebnik and E. Boyer and J. Ponce. On Computing Exact Visual Hulls of Solids Bounded by Smooth Surfaces. *Computer Vi-*

sion and Pattern Recognition (CVPR'01), Vol I, pages 156-161, December 2001.

[11] Y. Lee, D. Terzopoulos, and K. Waters. Realistic Modeling for Facial Animations. In *Proceedings of SIGGRAPH 95*, pages 5562, August 1995.

[12] W. S. Lee and N. Magnenat Thalmann. Fast Head Modeling for Animation. *Image and Vision Computing*, Vol. 18, No. 4, pages 355364, March 2000.

[13] H. P. A. Lensch, W. Heidrich, and H. Seidel. Automated Texture Registration and Stitching for Real World Models. In *Proceedings of Pacific Graphics '00*, October 2000.

[14] W. Matusik, C. Buehler, R. Raskar, L. McMillan, and S. J. Gortler. Image-Based Visual Hulls. In *Proceedings of SIGGRAPH 00*, July 2000.

[15] W. Matusik, H. Pfister, P. A. Beardsley, A. Ngan, R. Ziegler, and L. McMillan. Image-Based 3D Photography Using Opacity Hulls. In *Proceedings of SIGGRAPH 02*, July 2002.

[16] G. M. Nielson. Scattered Data Modeling. *IEEE Computer Graphics and Applications*, Vol. 13, No. 1, pages 60-70, January 1993.

[17] A. J. O'Toole, H. Abdi, K. A. Deffenbacher, and D. Valentin. Low-dimensional representation of faces in higher dimensions of the face space. *Journal of the Optical Society of America*, Vol. 10, No. 3, pages 405-411, March 1993.

[18] A. Pentland, B. Moghaddam, and T. Starner. View-Based and Modular Eigenspaces for Face Recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, 1994.

[19] F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, and D. Salesin. Synthesizing Realistic Facial Expressions from Photographs. In *Proceedings of SIGGRAPH 98*, July 1998.

[20] M. Potmesil. Generating Octree Models of 3D Objects from their Silhouettes in a Sequence of Images. *CVGIP* 40, pages 1-29, 1987.

[21] W. H. Press, B. P. Flannery, S. A. Teukolosky, and W. T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, New York, 1988.

[22] P. Ramanathan, E. Steinbach, and B. Girod. Silhouette-based Multiple-View Camera Calibration. In *Proceedings of Vision, Modeling and Visualization 2000*, pages 3-10, Saarbruecken, Germany, November 2000.

[23] C. Rocchini, P. Cignoni, C. Montani, and R. Scopigno. Multiple Textures Stitching and Blending on 3D Objects. In *Rendering Techniques '99 (Proc. 10th EG Workshop on Rendering)*, pages 119-130, 1999.

[24] Y. Shan, Z. Liu, and Z. Zhang. Model-Based Bundle Adjustment with Application to Face Modeling. In *Proceedings of ICCV 01*, pages 644-651, July 2001.

[25] L. Sirovich and M. Kirby. Low dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A.*, 4:519-524, 1987.

[26] C. Sminchisescu. Consistency and Coupling in Human Model Likelihoods. *IEEE International Conference on Automatic Face and Gesture Recognition*, May 2002.

[27] R. Szeliski. Rapid Octree Construction from Image Sequences. *CVGIP: Image Understanding*, Vol. 58, No. 1, pages 23-32, 1993.

[28] R. Szeliski and S. Kang. Recovering 3D Shape and Motion from Image Streams Using Non-Linear Least Squares. Technical Report, Robotics Institute, Carnegie Mellon University, March, 1993.

[29] USF DARPA HumanID 3D Face Database, Courtesy of Prof. Sudeep Sarkar, University of South Florida, Tampa, FL.

[30] M. Tarini, H. Yamauchi, J. Haber, and H.-P. Seidel. Texturing Faces. In *Proceedings Graphics Interface 2002*, pages 89-98, May 2002.

[31] M. Turk and A. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, Vol. 3, No. 1, 1991.

[32] T. Vetter and V. Blanz. Estimating Coloured 3D Face Models from Single Images: An Example Based Approach. In *Computer Vision - ECCV '98*, Vol II, Freiburg, Germany, 1998.

[33] J.Y. Zheng. Acquiring 3D Models from Sequences of Contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, No. 2, February 1994.