3D FACE RECOGNITION USING SHAPES OF FACIAL CURVES

Chafik Samir[1]

[1]MIIRE (LIFL UMR CNRS 8022)
GET/INT - Enic Telecom Lille 1
{chafik,daoudi}@enic.fr

Anuj Srivastava[2] *and Mohamed Daoudi*[1]

[2]Center for Applied Vision and Imaging
Sciences, Florida State University
anuj@stat.fsu.edu

ABSTRACT

Recognition of human beings using shapes of their full facial surfaces is a difficult problem. Our approach is to approximate a facial surface using a collection of (closed) facial curves, and to compare surfaces by comparing their corresponding curves. The differences between shapes of curves are quantified using lengths of geodesic paths between them on a pre-defined curve shape space. The metric for comparing facial surfaces is a composition of the metric involving individual facial curves. These ideas are demonstrated in the context of face recognition using the nearest-neighbor classifier.

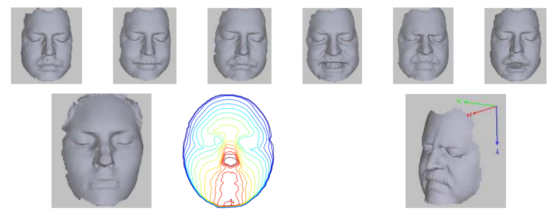

Fig. 1. Top: Examples of facial surfaces of a person under different facial expressions. Bottom left: Examples of facial curves C_λ for a surface S. Bottom right: A coordinate system attached to the face.

1. INTRODUCTION

Automatic face recognition has been actively researched in recent years, and various techniques using ideas from 2D image analysis have been presented. Although a significant progress has been made, the task of automated, robust face recognition is still a distant goal. 2D image-based methods are inherently limited by variability in imaging factors such as illumination and pose. An emerging solution is to use laser scanners for capturing surfaces of human faces, and use this data in performing face recognition [1]. Such observations are relatively invariant to illumination and pose, although they do vary with facial expressions. As the technology for measuring facial surfaces becomes simpler and cheaper, the use of 3D facial scans will be increasingly prominent. A measurement of a facial surface contains information about its shape and texture (more precisely, the reflectivity function). In general, one should utilize both the pieces of information for recognition. Given 3D scans of facial surfaces and textured images, the goal now is to develop metrics and mechanisms for comparing their shapes and textures.

Over the last few years, a number of approaches have emerged for comparing shapes of facial surfaces [2]. The earliest idea was to detect a set of feature locations – nose, nose bridge, eyes, lips, etc - in the face, and use their relative locations to characterize a face. The next idea was to generate range images from 3D scans and to utilize techniques from image analysis to recognize people. A more challenging problem is to recognize 3D face using the geometry of surface deformations [3], similar to the techniques developed for comparing shapes of anatomical objects [4]. Our approach is to derive *approximate* representations of facial surfaces, and to impose metrics that compare shapes of these representations. We exploit the fact that curves can be parameterized canonically, using the arc-length parameter, and thus can be compared naturally.

Rest of this paper is organized as follows: Section 2 describes a representation of a facial surface using a collection of facial curves, and presents metrics for comparing facial shapes under this representation. Section 3 describes the process of extracting facial curves from 3D face meshes. Section 4 presents some experimental results and shows the obtained recognition rate. We finish the paper with a brief summary in Section 5.

2. REPRESENTATION OF FACIAL SHAPES

Let S be a facial surface denoting a scanned face. Although in practice S is a triangulated mesh with a collection of edges and vertices, we start the discussion by assuming that it is a continuous surface. More precisely, it is an embedding of the upper unit hemisphere $\mathbb{S}^2_+$ in $\mathbb{R}^3$. In this definition, we have assumed that holes in S associated with eyes, nose, and mouth, are already patched. Some pictorial examples of S are shown in Figure 1 (top row) where facial surfaces associated with six facial expressions of the same person are displayed.

Let $F : S \mapsto \mathbb{R}$ be a continuous map on S. Let C_λ denote the level set of F, also called a **facial curve**, for the value $\lambda \in F(S)$, i.e. $C_\lambda = \{p \in S | F(p) = \lambda\} \quad \subset S$. We can reconstruct S through these level curves according to $S = \cup_\lambda C_\lambda$. Figure 1 (bottom left) shows some examples of facial curves along with the corresponding surface S. In principle, the collection $\{C_\lambda | \lambda \in \mathbb{R}_+\}$ contains all the information about S and one should be able to analyze shape of S via shapes of C_λs. In practice, however, a finite sampling of λ restricts our knowledge to a coarse approximation of the shape of S.

In this paper we choose F to be the **depth function**. Accordingly $F(p) = p_z$, the z-component of the point $p \in \mathbb{R}^3$. Our goal is to analyze shape of S invariant to action of the group of rigid motion $\mathbb{SE}(3) \equiv \mathbb{SO}(3) \ltimes \mathbb{R}^3$ on the surface S ($\ltimes$ implies a semi-direct product, that is the rotation is always applied before the translation). Let us investigate the variability of level sets of F with respect to these transformations. Rewrite $\mathbb{SE}(3)$ as $(\mathbb{SO}(2) \times \mathbb{S}^2) \ltimes (\mathbb{R}^2 \times \mathbb{R}^1)$, where we can interpret $\mathbb{SO}(2) \ltimes \mathbb{R}^2$ as a rigid motion in $x - y$ plane, i.e. perpendicular to the z axis, $\mathbb{S}^2$ as the direction of the z axis, and $\mathbb{R}$ as translation in z direction. We are assuming that $x - y - z$ axes form a *body centered Cartesian coordinate system*, so that z axis is aligned with the gaze direction, as shown in Figure 1 (bottom right). Our technique for comparing shapes of closed curves will be automatically invariant to planar transformations in $\mathbb{SO}(2) \ltimes \mathbb{R}^2$ and the z-translations in $\mathbb{R}$. However, we have no simple way of removing variability due to the change in z direction (or gaze direction) that varies over the $\mathbb{S}^2$; in general, one has to search over all rotations of a face in $\mathbb{S}^2$ to best align it with another face. In this paper, we avoid this search by using facial scans that are collected while the subjects were staring at the camera.

2.1. Comparing Shapes of Facial Curves

Consider facial curves C_λ as closed, arc-length parameterized, planar curves. Coordinate function $\alpha(s)$ of C_λ relates to the direction function $\theta(s)$ according to $\dot{\alpha}(s) = e^{j\,\theta(s)}$, $j = \sqrt{-1}$. To make shapes invariant to planar rotation, restrict to angle functions such that, $\frac{1}{2\pi}\int_0^{2\pi} \theta(s)ds = \pi$. Also, for a closed curve, θ must satisfy the *closure condition*: $\int_0^{2\pi} \exp(j\,\theta(s))ds = 0$. Summarizing, one restricts to the set $\mathcal{C} = \{\theta | \frac{1}{2\pi}\int_0^{2\pi} \theta(s)ds = \pi,\ \int_0^{2\pi} e^{j\theta(s)}ds = 0\}$. To remove the re-parameterization group $\mathbb{S}^1$ (relating to different placements of origin, point with $s = 0$, on the same curve), define the quotient space $\mathcal{D} \equiv \mathcal{C}/\mathbb{S}^1$ as the shape space.

Let C^1_λ and C^2_λ be two facial curves associated with two different faces but at the same level λ. Let θ_1 and θ_2 be the angle functions associated with these curves, respectively. An important tool in a Riemannian analysis of shapes is to construct geodesic paths between shapes and to use geodesic lengths as shape metric. The paper [5] provides a numerical procedure for computing geodesics between arbitrary points in $\mathcal{D}$. For any two shapes $\theta_1, \theta_2 \in \mathcal{D}$, they use a *shooting method* to construct the geodesic between them. The basic idea is search for a tangent direction g at the first shape θ_1, such that a geodesic in that direction reaches the second shape θ_2 in unit time. This search is performed by minimizing a "miss function", defined as a $\mathbb{L}^2$ distance between the shape reached and θ_2, using a gradient process. The geodesic is with respect to the metric $\langle g_1, g_2\rangle = \int_0^{2\pi} g_1(s)g_2(s)ds$. This choice implies that a geodesic between two shapes is the path that uses **minimum energy to bend** one shape into the other. Shown in Figure 2 are some examples of geodesic paths between corresponding facial curves of two different facial surfaces. Let $d(C^1_\lambda, C^2_\lambda)$ denote the length of geodesic connecting their representatives, θ_1 and θ_2, in the shape space $\mathcal{D}$. This distance is independent of rotation and translation in the $x-y$ plane, and 3D scaling of the facial surfaces.

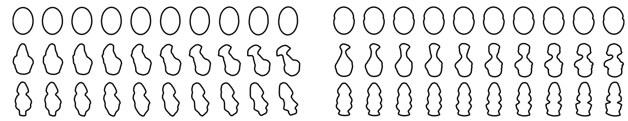

Fig. 2. Geodesic paths between different facial curves.

2.2. Metric for Comparing Facial Shapes

Now that we have defined a metric for comparing shapes of facial curves, it can be easily extended to compare shapes of facial surfaces. Assuming that $\{C^1_\lambda | \lambda \in \Lambda\}$ and $\{C^2_\lambda | \lambda \in \Lambda\}$ be the collections of facial curves associated with the two surfaces, two possible metrics between them are:

$$d_g(S^1, S^2) = \left(\prod_{\lambda \in \Lambda} d(C^1_\lambda, C^2_\lambda)\right)^{1/|\Lambda|} \tag{1}$$

$$d_e(S^1, S^2) = \left(\sum_{\lambda \in \Lambda} d(C^1_\lambda, C^2_\lambda)^2\right)^{1/2} \tag{2}$$

d_e denotes the Euclidean length and d_g denotes the geometric mean. Here Λ is a finite set of values used in approximating a facial surface by facial curves.

The choice of Λ is also important in the resulting performance. Of course, the accuracy of d_e and d_g will improve with increase in the size of Λ, but the question is how to choose the elements of Λ. In this paper, we have sampled the range of depth values uniformly to obtain Λ.

3. DATA COLLECTION AND CURVE EXTRACTION

In this section we describe the process of extracting facial curves from the scanned 3D meshes. Our general approach is to convert 3D scanned meshes into range images, and then use techniques from standard image analysis to extract level curves.

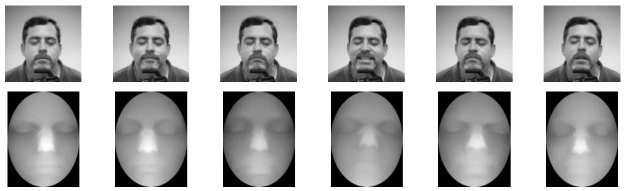

Fig. 3. Data capture: each subject was scanned for six different facial expressions. Range images of six different subjects under the same facial expression.

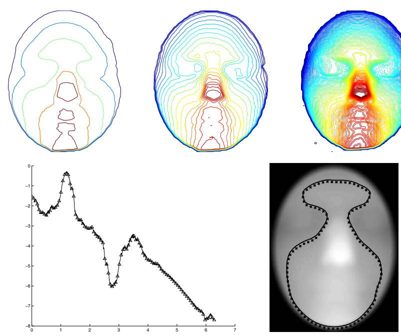

Fig. 4. Top: Level sets of depth function for several levels. Bottom: Angle functions, observed (marked) and fitted (solid), for a level curve of the depth function in the range image shown in right.

Range Image Generation: In this work we used GTS (GNU Triangulated Surface Library) to remove holes and to refine the original mesh, the resulting 3D model will have a high resolution, which allow us to generate high resolution range images, and to ensure smoothness of extracted level curves. We also crop the range images near the extremities of the image, using a standard masking image. Shown in Figure 3(bottom) are some examples of the resulting range images.

Level Curve Extraction: Extraction of level sets, of functions defined on 3D meshes, is not straightforward. Following steps summarize the extraction of level curves from range images: (i) Smooth the range image using a Gaussian filter, to help improve the extraction performance. Such smoothing does not significantly change the shape of resulting facial curves, and therefore is a valid step in pre-processing data. (ii) Extract pixel locations at a certain range, say λ_0. (iii) Interpolate between the extracted points to build a continuous curve, closed curve. Some examples of extracted level curves are shown in Figure 5.

Angle Function Representation: For an observed contour, denoted by an ordered set of non-uniformly sampled points in $\mathbb{R}^2$, one can generate a representative element $\theta \in \mathcal{D}$ as follows. For each neighboring pair of points, compute the chord angle θ_i and the Euclidean distance s_i between them. Then, fit a smooth θ function, e.g. using splines, to the graph

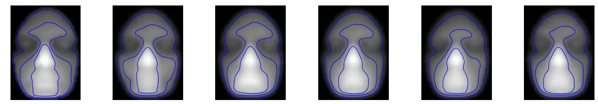

Fig. 5. Three level sets in each surface. six facial expressions, same subject.

formed by $\{(\sum_{j=1}^{i} s_j, \theta_i)\}$. Finally, re-sample θ uniformly (using arc-length parametrization) and project onto $\mathcal{D}$ using techniques described in [5]. Shown in Figure 4 right panel is an example. In the left panel, we show the original graph $\{(\sum_{j=1}^{i} s_j, \theta_i)\}$ in marked line, and a smooth fitted function in solid line. The corresponding facial curves are shown in the right panel, superimposed on the original range image. Figure 5 shows some more examples of the fitted smooth curves.

4. EXPERIMENTAL RESULTS

In this section we present some experimental results to demonstrate effectiveness of our approach. First, to demonstrate the success d_e and d_g by presenting a matrix of pairwise distances between a small set of faces. This matrix shows that faces for the same people are closer than faces of different people. We further emphasize that idea using a simple clustering example. We cluster a small number of facial surfaces using dendrogram clustering, and demonstrate that facial surfaces of sample people are clustered together, despite having different facial expressions. The original range images generated were of size 1201×900, but we down-sampled them to 376×449. From each range image we extract 1,2,..., 5 curves, depending upon the experiment, and used the aforementioned geodesic program to compute pairwise distances d_e and d_g.

4.1. Distances Matrices

In the first experiment, we considered 60 faces (six facial expression each for ten persons). These faces are labelled in order, i.e. 1-6 for person 1, 7-12 for person 2, etc. We computed the distance $d(S^1, S^2)$ for each pair and the results are shown in Figure 6(top). The left panel uses d_e while the right panel uses d_g. To improve the display we have truncated the values above a certain threshold. (If the pairwise distance exceeds a certain value, we have set that pixel to be white, while smaller distances are denoted by darker pixels.) It is easy to see that both d_e (left) and d_g (right) are successful in imposing smaller distances between different face scans of the same person.

To further demonstrate these metrics, we have performed a dendrogram clustering of faces using pairwise distances. In this experiment we used restricted to 30 facial surfaces associated with five people (as dendrogram becomes crowded with more data points). Shown in Figure 6(middle) are resulting

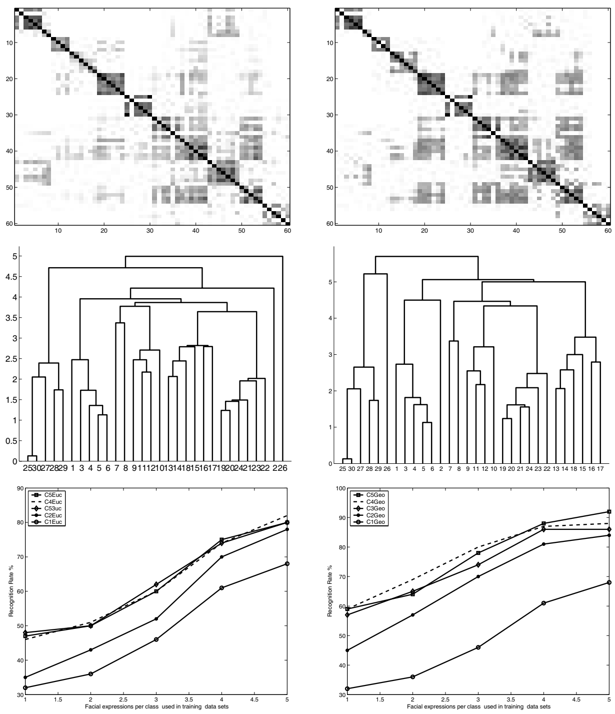

Fig. 6. (Top): Pairwise distances between 60 facial surfaces. (Middle): Dendrogram clustering between facial surfaces, indexed 1 to 30. (Bottom): Recognition rate plotted versus training data size per person.

dendrograms for two metrics. These graphs show that faces of same people have been successfully clustered together.

4.2. Nearest Neighbor Recognition

In this experiment we used a total of 300 facial surfaces (six facial expressions each for 50 persons). We divide this set into training and test sets by taking r ($r = 1, 2, \ldots, 5$) faces per person as labelled training data, and the remaining $6 - r$ faces/person as test data. Then, we use nearest neighbor classifier and the distance $d(S^1, S^2)$ to classify each test face. Since we know the true class of the test face, we can compute the percentage of correctly classified test faces. This recognition performance was studied by varying the setup as follows: first, we computed the recognition performance by changing r, the number of training faces. Shown in Figure 6(bottom) are results of this experiment. Each evolution denotes a different number of curves used (in coarsely approximating a facial surface). For example, $C1$ denotes the recognition performance obtained when only one curve was to represent a facial surface ($|\Lambda = 1|$). Similarly, $C2$ uses two curves, $C3$ uses three curves and so on. The left figure is for d_e and the right figure is for d_g. $CnEuc$ is the performance for Euclidean average, while $CnGeo$ is that for the geometric mean.

The recognition performance steadily increases with increase in r. This result is intuitive as more training data generally implies a better classification performance. Another interesting point is that for a fixed r, the recognition performance initially increases with the number of curves.

The performance of nearest neighbor recognition strongly supports the idea of using geometries facial curves to recognize people. Even with one curve per face, and one training data, we achieve more than 30% recognition rate. Remember that uniform sampling will result in only $1/50 = 2\%$ recognition rate. Using five training faces per person, and five facial curves per face, we can achieve a recognition rate of almost 92%. Considering that we perform a coarse sampling of a facial surface and that we completely ignore the surface textures, this rate is quite significant and points to the possibility of practical 3D shape-based face-recognition systems.

5. SUMMARY

A new surface descriptor is described for comparing shapes of facial surfaces via the shapes of facial curves. The basic idea is to coarsely approximate a facial surface S with a finite set of level curves, called the facial curves, of the height function on S. Curve extraction is accomplished using range images, and metric between facial curves are computed using an earlier method [5]. A metric on shapes of facial surfaces is derived by accumulating distances between corresponding facial curves. Results are presented from clustering and recognition of facial surfaces according to this metric.

6. REFERENCES

[1] Kyong I. Chang, Kevin W. Bowyer, and Patrick J. Flynn, "An evaluation of multimodal 2d+3d face biometrics.," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 4, pp. 619–624, 2005.

[2] P.W. Hallinan, G. Gordon, A. L. Yuille, P. Giblin, and D. Mumford, *Two- and Three-Dimensional Patterns of Face*, A. K. Peters, 1999.

[3] A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Three-dimensional face recognition," *International Journal of Computer Vision*, vol. 64, no. 1, pp. 5–30, 2005.

[4] Michael I. Miller and Laurent Younes, "Group actions, homeomorphisms, and matching: A general framework.," *International Journal of Computer Vision*, vol. 41, no. 1/2, pp. 61–84, 2001.

[5] E. Klassen, A. Srivastava, W. Mio, and S. Joshi, "Analysis of planar shapes using geodesic paths on shape spaces," *IEEE Pattern Analysis and Machine Intelligence*, vol. 26, no. 3, pp. 372–383, 2004.