Face Recognition From Range Images Using Point Signature

Feng Han, Chin-Seng Chua, Yeong-Khing Ho
School of Electrical and Electronic Engineering
Nanyang Technological University, Singapore 639798
P145135465@ntu.edu.sg

Abstract

In this paper, we present a novel face recognition algorithm based on the Point Signature - a representation for free-form surfaces. We treat the face recognition problem as a non-rigid 3D object recognition problem, which enables our algorithm to correctly recognize faces, despite they having different facial expressions. The rigid regions of the face of one person are extracted after registering the range data sets of faces having different facial expressions from the same person. These rigid regions are used to create a model library for efficient indexing. For a given test face, models are indexed from the library and the most appropriate models are ranked according to their similarity with the test face. Verification of each model proceeds according to their ranking. In this way, the correct model face can be quickly and efficiently identified. Experimental results with range data involving 18 human subjects, each with five different facial expressions, have demonstrated the validity and effectiveness of our algorithm.

1 Introduction

Automatic identification of human faces is a very challenging research area, which has gained much attention during the last few years. Most of this work, however, is focused on intensity or color images of faces [3]; only a few approaches deal with range images [8, 9, 1, 7]. However, there is evidence that range images have the potential to overcome some problems of intensity and color images. Some advantages of range images are the explicit representation of 3D shape, which is invariant under the change of color and reflectance properties of the objects.

Many researchers have handled the 3D-face recognition problem using differential geometry tools of computing of curvatures [8, 9, 7]. Lee and Milios [8] extracted the convex regions of human faces by segmenting the range images based on the signs of the mean and Gaussian curvature at each point. These regions form the basic set of features of a face. For each of these convex regions the Extended Gaussian Image is extracted which is then used to match the facial features of the two face images. Gordon [7] presented a detailed study of face recognition based on depth and curvature features. The curvatures on a face are calculated to find face-specific descriptors (e.g. nose ridge and eye features). Comparison between two faces is made based on their relationship in the feature space. Hiromi *et. al* [9] treated the face recognition problem as a 3D shape recognition problem of rigid free-form surfaces. Each face in both input images and model database, is represented as an Extended Gaussian Image (EGI), constructed by mapping principal curvatures and their directions at each surface points, onto two unit spheres, each of which represents ridge and valley lines respectively. Individual faces are then recognized by evaluating the similarities among others using Fisher's spherical correlation on EGI's of faces. Bernard *et. al* [1] extended the two face recognition approaches, which are based on grey level images, to deal with range images. These two approaches are based on eigenfaces and Markov models respectively.

However, all the research work focused on face surfaces with neutral facial expression and thus is based on the assumption that the shape of faces doesn't change significantly. Considering the fact that the shape of many parts of the face will change considerably (see Figure 1) due to some facial expressions, the face surface recognition problem should be treated as a 3D recognition problem of non-rigid surfaces if the face surface showing expressions can be correctly recognized. New methods are required to handle this difficult problem. In this paper we propose the use of Point Signature [6] to recognize non-rigid objects. This is an extension of its use in [6], which uses Point Signature to recognize 3D rigid objects. In particular, we are concerned with the identification of faces, despite they having different facial expressions.

While the face-shape of the same person may change, sometimes greatly, due to different facial expressions, there will still be regions, such as nose, eye socket region and forehead, which will keep their shape and position or be

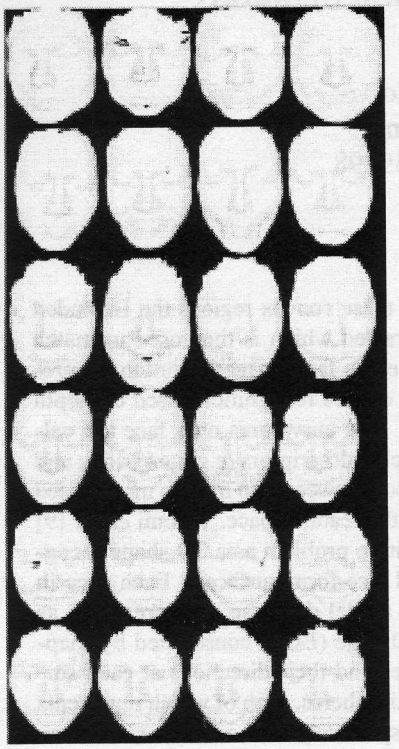


Figure 1. Rendered 3D faces of eight human subjects, each with four facial expressions.

subjected to much less deformation between any different expression. If these regions can be identified,the 3D non-rigid face recognition problem can be reduced to the rigid case. Based on this observation, our approach is to extract the rigid parts of the face and utilize them to realize the task of recognition. In this way, the greatly changing facial regions can be rejected and thus will not affect the recognition result. Moreover, only using the most stable and meaningful parts of the face will compress the information stored in the model library, which undoubtedly will improve the speed of recognition considerably.

This paper is organized as follows. Section 2 discusses the registration of two face-surfaces with different facial expressions using Point Signature. The optimal registration should be based only on the "rigid" regions of the face, which are not known *a priori*. At the same time, it is from this registration that the "rigid" regions of the face are derived. This section will propose an algorithm to solve this "catch-22" problem. Having established the approach to optimally register face-surfaces, Section 3 presents the methodology of recognizing a face from a library of face-subjects. In this section, the build-up of the face-library and the indexing of a candidate face is discussed. The efficiency of the algorithm is verified with test results and this is presented in Section 4. The tests are based on real range data, sampled from 18 different faces of human subjects.

2 Face-Surface Registration Using Point Signature

To extract the rigid regions of one person's face surface, we register the face surfaces with different expressions of this person to the face surface with a neutral expression of the same person, which is used as a reference face surface. From the registration the rigid regions are identified as containing range points with low registration errors. Conversely, the non-rigid, or flexible, regions will have range points of high registration errors. The non-rigid regions identified from each registration are rejected from the reference face surface. The remaining regions of the reference face surface are identified as the rigid regions and used to build the library of the face-surfaces which will be discussed in Section 3. Compared with the feature extraction, this method is more robust and reliable when dealing with the natural objects, like human faces, in noisy images.

2.1 Definition of point signature

The definition of the Point Signature is summarized here for completeness. For details, the reader may refer to [6].

For a given point p, we place a sphere of radius r, centered at p. The intersection of the sphere with the object surface is a 3D space curve, C, whose orientation can be defined by an orthonormal frame formed by a normal vector, $\mathbf{n}_1$, a "reference" vector, $\mathbf{n}_2$, and the vector cross-product of $\mathbf{n}_1$ and $\mathbf{n}_2$. $\mathbf{n}_1$ is defined as the unit normal vector of a plane fitted through the space curve C which need not be planar; the planar fit is used as an approximation to provide a vector direction. A new plane $\mathbf{P}'$ is defined by translating the fitted plane to the point p in a direction parallel to $\mathbf{n}_1$. The perpendicular projection of C to $\mathbf{P}'$ forms a new planar curve C' with the projection distance of points on C' forming a signed distance profile. The reference direction, $\mathbf{n}_2$, is defined as the unit vector from p to the projected point on C' which gives the largest positive distance.

Every point on C may now be characterized by :

1. The signed distance from itself to the corresponding point on C.

2. A clockwise rotation angle θ about $\mathbf{n}_1$ from the reference direction $\mathbf{n}_2$.

We refer to this profile of distances, $d(\theta)$, with $0 \leq \theta \leq 360$ degrees, as the signature at point p. Practical consideration, however, do not permit us to consider all points on the space curve C but rather an angular sampling of points starting from the reference direction $\mathbf{n}_2$. Depending on the

resolution of the input range data, the angular sampling interval, $\triangle\theta$, may vary from 5 to 15 degrees. (In our implementation $\triangle\theta = 10$ is used). Hence, the distance profile may now be represented by a discrete set of values $d(\theta_i)$ for $i = 1, \ldots, n_\theta$, $0 \leq \theta_i \leq 360$, where n_θ is the number of samples.

2.2 Matching of point signatures

The similarity matching between a given signature $d_s(\theta_i)$ and a candidate signature $d_m(\theta_i)$ is as follows:

If the signature $d_s(\theta_i)$ has a global maximum that is significantly different from the other local maximum, then $d_s(\theta_i)$ would be uniquely oriented by the reference vector $\mathbf{n}_2$. In this case $d_s(\theta_i)$ can be directly compared with the candidate signature $d_m(\theta_i)$ using the following constraint:

$$|d_s(\theta_i) - d_m(\theta_i)| \leq \varepsilon_{tol}(\theta_i)\ \forall i = 1, \ldots, n_\theta \qquad (1)$$

where the tolerance band $\varepsilon_{tol}(\theta_i)$ is used to handle the errors in computing Point Signature and achieve a better acceptance and rejection of candidate signatures.

However, if $d_s(\theta_i)$ has more than one local maximum which are similar, then for each position of local maximum, $d_s(\theta_i)$ is phase shifted before comparison is carried out at that position:

$$|d_s(\theta_i - \theta_j) - d_m(\theta_i)| \leq \varepsilon_{tol}(\theta_i)\ \forall i = 1, \ldots, n_\theta, \quad \theta_j \in \text{position of local discrete maximum} \qquad (2)$$

2.3 Registering two face-surfaces with different expressions

Given two face surface data sets, $S = \{s_i : i = 1, \ldots, n_s\}$ and $M = \{m_j : j = 1, \ldots, n_m\}$, the task of 3D registration is to discover the transformation (translation and rotation) which will optimally align the rigid regions of S with those of M. The two face shapes need not be identical, which makes this registration different from the traditional one [2, 5, 10, 6, 4].

First, we select three points s_α, s_β and s_γ in S. Next, three points m_α, m_β and m_γ in M are hypothesized to correspond to s_α, s_β and s_γ. If the hypothesis is acceptable, the transformation from the 3-tuple $\langle s_\alpha, s_\beta, s_\gamma \rangle$ in S to the 3-tuple $\langle m_\alpha, m_\beta, m_\gamma \rangle$ in M represents a possible registration solution. If this registration meets the need of certain criteria, registration is achieved with measurable quality. The details are discussed below.

A) Selecting s_α, s_β and s_γ

Our purpose is to register the rigid regions of surface S and surface M. So s_α, s_β and s_γ should be chosen on the rigid regions of surface S. If any one of them is on the deforming regions, the transformation produced will definitely be incorrect.

From the definition of Point Signature we can see that the signature of one point describes the local underlying surface structure in the neighborhood of this point. With the assumption that the local surface structure of the points on the deforming regions will not change into that of the points on the rigid regions, the points on the rigid regions of S can be identified as points which will find points with the same signature on M, because their corresponding points on M will have similar local underlying surface structure and signature. However, due to the complexity of the deformation, this assumption cannot be completely assured. We can only expect that most of the points identified in this way will be on the rigid regions. Based on this observation, we pick up all the points on S, which can find points with the same signature on M, to form a point set, namely S^r. Selecting s_α, s_β and s_γ from S^r, we can make it more likely that they will produce correct registration and avoid unnecessary computation.

From S^r, many 3-tuples of $\langle s_\alpha, s_\beta, s_\gamma \rangle$ can be formed. If the triangle formed by s_α, s_β and s_γ is too small, the computed transformation will be very sensitive to noise [4]. To prune this redundant set of S^r, we select only those 3-tuples where the lengths of all three sides of $\langle s_\alpha, s_\beta, s_\gamma \rangle$ exceed a predefined threshold. Also, we do not allow close neighbours to be included once a particular point is chosen. The results of all valid selections consist of a set $S^t = \{\langle s_{\alpha i}, s_{\beta i}, s_{\gamma i} \rangle : i = 1, \ldots, t_s\}$, from which the needed s_α, s_β and s_γ can be chosen.

B) Selecting m_α, m_β and m_γ

Having chosen s_α, s_β and s_γ on S, the next step is to find all the possible m_α, m_β and m_γ on M, that may correspond to s_α, s_β and s_γ. To select any set of three model points as possible correspondents is not only computationally expensive but also unnecessary. Correspondence can be established based on the following constraints:

1. Point Signature constraint

Let $d_{s_p}(\theta_i)$ and $d_{m_p}(\theta_i)$ denote the Point Signature of point s_p and m_p. According to section 2.2, for correspondence they should match with each other within an error tolerance band $\varepsilon_{tol}(\theta)$.

2. Distance Constraint

Let $d(s_i, s_j)$ represent the Euclidean distance between range points s_i and s_j . For correspondence, the distance between any two points of s_α, s_β and s_γ should match, within an error of tolerance, ε_d, to the corresponding two points of m_α, m_β and m_γ.

3. Direction Constraint

Let $n_1(\cdot)$, $n_2(\cdot)$ and $n_3(\cdot)$ denote finding, at the point of interest, the unit vectors representing the directions of the normal vector, reference vector and the vector cross-product of normal vector and reference vector. Let $\hat{\mathbf{p}}_1$ and $\hat{\mathbf{p}}_2$ represent the unit vectors from s_α to s_β and s_γ. Similarly, let $\hat{\mathbf{q}}_1$ and $\hat{\mathbf{q}}_2$ represent the unit vectors from m_α to m_β and m_γ.

To provide a tolerance for error, ε_θ, for correspondence (see Figure2):

$$\triangle\theta_1| = |\arccos(\mathbf{n_1}(s_\alpha)\cdot\hat{\mathbf{p}_1}) - \arccos(\mathbf{n_1}(m_\alpha)\cdot\hat{\mathbf{q}_1})| \leq \varepsilon_\theta \quad (3)$$

$$\triangle\theta_2 = |\arccos(\mathbf{n_2}(s_\alpha)\cdot\hat{\mathbf{p}_1}) - \arccos(\mathbf{n_2}(m_\alpha)\cdot\hat{\mathbf{q}_1})| \leq \varepsilon_\theta \quad (4)$$

$$\triangle\theta_3 = |\arccos(\mathbf{n_3}(s_\alpha)\cdot\hat{\mathbf{p}_1}) - \arccos(\mathbf{n_3}(m_\alpha)\cdot\hat{\mathbf{q}_1})| \leq \varepsilon_\theta \quad (5)$$

The same condition applies for $\hat{\mathbf{p}_2}$ and $\hat{\mathbf{q}_2}$.

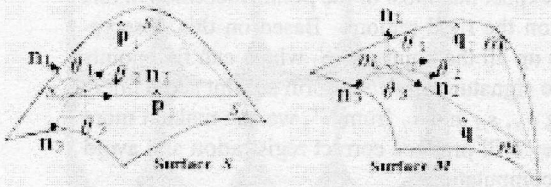


Figure 2. The direction constraints for correspondence.

Constraint 1 is first used to find a set of points on M: $m_{\alpha i}, i = 1, \ldots, p$ whose signatures match with that of s_α. For each association of s_α and $m_{\alpha i}$, the directional frame for s_α can be aligned with that of $m_{\alpha i}$ by a rigid transformation [5]. Using this transformation, we can transform s_β and s_γ to the space of M. The positions of these two transformed points indicate the approximate regions in which $m_{\beta i}$ and $m_{\gamma i}$ should be searched with the above three constraints as the corresponding points to s_β and s_γ.

Repeating the above procedure, we obtain a set of n_t 3-tuples $\langle m_{\alpha k}, m_{\beta k}, m_{\gamma k}\rangle, k = 1, \ldots, n_t$. For each association of $\langle s_\alpha, s_\beta, s_\gamma\rangle$ and $\langle m_{\alpha k}, m_{\beta k}, m_{\gamma k}\rangle$ a transformation is computed using the method in [5]. For the set of n_t 3-tuples, a set of n_t transformations, $\tau = \{T_k : k = 1, \ldots, n_t\}$, is formed.

C) Selecting the optimal transformation and verifying

A set of n_t possible transformations $\tau = \{T_k : k = 1, \ldots, n_t\}$ have been found based on the correspondences between the 3-tuple $< s_\alpha, s_\beta, s_\gamma >$ and the 3-tuples $< m_{\alpha k}, m_{\beta k}, m_{\gamma k} >$. The problem here is to single out the optimal transformation and determine if it is good enough to be an accepted solution for the registration task. To attenuate the disturbance of the points on the non-rigid regions, we only apply the transformation to the points in S^r, from which we select s_α, s_β and s_γ. When the optimal transformation is applied to the points of S^r, it will closely register most of the points on S^r with the points on M.

Let ϑ be the estimation of the maximum distance between one point on S and its closest points on M for a good estimate of the best registration (it depends on the noise level of the range data). With the transformation T_k, all the points $s_i : i = 1, \ldots, n_{sr}$ in S^r can be brought to new positions $s'_i : i = 1, \ldots, n_{sr}$. We count the number of occurrences, namely n_r, that the distance between s'_i and its nearest point on M is smaller than ϑ. We call the ratio of n_r to n_{sr} the *registration rate*. The transformation in τ with the highest *registration rate* is selected as the optimal one. If the *registration rate* of the optimal transformation is greater than a threshold, this transformation will be verified as one accepted solution to the registration task. Otherwise, another s_α, s_β *and* s_γ will be picked up from S^t to repeat the process.

The registration provided by our method is very good and accurate to some extent. This result can be further refined by some well-known iterative procedures [2, 10]. In our method, the ICP method in [10] is used for refinement with two improvements:

1.Since the two given surface are not globally rigid, it does not make sense to register based on the whole point-set of S, as was originally done in [2, 10]. In our case, we only make use of points in S^r, which can be considered as "locally rigid".

2. In ICP, "pseudo-correspondences" must be established between points on S and M. For a particular transformation, T_k and a given point s on S, the corresponding point on M is hypothesized as $m \in M$ which is closest (in Euclidean 3D space) from the transformed point of $s \in S$ (that is $T_K s$). However, due to the non-rigidity of S and M, this measure used in ICP is no longer valid. In many cases, the closest point on M may not have the same structure (i.e. the point signature) as the transformed point $T_k s$. To ensure the conformity of the underlying structure, we preselect the points on M having similar point signature as s, and from these choose the nearest point on M for the "pseudo-correspondence".

The registration result is shown in Figure 3.

Figure 3. Registration of two face-surfaces. (a), (b) rendered surfaces of two facial expressions. (c) registration of surface (a) (wire-frame) onto (b) (rendered). (d) side-view of the registration.

2.4 Finding the rigid parts of the face surface

After registering two different face surfaces, it is crucial to separate the rigid parts from the non-rigid parts. Although the rigid parts of two face surfaces will be nearer to each other than the non-rigid parts after registration, it is difficult to define a constant threshold to discriminate this difference. To handle this, we model the point distances between two registered face surfaces as a Gaussian Distribution:

$$f(x) = \frac{1}{(2\pi\sigma)^{1/2}} e^{-\frac{(x-u)^2}{2\sigma^2}}, \quad -\infty < x < +\infty \qquad (6)$$

where x is the distance between any point on S and its nearest point on M after registration, μ is the mean of the distance distribution and σ is the standard deviation. For each registration, we compute an adaptive threshold as $(\mu + \sigma)$ to distinguish between rigid and non-rigid parts. Figure 4 shows the rigid parts extracted after registration for each of the first three candidates shown in Figure 1.

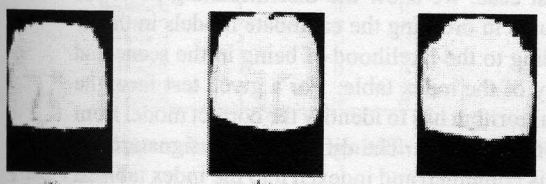


Figure 4. The extracted rigid face regions of each of the first three persons in Figure 1. (a) person 1. (b) person 2. (c) person 3.

3 Face Recognition

3.1 Creating the model library

Prior to recognition, the model library must be built up. This is done by computing the point signature for every point on the rigid regions of each model, and for every model in the library. Due to the symmetry of the human face surface, there are many different points in a same model having the same point signature. Moreover, due to the different similarity between different face surfaces, there are also many different points in different models having the same point signature. We can group all the different points with a same point signature d_s from these two sources together and represents them by a set $\langle ps,\ (model_i,\ num) \cdots \rangle$, in which the ps represents the point signature d_s and $(model_i,\ num)$ denotes that model i contains num points whose signature is the same as d_s. In such a way, we can store only one $\langle ps,\ (model_i,\ num) \cdots \rangle$ in the model library to represent all the points with a same point signature. This will considerably reduce the time needed by the voting process which will be discussed later.

For fast retrieval of every point signature in each set $\langle ps,\ (model_i,\ num) \cdots \rangle$, a three dimensional indexing table indexed by the number of peaks (the local maximum and minimum values), maximum, minimum of the point signatures is tabulated for all signatures. Each entry of the table serves as a quick reference to the group of sets in which the ps satisfies the peak number, maximum, and minimum constraints. Because the library doesn't contain all the information of the model faces, but only the most stable and meaningful regions, this considerably deduces the complexity of the indexing table and thus the time complexity of indexing process. The indexing table is illustrated in Figure 5.

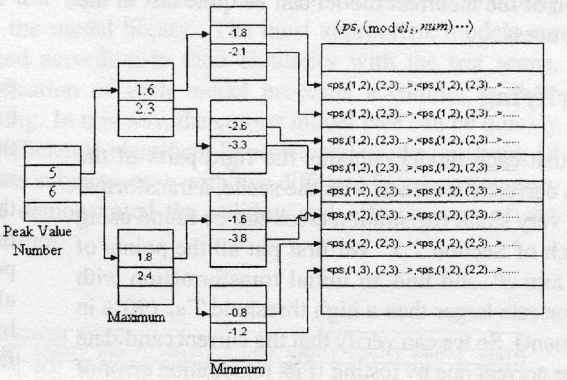


Figure 5. The index table used for fast retrival of every point signature.

3.2 Selecting the most likely model

For a given range image of a test face, we want to select a suitable set of scene points that can be used to hypothesize and "vote" for likely model faces. The scene data is preprocessed by computing the point signature of each range point and selecting only those points with different point signatures. This reduced set of scene points, S_c, are used to "vote" for the model faces. By voting, we mean that a scene point is used to index into the model library (Section 3.1) and every point with the similar signature of each model will be marked as "voted". This is repeated for the other points in S_c. Due to the fact that the correct model contains the rigid regions of the face it represents, all the points in

it will be "voted" assuming that (1) both the test and model face are in similar orientation, and (2) negligible errors in both source range data, computation and matching of point signatures.

However, the above cannot be assured and usually even the points of the most appropriate model will not entirely be "voted". Nevertheless, the models can be ranked according to their similarity with the test scene. For each model, we term the ratio of number of the "voted" points, say n_{voted}, to the total number of points, n_{all}, as the voting rate. The voting rate represents the likelihood of each model being correctly matched with the scene. Using the voting rate for each model, inappropriate models with low voting rate are rejected. The remaining candidate models are ordered according to the voting rate. With this ordering, verification can be carried out starting with the most likely candidate. It may happen that another similar face model has higher voting rate than the correct one, but such cases seldom occur as can be seen from the experimental results. Even then, the rejection of the incorrect model can be quite fast in the verifying process.

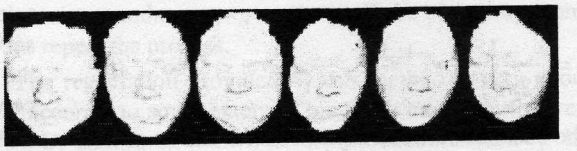
3.3 Verifying

Noting that each model contains the rigid parts of the face, if it is correctly matched with the scene, a transformation with a very small registration error can be found using the approach of Section 2.3. We first put all the points of the model into S^r and find an initial transformation with a registration rate larger than a high threshold Ts_r (90% in our experiment). So we can verify that the current candidate model is the correct one by testing if its registration error of the refined registration with the scene is small enough.

If another similar face model has a higher voting rate than the correct one, it can be rejected fast because an initial registration with a registration rate larger than Ts_r cannot be obtained. This is because, even though there may be many model points having similar point signatures as the scene points, they are at the different relative positions. Of the few possible 3-tuples of $< m_\alpha,\ m_\beta,\ m_\gamma >$ that will be found to correspond to the selected $< s_\alpha,\ s_\beta,\ s_\gamma >$, none will produce the transformation with a registration rate higher than Ts_r. In this way, the incorrect model can be rejected fast, which can be seen from the experimental results (discussed below).

4 Experimental Results

We have implemented the 3D Point Signature-Based face recognition algorithm on a Pentium III 450. We have tested the algorithm for 18 face range images extracted from a range finder with a resolution of 1 part in 7500 of the field of view. Four face range images with different expressions are captured for each of the 18 persons (the range images of the first eight persons shown in Figure 1). These range images are used to extract the rigid parts and subsequently to build the model library. Another range image is extracted for each person to represent the test scene (Figure 6 shows the test scenes extracted from the first eight persons). While our experiments in this paper are restricted to only frontal views, future work is currently conducted to handle non-front views as well.



Two test cases are examined. The first test case is to illustrate the fast ranking of the candidate models using point signature. The second one shows the ability to reject incorrect models.

Test case 1

In this test case, we show the discriminating power of point signatures in ordering the candidate models in the library according to the likelihood of being in the scene and the efficiency of the index table. For a given test face, the recognition algorithm has to identify the correct model from the library of 18 models. The different point signatures of the test face is computed and indexed into the index table of point signature. The voting rate is obtained after collecting all the "voted" points of each model. The test is repeated by using a different face in the scene, for each of the 18 test faces. The voting results (shown as the voting rate) for the 18 test faces are shown in Table 1. For each test face, the correct model is identified. The approximate execution time for each test face scene are listed in Table 2.

Test case 2

We have mentioned earlier that even if a similar but incorrect model is ordered first, our algorithm would still be able to reject the incorrect model quickly. To demonstrate this fact, we intently deleted the correct model from the library in testing each of the 18 test faces and test the time taken for the algorithm to detect the absence of the correct model in the library. For example, in testing the first test face, we deleted the model of this face from the library, etc. Although we could reject the inappropriate models with too low a voting rate (e.g., less than 50%), we have not done so for this experiment in order to test the rejection ability of our algorithm. The *registration rate* used to reject every incorrect model in the library for each test face is illustrated in Table 3. The computation time required to reject all the incorrect library models for each test face is illustrated in

Table 4.

	voting rate of each model (%)																	
	Scene1	Scene2	Scene3	Scene4	Scene5	Scene6	Scene7	Scene8	Scene9	Scene10	Scene11	Scene12	Scene13	Scene14	Scene15	Scene16	Scene17	Scene18
Model 1	**92.58**	60.61	36.36	46.21	**34.09**	28.79	80.14	62.03	66.67	40.12	73.16	53.03	43.71	67.83	63.48	74.35	79.01	43.86
Model 2	57.02	**93.97**	23.97	48.76	37.19	28.93	71.25	52.14	56.43	63.17	63.29	65.08	74.02	69.43	79.01	80.01	50.32	74.35
Model 3	73.74	70.42	**93.57**	31.03	56.25	46.37	67.40	67.34	67.87	60.79	79.05	57.08	61.74	77.83	68.63	75.62	69.73	74.05
Model 4	40.78	63.54	20.71	**90.35**	41.43	33.57	46.79	69.30	67.94	65.09	64.07	67.11	53.47	65.44	74.13	56.97	67.81	59.43
Model 5	60.14	76.08	47.55	62.94	**90.21**	49.65	58.30	78.35	57.67	70.23	76.17	69.03	70.23	67.04	66.32	60.71	69.22	67.99
Model 6	76.92	60.19	42.31	70.19	71.15	**88.85**	63.47	80.12	64.12	70.22	63.45	73.73	60.74	59.67	74.06	71.39	64.91	59.82
Model 7	82.58	60.61	36.36	46.21	34.09	28.79	**90.14**	40.32	63.67	70.12	73.29	64.05	78.21	77.63	74.82	67.54	59.14	73.49
Model 8	57.02	83.97	23.97	48.76	37.19	28.93	72.04	**92.03**	66.67	30.92	74.32	78.91	63.06	76.13	73.38	64.75	59.01	54.12
Model 9	73.44	70.94	83.75	61.25	56.25	46.88	76.39	70.21	**96.67**	72.21	79.36	67.21	72.03	80.12	73.07	69.38	72.04	68.09
Model 10	40.25	63.57	20.71	80.12	41.52	33.57	78.46	70.29	70.06	**90.12**	63.76	63.94	53.32	67.93	73.68	64.95	69.91	53.86
Model 11	60.14	76.08	47.55	62.94	80.21	49.65	69.28	49.36	69.07	68.09	**93.16**	67.03	70.19	59.97	63.49	65.38	80.14	64.43
Model 12	76.92	60.19	42.31	70.19	71.15	78.85	37.09	68.39	70.23	60.34	63.34	**93.03**	63.82	44.37	66.13	63.12	67.22	46.73
Model 13	72.58	60.61	36.36	46.21	34.09	28.79	54.03	74.06	78.09	59.67	49.83	67.01	**93.71**	58.09	65.30	56.72	78.91	65.73
Model 14	57.02	73.97	23.97	48.76	37.19	28.93	59.29	68.03	49.78	50.72	78.13	63.93	73.61	**97.83**	53.78	69.35	80.01	75.46
Model 15	73.44	70.94	73.75	31.25	56.37	46.26	79.46	66.47	65.49	73.24	70.12	49.86	50.12	49.30	**93.48**	53.20	78.24	69.43
Model 16	40.32	63.57	20.71	80.45	41.43	33.57	73.09	50.34	68.71	50.31	73.25	67.28	54.37	67.36	60.38	**94.35**	79.01	67.03
Model 17	60.14	76.08	47.55	62.94	80.21	49.65	68.23	79.90	69.93	74.39	67.07	70.10	57.36	68.09	65.37	79.80	**89.01**	58.34
Model 18	76.92	60.19	42.31	70.98	71.15	78.23	76.71	63.89	35.08	48.39	70.24	68.49	64.57	64.30	53.68	76.36	54.12	**93.86**

Table 1. *voting rate* **of each model for every scene**

	Time taken (seconds)		
	Ordering models	Verifying	Total
Face Scene 1	3	10	13
Face Scene 2	1	11	12
Face Scene 3	3	10	13
Face Scene 4	1	7	8
Face Scene 5	1	8	9
Face Scene 6	2	10	12
Face Scene 7	3	12	15
Face Scene 8	2	13	15
Face Scene 9	3	10	13
Face Scene 10	4	9	13
Face Scene 11	4	11	15
Face Scene 12	1	10	11
Face Scene 13	2	12	14
Face Scene 14	2	13	15
Face Scene 15	3	15	18
Face Scene 16	3	13	16
Face Scene 17	4	14	18
Face Scene 18	4	16	20

Table 2. Approximation recognition time for Test case 1.

don't assume limiting assumptions of rigid (or approximately rigid) surfaces but have demonstrated with faces having widely different facial expressions for each human subject. Four different facial expressions are used. The extracted range data sets are registered together, the registration being defined by the rigid parts of the face. In this way, a library of faces can be built up. Recognition of a test face is relative fast with the efficient indexing into the model library. The most appropriate models are ranked according to their similarity with the test scene. Verification of each model proceeds according to their ranking. In this way, the correct model face can be quickly and efficiently identified. Experimental results involving 18 human subjects, each with five different facial expressions, have demonstrated the validity and effectiveness of our algorithm.

	registration rate of each incorrect model (%)																	
	Scene1	Scene2	Scene3	Scene4	Scene5	Scene6	Scene7	Scene8	Scene9	Scene10	Scene11	Scene12	Scene13	Scene14	Scene15	Scene16	Scene17	Scene18
Model 1		60.61	56.36	46.21	34.09	28.79	36.84	56.19	73.16	43.71	74.35	63.48	43.86	64.13	32.58	54.10	45.08	63.16
Model 2	67.02		23.97	48.76	37.19	28.93	64.06	70.32	26.72	33.13	49.04	43.33	43.17	44.93	52.67	64.25	55.38	70.23
Model 3	73.44	70.94		31.25	56.25	46.88	33.42	36.98	63.06	53.29	70.37	23.25	33.27	54.57	65.03	74.91	68.08	45.34
Model 4	40.34	63.57	20.71		41.43	33.57	43.37	45.34	73.59	31.12	79.78	64.39	74.07	56.09	43.25	40.36	54.87	33.41
Model 5	60.14	76.02	47.55	62.94		49.65	44.83	39.09	69.78	56.01	37.34	40.82	33.19	22.13	58.03	54.67	55.32	49.89
Model 6	76.92	60.19	42.31	70.19	71.15		46.73	66.08	37.34	41.29	34.68	61.27	45.65	69.11	37.54	56.13	55.06	22.45
Model 7	82.58	60.61	36.36	46.21	34.09	28.79		30.29	73.01	73.92	57.93	36.18	62.03	69.74	37.01	43.21	33.45	23.03
Model 8	57.02	83.97	23.97	48.76	37.19	28.93	43.29		69.83	49.29	70.39	34.61	46.38	54.03	51.32	39.45	58.93	29.82
Model 9	73.44	70.94	83.75	31.25	56.25	46.88	39.33	49.11		45.09	69.47	71.03	69.37	46.01	69.74	46.07	55.69	53.02
Model 10	40.19	63.57	20.71	80.22	41.34	33.57	63.04	65.18	73.14		70.72	34.51	45.69	46.71	30.31	36.25	36.44	45.27
Model 11	60.14	76.08	47.55	62.94	80.21	49.65	36.18	54.96	37.21	71.03		39.12	21.09	24.56	60.31	49.01	60.83	73.05
Model 12	76.92	60.19	42.31	70.19	71.15	78.85	72.39	65.34	49.23	36.09	34.12		70.46	55.34	50.13	36.01	33.24	63.07
Model 13	82.58	60.61	36.36	46.21	34.09	28.79	39.51	50.93	70.91	35.64	49.82	33.45		79.08	33.47	45.92	50.91	29.03
Model 14	57.02	83.97	23.97	48.76	37.19	28.93	56.04	53.12	70.23	49.78	64.57	67.01	33.67		43.22	57.09	24.57	80.24
Model 15	73.44	70.94	93.75	31.25	56.25	46.88	47.26	65.24	78.01	49.79	73.21	60.79	44.05	34.12		20.98	57.92	70.45
Model 16	40.34	63.57	20.71	80.54	41.43	33.57	39.08	69.77	49.03	30.59	64.59	59.37	30.49	60.42	43.23		39.03	70.47
Model 17	60.14	76.08	47.55	62.94	80.21	49.65	49.08	48.91	36.29	50.71	64.56	59.44	23.94	42.03	72.05	75.09		56.73
Model 18	78.97	60.19	42.31	70.19	50.54	78.85	45.87	68.03	74.05	47.91	49.08	57.98	33.47	34.01	55.67	37.04	64.09	

Table 3. *registration rate* **of each incorrect model for every scene**

5 Conclusion

We have proposed and implemented a 3D face recognition approach using Point Signature. In our approach, we

References

[1] B. Achermann, X. Jiang, and H. Bunke. Face recognition using range images. *International Conference on Virtual Systems and Multimedia*, pages 129–136, 1997.

	Ordering models (secs)	Attempt to verify (secs)	Total time to detect absence of models (secs)
Face Scene 1	4	46	50
Face Scene 2	1	54	55
Face Scene 3	3	47	50
Face Scene 4	1	53	54
Face Scene 5	1	49	50
Face Scene 6	2	51	53
Face Scene 7	3	43	46
Face Scene 8	2	54	56
Face Scene 9	3	45	48
Face Scene 10	5	47	52
Face Scene 11	4	49	53
Face Scene 12	1	48	49
Face Scene 13	2	46	48
Face Scene 14	2	49	51
Face Scene 15	3	45	48
Face Scene 16	3	47	50
Face Scene 17	4	48	52
Face Scene 18	5	52	57

Table 4. Approximation time taken to detect absence of model in library.

[2] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, February 1992.

[3] R. Chellapa, C. L. Wilson, and S. Sirohey. Human and machine recognition of faces: A survey. *in Proceedings of the IEEE*, 83(5):705–740, May 1995.

[4] C.-S. Chen, Y.-P. Huang, and J.-B. Cheng. A fast automatic method for registration of partially-overlapping range images. *Sixth International Conference on Computer Vision*, pages 242–248, 1998.

[5] C. S. Chua and R. Jarvis. 3d free-form surface registration and object recognition. *International Journal on Computer Vision*, 17(1):77–99, 1996.

[6] C. S. Chua and R. Jarvis. Point signature: A new representation for 3d object recognition. *International Journal on Computer Vision*, 25(1):63–85, 1997.

[7] G. G. Gordon. Face recognition based on depth and curvature features. *in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 808–810, 1992.

[8] J. C. Lee and E. Milios. Matching range image of human faces. *Third International Conference on Computer Vision*, pages 722–726, 1990.

[9] H. T. Tanaka, M. Ikeda, and H. Chiaki. Curvature-based face surface recognition using spherial correlation. *Third IEEE International Conference on Automatic Face and Gesture Recognition*, pages 372–377, 1998.

[10] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal on Computer Vision*, 13(2):119–152, 1994.