3D Face Mesh Modeling from Range Images for 3D Face Recognition

A-Nasser Ansari, Mohamed Abdel-Mottaleb, and Mohammad H. Mahoor

University of Miami, Department of Electrical & Computer Engineering
1251 Memorial Drive, Coral Gables, FL 33124
E-mails: a.alansari@umiami.edu, mottaleb@miami.edu, mmahoor@umsis.miami.edu

ABSTRACT

We present an algorithm for 3D face deformation and modeling using range data captured by a 3D scanner. Using only three facial feature points extracted from the range images and a 3D generic face model, the algorithm first aligns the 3D model to the entire range data of a given subject's face. Then each aligned triangle of the mesh model, with three vertices, is treated as a surface plane which is then fitted to the corresponding interior 3D range data, using least squares plane fitting. Via triangular vertices subdivisions, a higher resolution model is generated from the coordinates of the aligned and fitted model. Finally the model and its triangular surfaces are fitted once again resulting in a smoother mesh model that resembles and captures the surface characteristic of the face. Application of the final deformed model in 3D face recognition, using a publicly available database, shows promising results.

Index Terms— 3D face modeling, face recognition.

1. INTRODUCTION

3D range data is rich yet making full use of its high resolution for face recognition is very challenging. It is difficult to extract numerously reliable facial features in 3D. As a result, it becomes more challenging and computationally expensive to accurately match two sets of 3D data (e.g., matching a subject's probe data and the gallery's). Our objective in this research is to represent the 3D data of a given subject by a deformed 3D mesh model using a minimum number of extracted features. The deformation processes of the model essentially capture the surface characteristics of the data and represent the face with a massively reduced amount of data points, i.e., the mesh model's vertices.

We briefly review algorithms in the literature, specifically targeting those relevant to our approach. We can broadly classify 3D face modeling for recognition into three categories. Namely, 3D matching, representative domain, and model based approaches. A matching method, known as Iteratively Closest Point (ICP) approach, is becoming popular and is often used as a necessary step in aligning or matching the datasets of two subjects [1,2]. ICP is based on the search of pairs of nearest points in the two datasets and estimation of the rigid transformation that aligns them. Then the rigid transformation is applied to the points of one set and the procedure is iterated until convergence. Hausdorff distance is another matching approach which is often used in conjunction with ICP [3,4]. Hausdorff distance attempts to match two datasets based on subset points from the datasets. The problems with these two approaches are expensive computations and sometimes fail to give accurate results. The main reason for using ICP or Hausdorff is not having direct correspondences between the two compared datasets. In our work, the two compared datasets have direct feature correspondences, which eliminates the need for the above alignment/matching algorithms. Other researchers attempted to represent the 3D data in a different domain and made recognition comparison in the representative domain. Examples of those are 3D PCA [13], shape index [2], point signature [11], spine image [10], and local shap map [12]. Model based approaches use a priori 3D face mesh model and morph it to a given face. In this regard, our system is considered a model based approach. A famous model based approach is that developed by Vetter et al. [14]. [14] developed a system to create 3D face models from a single image. However, their system initially requires 200 scanned face models and uses initial manual intervention to create the 3D models. [15] adapts a generic face model to the facial features extracted from both registered range and color images. The deformation iteratively moves the vertices of the mesh model using vertices displacement propagation. In our previous research [16], we deformed a generic model to range data obtained from two frontal and one profile view stereo images which required internally and externally calibrated cameras. In this paper, our method is similar to the algorithm by Chenghua et al. in mesh subdivisions [17], but differs in many ways as follows: (a) we use a generic face mesh model and [17] uses a mesh grid model, (b) we deform the aligned model's mesh triangles coordinate to the data and [17] simply align the grid mesh coordinates to the range data then copy the z coordinate at each x and y coordinates, hence in their way the pose of the z coordinate pose is not considered, (c) in our system, because we have labeled feature vertices, our method establishes direct correspondences with other models in the database, hence direct comparison is achieved in recognition, while [17] method has no correspondences for features and would require aligning all the tip of the nose of all models in the database to be at the origin of the 3D coordinate system. Their approach would require optimization for alignment and matching before recognition.

In our algorithm, we bypass the alignment of ICP and the matching of Hausdorff with their costly computations, and present an algorithm that deforms a predefined and labeled-vertices of a generic mesh model to a given subject's range image. Our approach for initially aligning the 3D model to the range data only requires the extraction of three salient facial features, namely; the tip of the nose and the two inner corners of the eyes. These features are the most easily extracted from 3D facial range images. A 3D model alignment and deformation are then used to fit the model to the range data. The end result is

a person specific deformed face model with vertices coordinates which are in correspondence with the other deformed models' vertices. Hence, direct alignment and recognition comparison can be easily achieved.

This paper is organized as follows: Section 2 explains the preprocessing and filtering of the range images along with the 3D facial features extraction algorithm. Section 3 introduces our 3D generic mesh model, its alignment and its deformation with experimental results. Section 4 presents results of applying our final 3D deformed models in 3D face recognition. Conclusions and future work are given in section 5.

2. DATA PREPROCESSING AND 3D FACIAL FEATURES EXTRACTION

This section explains the preprocessing of the data, localization of the facial region, and the facial features extraction. Further details are given in our published research in [19]. Range images, captured by laser scanners, have some artifacts, noise, and gaps. In the preprocessing step, we first apply median filtering to remove sharp spikes and noise, that occur during the scanning of the face, followed by interpolation to fill up the gaps, and low pass filtering to smooth the final surface. This is followed by face localization using facial template matching to discard the neck, hair, and the background areas of the range image. The facial range image template is correlated with the range images of a given face using normalized cross-correlation. We start by roughly detecting the location of the nose tip and then translate the template such that the detected tip of the nose is placed on the location of the nose tip of the range image under test. Afterward, we iteratively apply a rigid transformation to the template and cross-correlate the result with the subject's range image to find the best pose. Finally, the area underneath the template with the maximum correlation is considered as the localized facial region. Subsequently, we use Gaussian curvature to extract the two inner corners of the eyes and the tip of the nose. From [18], the surface that either has a peak or a pit shape has a positive Gaussian curvature value ($K > 0$). Each of the two inner corners of the eyes has a pit surface type and the tip of the nose has a peak surface type that is detectable based on the Gaussian curvature. These points have the highest positive Gaussian curvature values among the points on the face surface. Fig.1.a shows the result of calculating Gaussian curvature for one of the sample range images in the gallery.

Figure 1 Features extraction process (a) Gaussian curvature showing high values at the nose tip and eyes corners (b) Result of thresholding Fig.1.a (c) Final result of feature extraction.

The highest points in Fig.1.a correspond to the points with pit/peak shape. We threshold the Gaussian curvature to find the areas that have positive values greater than a threshold, producing a binary image. See Fig.1.b. This threshold is calculated based on a small training data set different from the images used in the recognition experiments. Finally, the three regions with the largest average value of Gaussian curvature are the candidate regions that include the feature points. The locations of the points with maximum Gaussian curvature in these regions are labeled as feature points. Fig.1.c shows a final result of the three feature extraction points. These features are used in the 3D model alignment as we show next.

3. 3D FACE MODELING

This section deals with molding the human face using its extracted features and a generic 3D mesh model. The idea is to align the 3D model to a given face using the extracted 3D features then proceed with fitting the aligned triangles of the mesh to the range data, using least square plane fitting. Next, the aligned triangles of the model are subdivided to higher resolution triangles, before applying a second round of plane fitting, to obtain a more realistic and a smoother fitted surface resembling the actual surface of the face. Fig.2.a shows our neutral 3D model with a total of 109 labeled feature vertices and 188 defined polygonal meshes. In addition, the model is designed such that the left and right sides of the jaw fall within but not on the edges of the face boundary. This approach avoids incorporating inaccurate data at the facial edges of the captured range images. We explain next the process of aligning the mesh model to the range data.

3.1. Global alignment

In the global alignment step, we rigidly align the 3D model using the 3D feature points, P_I, obtained from the range image, and their corresponding feature vertices, P_M, in the model. Subscripts I and M indicate image features and model vertices, respectively. To achieve this goal, the model must be rotated, translated, and scaled. Eq.1 gives the sum squared error between P_I and P_M in terms of scale S, rotation R, and translation T for N points.

$$Min \;\; E(S,R,T) = \sum_{j=1}^{N} \left\| P_{I_j} - P_{M_j} \right\|^2 \qquad (1)$$

An example of the aligned 3D model is demonstrated in Fig.2.b and Fig.2.c for 2D view and 3D view, respectively. As shown in the figures, the triangles of the model are buried either totally or partially above or below the 3D data. We show next how to fit and deform the model's triangles to be as closely as possible to the 3D data.

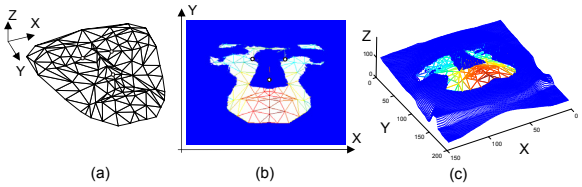

Figure 2 (a) 3D mesh model (b) 2D view of aligned 3D model vertices to the range data (c) 3D view of model to the range data.

3.2. 3D face model deformations

The first step in deforming the model is to extract the 3D data facing (above, below, or within) each triangle using barycentric coordinate [5]. The basic principle is that for any triangle with three vertices coordinates P_v=(X,Y,Z) given by P_1, P_2, and P_3, barycentric coordinate can determine whether any other point P lies inside or outside the perimeters of the triangle (subscript v indicates the vertex number). Once the cloud of the 3D data points is segmented by the barycentric coordinate, they are represented by a plane using least square fitting. The general

equation of a plane, with non-zero normal vector N, is defined in 3D as

$$aX + bY + cZ + d = 0 \text{, where } N = (a,b,c) \qquad (2)$$

For n number of points, Eq.2 can be written in least square form as

$$\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 \\ X_2 & Y_2 & Z_2 & 1 \\ : & : & : & : \\ X_n & Y_n & Z_n & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = AB = 0 \qquad (3)$$

where the coordinates (X_i,Y_i,Z_i)'s are those of all the data points segmented by the barycentric coordinate. Eq.3 can be solved for the plane equation parameters, $B = [a, b, c, d]$, which is then substituted in Eq.2, leading to a plane representing the 3D data points. Fig.3.a illustrates a concept example of a triangle with 3D data points in 3D space. Fig.3.b shows the segmented data within the triangle which are represented by a plane using Eq.2. From the mathematical geometry of a plane, having the parameters of B, any point on the plane can be evaluated. In our work, we deform each corresponding mesh triangle to the 3D data points, by first discarding the three vertices Z coordinates, evaluating the X and Y coordinates, and solving for the new Z coordinate (given the parameters in B from Eq.3). This produces a mesh triangle, with new depth coordinates, lying on the plane that is approximated by the dense 3D data points. Fig.3.c shows the concept of deforming the mesh triangle to the plane representing the data. Essentially, the pose of the triangle is changed to match that of the plane.

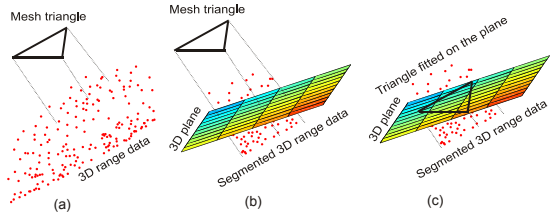

Figure 3 The process of deforming the triangles of the 3D mesh model, (a) Given cloud of 3D data and a mesh triangle (b) Segmenting the 3D data and plane fitting (c) Deforming the mesh triangle to the plane representing the 3D data.

Subsequently, we repeat the deformation process to all the triangles of the mesh model. Fig.4. shows an example of a complete deformed model superimposed on the data in 2D and 3D views. Comparing Fig.4.a-b with the initially aligned model of Fig.2.c-d, we see that the deformation and fitting of the model to the range data are clearly observed. The triangles of the mesh model have come closer to the data.

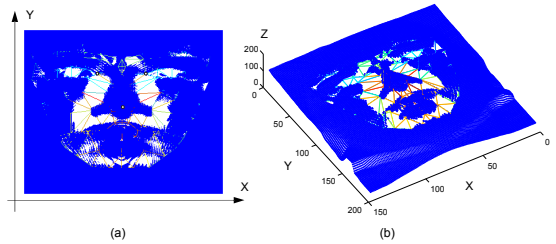

Figure 4 Deformed model superimposed on the range data. (a) 2D view (b) 3D view.

The deformed model of Fig.4 is a good representation of the data, yet it's not smooth enough to represent the high resolution and curvatures of the 3D data. In the next step, we subdivide the triangles of the model to a higher resolution in a manner shown in Fig.5.a. New vertices are computed based on the locations of the deformed vertices.

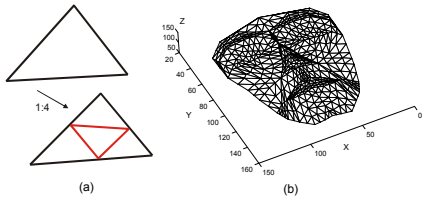

Figure 5 (a) 1 to 4 triangle subdivision (b) Result after applying triangle subdivisions to the deformed model of Fig.4.

Fig. 5.b shows the result of subdividing the deformed model of Fig.4. This increases the number of vertices and triangle meshes of the original model from 109 and 188, respectively, to 401 vertices and 752 polygonal meshes. Finally, because the new triangles do not reflect actual deformation to the data, we deform them once again using the same deformation process explained above. The introduction of smaller triangles gives more effective triangle fitting of the data especially at areas of high curvatures. Fig. 6.a-b-c show the final result of the deformed model, superimposed on the data, in 2D view, 3D view, and a profile 2D view, respectively. In Fig.6.a-b-c, because most of the models' vertices are embedded within the data, we use the "*" symbol to clearly show their locations. Fig.6.d shows a profile (YZ-axis) view of the model in Fig.6.c without the data. This deformed model, containing 401 vertices points, is the final representation of the facial data, which originally contained about 19,000 points (based on an average range image size of 150 by 130). This is nearly a 98 % data reduction. We show next the application of this model in 3D face recognition.

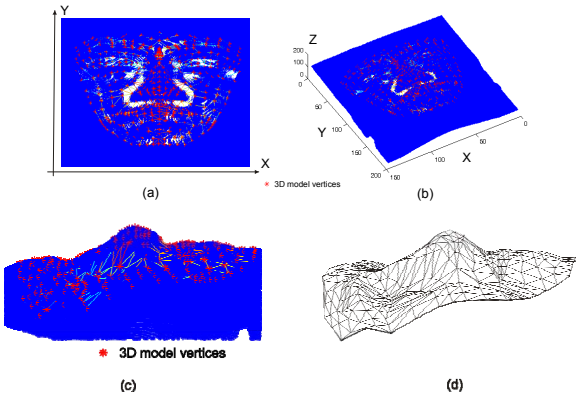

Figure 6 Final deformed model. (a) 2D view (b) 3D view (c) profile view superimposed on the data (d) mesh model without showing the data.

4. 3D FACE RECOGNITION

Face recognition has received great attention in the past few years. A recent literature survey of 3D face recognition is given in [9]. The final result of Fig.6 gives a model with 401 deformed vertices specific to the given subject's 3D range data. In this section we explore the use of the deformed final model in 3D face recognition using the GAVAB publicly available

database captured by a 3D scanner [8]. This database contains 61 subjects' range images with two frontal datasets per subject. In our recognition system, we use one set of the range data for the query and the second set for the database. For both sets we obtain the 3D face models as outlined in previous section. We test the recognition using a voting-based classier. The voting classier is a distance classifier that counts the maximum number of features in a gallery model that have minimum distances to the corresponding features of the probe. A face is recognized when it has the maximum number of votes. Fig.7 shows the Cumulative Match Characteristic (CMC) curve of our recognition system attaining 90.3 % rank one recognition rate. It has been reported that the same database was used in [6] achieving 78 % rank one recognition rate for 60 out of 61 subjects using 68 curvature-based extracted features. In our experiment six subjects out of the 61 were not correctly recognized. The wrong recognition was mainly due to dataset being either very noisy, incomplete, or the query range image set looks very different from the database set.

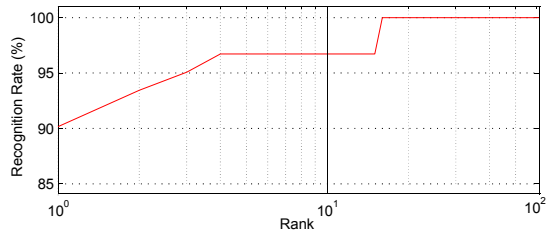

Figure 7 Cumulative Match Characteristic (CMC) curve.

Fig.8 shows two of the six subjects that were not recognized. Both query and database sets of Fig.8.a show noisy and incomplete facial scan at the left and right sides of the face. Fig.8.b shows similar incomplete data at the eye region. Our preprocessing step may deal well with noisy data but cannot cope with large areas of partially missing scans. In these areas the triangles of the model have no data for mesh fitting.

Figure 8 Query and database range images of two out of six misrecognized subjects.

5. CONCLUSION AND FUTURE WORK

We presented an algorithm for 3D face modeling using range data. The algorithm relies on deforming the triangular meshes of the model to the range data establishing direct model vertices correspondences with other deformed models in the database. These feature correspondences greatly facilitate faster computational time, accuracy, and recognition comparisons. By only detecting three facial features and a generic model, we achieved a 90.3 % rank one face recognition using a noisy database. In order to demonstrate the performance of our algorithm on a large scale dataset, we are in process of applying the algorithm to the Face Recognition Grand Challenge (FRGC) database [7]. In addition, we are researching reliable ways to incorporate facial expression in our deformation algorithm.

6. REFERENCES

[1] P. Besl and N. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.14, pp.239-256, 1992.

[2] X. Lu, A. Jain, D. Colbry, "Matching 2.5D face scans to 3D models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no.1, pp. 31-43, 2006.

[3] D. Huttenlocher, G. Klanderman, and W. Rucklidge, "Comparing images using the Hausdorff distance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 9, pp. 850–863, 1993.

[4] T. Russ, K. Koch, and C. Little, "A 2D range Hausdorff approach for 3D face recognition," *IEEE Workshop on Face Recognition Grand Challenge Experiments*, 2005.

[5] Coxeter, H. S. M, *Introduction to geometry*, 2nd ed. New York: Wiley, pp. 216-221, 1969.

[6] A. Moreno, Ángel Sánchez, J. Vélez, F. Díaz, "Face recognition using 3D surface-extracted descriptors," *The Irish Machine Vision and Image Processing Conference (IMVIP 2003)*, September 2003.

[7] P. Phillips, P. Flynn, T. Scruggs, K. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, "Overview of the face recognition grand challenge," *Proc. IEEE CVPR* , San Diego, pp. 947-954, June 2005.

[8] A. Moreno and A. Sanchez, "GavabDB: A 3D face database," *Proc. 2nd COST Workshop on Biometrics on the Internet: Fundamentals, Advances and Applications*, Vigo, Spain, March, 2004.

[9] K. Bowyer, K. Chang,, and P. Flynn, "A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition," *Computer Vision and Image Understanding.* vol. 101, no. 1, pp.1-15, 2006.

[10] A. Johnson, M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 21, no.5, pp.433-449, 1999.

[11] C. Chua, F. Han, Y. Ho, "3D human face recognition using point signature," *Proc IEEE International Conference on Automatic Face and Gesture Recognition*, p.233-238, 2000.

[12] Z. Wu, Y. Wang, G. Pan. "3D face recognition using local shape map," *Proc. IEEE International Conference on Image Processing*, vol. 3, pp. 2003- 2006, Oct. 2004.

[13] C. Hesher, A. Srivastava, G. Erlebacher, "A novel technique for face recognition using range images," *The Seventh International Symposium on Signal Processing and Its Application*, 2003.

[14] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3D faces," *Proc. Of Int. Conference on Computer Graphics*, pp.187-194, 1999.

[15]] R. L. Hsu, A. K. Jain, "Face modeling for recognition," *Proc. of IEEE ICIP,* Greece, Oct., 2001.

[16] A. Ansari, M. Abdel-Mottaleb, and M. Mahoor, "Disparity-based 3D face modeling using 3D deformable facial mask for 3D face recognition," *Proc. of IEEE ICME*, Toronto, Ontario, Canada, July 2006.

[17] Chenghua Xu, Yunhong Wang, and Tieniu Tan, "Three-dimensional face recognition using geometric model," *Proc. of the SPIE*, vol. 5404, pp. 304-315, August 2004.

[18] C. Dorai, A. Jain, "COSMOS- A representation scheme for 3D free-form objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 10, pp. 1115-1130, October, 1997.

[19] M. H. Mahoor and M. Abdel-Mottaleb, "3D face recognition based on 3D ridge lines in range data," *Proc. of IEEE ICIP*, San Antonio, Texas, September 16-19, 2007.