Eurographics Workshop on 3D Object Retrieval (2011)
H. Laga, T. Schreck, A. Ferreira, A. Godil, I. Pratikakis, and R. Veltkamp (Editors)

SHREC'11 Track: 3D Face Models Retrieval

R. C. Veltkamp[1], S. van Jole[1], H. Drira[2,3], B. Ben Amor[3,4], M. Daoudi[3,4], H. Li[5], L. Chen[5], P. Claes[6], D. Smeets[6], J. Hermans[6], D. Vandermeulen[6], P. Suetens[6]

[1]Utrecht University, Department Computer Science, The Netherlands
[2] University of Lille1, [3] LIFL (UMR CNRS 8022), [4] Institut TELECOM/TELECOM Lille 1, France.
[4]Ecole Centrale de Lyon, LIRIS UMR5205, Lyon, France
[6]K. U. Leuven, Department of Electrical Engineering, Leuven, Belgium

Abstract

In this paper we present the results of the 3D Shape Retrieval Contest 2011 (SHREC'11) track on face model retrieval. The aim of this track is to evaluate the performance of 3D shape retrieval algorithms that can operate on 3D face models. The benchmark dataset consists of 780 3D face scans of 130 individuals. Four groups have participated in the track with 14 method variations in total.

Categories and Subject Descriptors (according to ACM CCS): I.5.4 [Pattern Recognition]: Applications—Computer vision; H.3.3 [Computer Graphics]: Information Systems—Information Search and Retrieval

1. Introduction

For this contest we have created a new three-dimensional data set consisting of scans of masks of faces. 130 masks have been scanned using a Roland PICZA LPX-250 scanner. Five range scans were made using an Escan laser scanner. The position of the mask has been varied for each of the range scans.

Using this dataset we can compare different methods and evaluate how well these performs in overcoming problems such as the pose and missing portions of the faces due to occlusion.

The process is as follows: a query is compared to all objects in the dataset. Using the recognition rate we can determine how well a certain method is able to retrieve the corresponding scan from the database. However, using the recognition rate alone limits the evaluation of the different methods. We use several other evaluation measures, such as the first and second tier recall, which takes more than only the first returned object into account, and the mean average precision, which considers the position of all relevant objects in the entire ranked list.

2. Datasets

The dataset is based on a new collection of scans, made of an anthropological collection of 130 masks, which have been acquired about a century ago [dZ14]. Two sets of scans are made. Each of the 130 masks are scanned by a Roland scanner from a single orientation, and by an Escan scanner from 5 viewing directions, see figure 1 for examples. The complete data set of 780 scans was split into three different subsets; the training set, the query set and the test set. The training set consists of 60 randomly selected Roland scans. The query set consists of 70 randomly selected Escan scans. The testset is the remainder, resulting in 70 Roland scans and 580 escans.

3. Evaluation

Every participant submitted a single ranked list per run containing every query object matched against all objects in the testset, resulting in 70 ranked lists containing 650 distance values. Scans are considered relevant only if it belongs to the same person. Since the class sizes per query object are rather small, we've decided not to use the F-measure and Precision. The numbers would be rather meaningless. Instead we calculate the recognition rate (the frequency that the highest rank is from the same class), first tier recall (the fraction of masks from the right class among the first n), second tier recall (the fraction of masks from the right class among the first $2n$), where n is the class size of the specific object, and the mean average precision (the ratio of class object among the first k ranked objects, averaged over k, where the mean is taken

DOI: 10.2312/3DOR/3DOR11/089-095

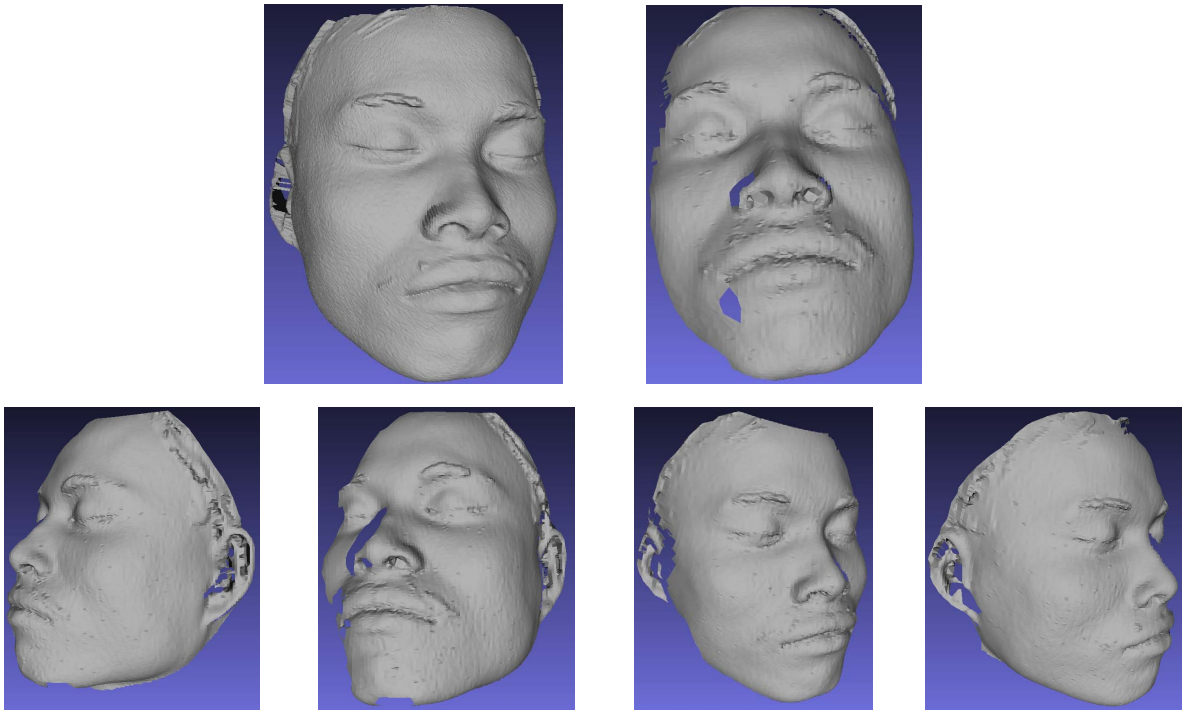

Figure 1: *Frontal scan with the Roland and with the Escan (top), and side view scans with the Escan (bottom).*

over all queries). Note that n is not a constant number since the query objects have been selected randomly, so the same object might occur more than once in the querylist.

4. Participants

Four groups have participated in SHREC'11 track on 3D face models.

- Huibin Li and Liming Chen from Ecole Centrale de Lyon, LIRIS UMR5205, Lyon, France. Their method is based on salient points, and described in section 5.
- Hassen Drira, B. Ben Amor, and Mohamed Daoudi from University of Lille1, LIFL (UMR CNRS 8022), and Institut TELECOM/TELECOM Lille 1, France. Their method is based on radial curves, and described in section 6.
- Stefan van Jole and Remco Veltkamp from Utrecht University. Their method is based on a morphable model, and described in section 7.
- Peter Claes, Dirk Smeets, Jeroen Hermans, Dirk Vandermeulen, and Paul Suetens from K. U. Leuven, Department of Electrical Engineering, Leuven, Belgium. Their method is also based on a statistical model, fitted in by a Bayesian strategy, and described in section 8.

5. Salient points

Our method is an extension of several works, including SIFT [Low04], mesh-SIFT [CTK*10], daisy [TLF08] and shape index map based SIFT matching [HAWC10]. First, feature points are detected on each 3D face models; then, the quasi-daisy local shape descriptor of each feature point can be obtained using histograms of multiple order surface differential quantities; finally, these local descriptors are matched by computing their angles like SIFT-matching. The number of matched points is employed as similarity between two face models. Fig 1 illustrate the flowchart of our method.

5.1. Salient points detection

Given a 3D mesh face model, feature points are detected by the feature detector of 3D Gaussian scale space extrema technique similar to [CTK*10, ZBVH09]. In order to detect sufficient salient points at different locations, we employ maximum and minimum principal curvatures as scalar field functions [ZBVH09].

5.2. local descriptor and feature matching

We extract multiple order surface differential quantities based local descriptor at the scale which salient points are detected. To obtain orientation invariant local descriptor, the geodesic neighboring points of each selected salient point are transformed into a rotation invariant local coordinate system as in RIFT [SS07]. Under this new local coordinate system, a local shape descriptor based on a quasi daisy neighborhood [TLF08] with nine circle topology can be obtained

on the tangent plane of the salient point. To comprehensively characterize 3D facial surfaces and their variations, we calculate weighted statistical distributions of multiple order surface differential quantities, including histogram of mesh gradient (HoG), histogram of shape index (HoS) and histogram of gradient of shape index (HoGS) in each circle. Thus, the new shape descriptor describes local shape orientation as well as shape bending and variation of shape bending. The final shape descriptor can be achieved by concatenating these histograms. The curvatures and shape index are estimated by local cubic-order fitting method [GI04]. The gradient of shape index is estimated by finite element method (FEM) based method [MEM01]. Once the shape description process has been completed for two face models, we then find the correspondence by matching shape descriptors from each of the models. We utilize a greedy feature matching process similar to the one used by SIFT. The number of correspondences points is employed as similarity between two face models.

5.3. Experimental results

In our experiments, the methods 1-3 perform fusion of the number of correspondences points originally detected by minimum and maximum curvatures with SIFT matching rate of 0.8, 0.75 and 0.7 respectively. Methods 4 and 5 are obtained by only using maximum curvature and minimum curvature respectively and the same SIFT matching rate of 0.75. Experimental results show that method 3 achieves best results for recall (47.8%), mean average precision (47.6%) and rank-1 recognition rate (92.9%). The experiments shows that the average number of feature points detected by minimum and maximum curvatures is about 100. In the matching step, because we simply use the angle of feature vectors as the similarity between two local descriptors, there will obtain some wrong matching points at the locations which have more than two similar local shapes. Further on, for a given query, there are many tests models matched the same numbers of points to it. Since we use the number of matched points as the dissimilarity between two 3D face models, there are many tests having the same dissimilarity to the given query. Thus, the mathching step and dissimilarity distance measure scheme make our method is not robust to the query changes.

6. Radial curves

6.1. Overview of proposed method

Figure 3 illustrates the overall proposed 3D face recognition method. First of all, the probe **P** and the gallery **G** images are preprocessed. This step is essential to improve the quality of raw images and to extract the useful area of the face. It consists of a Laplacian smoothing filter to reduce the acquisition noise, a filling hole filter that identifies and fills holes in input mesh, and a cropping filter that cuts and returns the part of the input mesh inside of a specified sphere. Then, a coarse alignment is performed based on the translation vector formed by the tips of the noses. This step is followed by a finer alignment based on the well-known ICP algorithm in order to normalize the pose. Next, we extract the radial curves emanating from the nose tip and having different directions on the face. Within this step, a quality control module inspects the quality of each curve on both meshes and keeps only the *good* ones based on defined criteria.

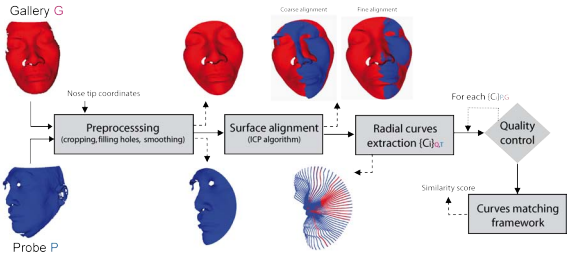

Figure 3: *Overview of the proposed method.*

Finally, using a past approach on shape analysis of curves, we obtain algorithm for computing geodesics between pairwise of radial curves on gallery and probe meshes. The length of one geodesic measure the degree of similarity between one pair of curves. The fusion of the scores on *good* common curves, produced similarity score between the faces **P** and **G**. We details the pipeline of all these stages contained in our method in the following sections.

6.2. Preprocessing and surfaces alignment

As illustrated in figure 3, we start by preprocessing the input raw images in order to improve their quality. Indeed, these images present some imperfections as holes, spikes and includes some undesired parts (clothes, neck, ears, hair, etc.) and so on. This step consists of a pipline of 3D mesh processing filters. **(i)** *Smoothing filter* reduces high frequency information (spikes) in the geometry of the mesh, making the cells better shaped and the vertices more evenly distributed. **(ii)** *Cropping filter* cuts and returns parts of the mesh inside a defined implicit function. The last-mentioned function is a sphere defined by the nose tip as its center and the radius 75mm in order to avoid as much hair. **(iii)** *Filling holes filter* identifies and fills holes in input meshes. Holes are created either because of absorption of laser in dark areas such as eyebrows and mustaches, occlusion or mouth opening. They are identified in the input mesh by locating boundary edges, linking them together into loops, and then triangulating the resulting loops. After meshes preprocessing, we correct their poses to properly compare the faces by establishing correct correspondance between sets of curves of probe and gallery images. **(iv)** *Coarse alignment filter* fix the probe image P onto the Gallery image G at the nose tip. In other words, this

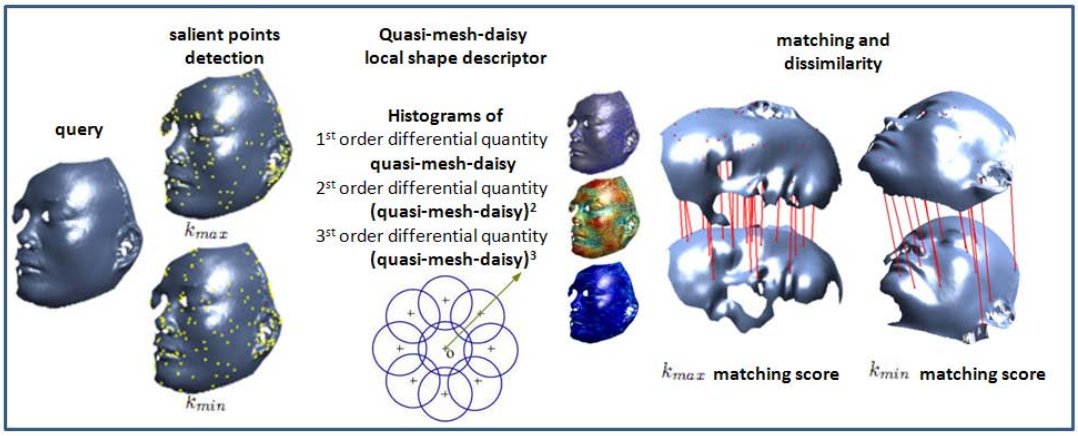

Figure 2: *Quasi-mesh-daisy local shape descriptor based free-form surface matching framework.*

filter perform a translation transform along the vector defined by the two tips of the noses. This step presents a good initialisation to the **(v)** *Fine alignment filter* which performs ICP algorithm on obtained meshes. The fine alignement filter corrects non-frontal meshes to make them frontal.

6.3. Radial curves extraction and quality control

Now we introduce our mathematical representation of a facial surface. Let S be a facial surface denoting the output of the previous preprocessing step. Although S is a triangulated mesh, we start the discussion by assuming that it is a continuous surface. Let β_α denote the radial curve which make an angle α with a reference radial curve. The reference curve is chosen to be the vertical curve once the face has been rotated to the upright position.

If needed, we can approximately reconstruct S from these radial curves according to $S \approx \cup_\alpha \beta_\alpha = \cup_\alpha \{S \cap P_\alpha\}$. This indexed collection of radial curves captures the shape of a facial surface and forms our mathematical representation of that surface. We have chosen to represent a surface with a collection of curves since we have better tools for analyzing shapes of curves than we have for surfaces.

Notice that some curves can reduce the performance of the algorithm as they have bad quality. These curves should be, first, detected then removed using the quality filter. The aim of that filter is to remove curves which have bad quality. To pass the quality filter, a curve should be continous and having a minimum of length. The discontinuity or a shortness of a curve results on missing data on the face or presence of reat noise.

Once the quality of the features is controlled, we proceed to faces comparison. In the following sections we describe our framework to analyze shape of curves in order to compare the selected features.

6.4. Shape analysis of radial curves

Let $\beta : I \to \mathbb{R}^3$, for $I = [0,1]$, represent a facial curve generated as described above. To analyze the shape of β, we shall represent it mathematically using a *square-root velocity function* (SRVF), denoted by $q(t)$, according to:

$$q(t) \doteq \frac{\dot{\beta}(t)}{\sqrt{\|\dot{\beta}(t)\|}} \,. \tag{1}$$

$q(t)$ is a special function introduced by [JKSJ07] that captures the shape of β and is particularly convenient for shape analysis. It has been shown in [JKSJ07] that the classical elastic metric for comparing shapes of curves becomes the L_2-metric under the SRVF representation. This point is very important as it simplifies the calculus of elastic metric to the well-known calculus of functional analysis under the L_2-metric. We define the set:

$$C = \{q : I \to \mathbb{R}^3 | \|q\| = 1\} \subset L_2(I, \mathbb{R}^3) \,. \tag{2}$$

With the L_2 metric on its tangent spaces, C becomes a Riemannian manifold. In particular, since the elements of C have a unit L_2 norm, C is a hypersphere in the Hilbert space $L_2(I, \mathbb{R}^3)$. In order to compare the shapes of two radial curves, we can compute the distance between them in C under the chosen metric. This distance is defined to be the length of the (shortest) geodesic connecting the two points in C. Since C is a sphere, the formulas for the geodesic and the geodesic length are already well known. The geodesic length between any two points $q_1, q_2 \in C$ is given by:

$$d_c(q_1, q_2) = cos^{-1}(\langle q_1, q_2 \rangle) \,, \tag{3}$$

and the geodesic path $\alpha : [0,1] \to C$, is given by:

$$\alpha(\tau) = \frac{1}{\sin(\theta)} (\sin((1-\tau)\theta) q_1 + \sin(\theta\tau) q_2) \,,$$

where $\theta = d_c(q_1, q_2)$.

6.5. Shape analysis of facial surfaces

Now we extend the framework from radial curves to full facial surfaces. As mentioned earlier, we are going to represent a face surface S with an indexed collection of radial curves. That is, $C \leftrightarrow \{\beta_\alpha, \alpha \in [0, \alpha_0]\}$. Through this relation, each facial surface has been represented as an element of the set $C^{[0,\alpha_0]}$. The indexing provides a correspondence between curves across faces. For example, a curve at an angle α on probe face is matched with the curve at the same angle on gallery face. With this correspondence, we can compute pairwise geodesic paths and geodesic distances between the matched curves across faces. This computation has several interesting properties. Firstly, it provides a Riemannian distance between shapes of full facial surfaces by combining distances between the corresponding radial curves.

7. Morhable model

7.1. Method

First of all, a morphable model was built using the training set. Building this model requires the trainingset to be in full correspondence. In order to achieve this, all scans were pose normalized in such a way that the orientation is the same and the nose tip is located at the origin. This is achieved by using a nose template to find the position most likely to be the nose tip, and a face template to calculate the correct orientation. After pose normalizing, 26 landmarks were manually assigned, see figure 4. These landmarks are used to build a low resolution face mesh. We subdivide each triangle into four smaller congruent triangles to generate face meshes in a higher resolution. In the end, a set of dense correspondences is acquired for each input face. After calculating the correspondence sets, a morphable model can be calculated by applying Principal Component Analysis to all the face scans so that each instance can be described as a linear combination of principal shape vectors: $S_{inst} = \overline{S} + \sum_{i=1}^{m} w_i \sigma_i s_i$. The morphable model has been built using 140 scans (the trainingset and it's mirrored variant). This means there is one average face (see figure 5), and 139 shape vectors. An instance can thus be defined as a single vector containing the weights for every shape vector.

In order to calculate the distance between a query object and a test object, we first pose normalize all query and test objects as described above. Then we fit the morphable model to each object in the query set and the test set. Two objects are compared by comparing the first 99 weights. This results in $m+n$ vectors, where $m = 70$ and $n = 650$, that represent the weights for the eigenvectors. These vectors were then compared using the Euclidean Distance (run 1) and the Weighted Minkowski Distance (run 2). The formulas for the Weighted Minkowski Distance is:

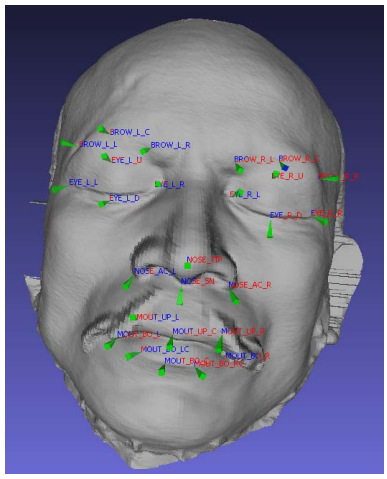

Figure 4: *Assigning landmarks.*

$$d(X,Y) = \sqrt{\sum_{i=1}^{99} w_i (x_i - y_i)^2}, \text{ where } w_i = \sum_{j=1}^{99-i} \frac{1}{99-j}$$

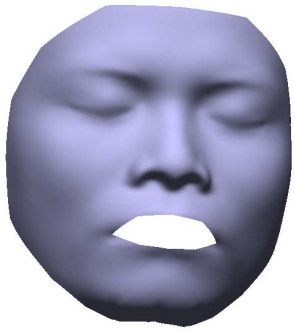

Figure 5: *Mean morphable model.*

7.2. Discussion

As can be seen in the results, both our methods score lower than the other methods, because 39 out of 70 query scans were incorrectly pose normalized. The nose tip was often located on a different part of the face that approximates the shape of the nose template (for example the cheek or the supraorbital ridge). Therefore, the morphable model had difficulties fitting the scan correctly. Another reason for failing to fit the scan accurately is because of missing face data due to occlusion which occured mainly on the non-frontal scans. In general, the morphable model was able to fit the remaining 31 scans more precisely.

The method scored relatively low for some queries, and high for some others. We will compare three query objects, of which at least one has a high

Results Per Query						
	QC	CS	CP	R1	R2	AP
query0025	Y	4	3	0.7500	0.7500	0.7520
query0037	Y	3	2	0.6667	0.6667	0.6684
query0052	N	4	2	0.0000	0.0000	0.0055

Table 1: *QC: Query correctly pose normalized?; CS: Class size; CP: Correctly pose normalized; R1: 1st Tier Recall; R2: 2nd Tier Recall; AP: Average Precision*

score and another one has a low score, see table 1. For more results per query, see http://give-lab.cs.uu.nl/SHREC/shrec2011/faces/results/. When looking at query0025, we see that the query object and three out of four objects in the testset were correctly pose normalized, thus yielding higher results. For query0052, only 2 out of 4 test objects were correctly pose normalized. The pose for the query object was also incorrect.

An advantage of our method is that we do not have to compare complete objects to each other, only their 99 weights. We simply fit the morphable model to every query and test object, in order to retrieve a single weight vector for every object. This makes comparing two objects generally easy. The main disadvantage of our method is the sensitivity to correcty align the pose, and to locate accurately the nose tip and in the case of of missing data.

8. Robust fitting of statistical model

8.1. Method

In a first stage, a PCA based shape model was learned from the given training set. To do so, the average face from an existing shape model in [ACVC10] was used here as a reference template to start from. 12 manually indicated and homologous landmarks were used to provide a rigid and crude registration for each of the training scans with the reference template. Subsequently the template was warped onto the training scans using a non-rigid registration based on variational implicit functions [Cla07]. Following a Generalized Procrustes alignment of the dense point correspondences, a principal component analysis was performed to obtain the shape model. Only 99% of the variations in the training data was retained, meaning that only 37 principal components were used hereafter.

In a second stage this model was initialized and subsequently fitted onto given test and query scans in order to generate model-based descriptions thereof for recognition purposes. The challenge and the main contribution of this work was to perform this automatically and robustly as in insensitive to spikes, holes and missing data or occlusions in the facial scans.

To normalise or initialize the test and query scans for pose with the average of the shape model, a local feature method, called meshSIFT, was used consisting of three major components: key point detection, local features description and local feature matching [MFK*10]. The algorithm first detects scale space extrema as local feature locations. In order to have an pose-invariant descriptor, each key point is assigned a canonical orientation. The meshSIFT algorithm then describes the neighbourhood of every scale space extremum in a feature vector consisting of concatenated histograms of shape indices and slant angles. The feature vectors are matched by comparing the angle in feature space. Using RANSAC, the best rigid transformation can be estimated based on the matched features. In less than 2.5% of the scan, a manual initialisation was needed.

After pose initialization, the model was fitted following a Bayesian strategy with outlier detection and estimation. The result was an EM alike optimization wherein iteratively the model updates are alternated with outlier updates. On the one hand, outliers were estimated following a stochastic approach adopted from [VLMV*01]. This dealt with spikes and other gross errors within the scan meshes. On the other hand, outliers were detected following a deterministic approach using scan border information. Model points whose closest point to the scan were located on the border were flagged as outliers. This dealt with missing data. The combination of both stochastic and deterministic outliers was done as outlined in [Cla07].

Test and query scan model-descriptions were compared using the Euclidean Cosine (run 1) and the Mahalanobis Cosine (run 2) similarity measures.

8.2. Discussion

As can be seen in the results, our method using the Euclidean Cosine as similarity measure (run 1) outperforms all participants with respect to recall and mean average precision. The performance decreases when using the Mahalanobis Cosine (run 2) as similarity measure, indicating that discriminative variation can be found following the first principal components. Wrong recognition is mostly due to model underfitting, especially near the nose and jaw regions, which are often not fully scanned. On the other hand, over-fitting occurs for query shapes for which the training database is not representative.

The main advantage of our method is that it can deal with missing data. Since the method uses a global shape model and a robust model fitting, it allows to compare face scans with little or no overlap. Local feature methods on the other hand require the scans to overlap. Another benefit is that the query scans must not be matched to every test scan, but only to the statistical shape model. Face comparison is done by comparing the corresponding model coefficients. This makes statistical methods generally the fastest. An important disadvantage of our method, and in extenso of all learning based methods, is the need for representative train-

Results				
Method	Run	Recall	MAP	RR
Salient points	1	0.3963	0.4019	0.8571
Salient points	2	0.4573	0.4536	0.9000
Salient points	3	0.4781	0.4758	0.9286
Salient points	4	0.4019	0.3957	0.8571
Salient points	5	0.3737	0.3639	0.7714
Radial curves	1	0.2046	0.2111	0.4143
Radial curves	2	0.2424	0.2477	0.4429
Radial curves	3	0.2095	0.2102	0.3143
Radial curves	4	0.2200	0.2224	0.4243
Radial curves	5	0.2186	0.2212	0.4143
Morph model	1	0.1967	0.1908	0.4143
Morph model	2	0.1948	0.1824	0.3857
Robust PCA fit	1	0.6837	0.7029	0.8857
Robust PCA fit	2	0.6123	0.6190	0.8000

Table 2: *Results per method. MAP: mean average precision, RR: recognition rate*

.

ing data. In the available training set, the bottom of the jaw is often missing and only a few female scans are included.

9. Results

Table 2 give a summary of results, in terms of recall, mean average precision, and recognition rate. The table show high performances for the method "Robust PCA fit". The recognition rates of both that method, and the "Salient Points" method are high, meaning that for most of the queries a relevant face is retrieved on top of the ranked list. The recall is the average of the first tier and second tier recall.

10. Concluding Remarks

For this content we have used a new dataset of 3D face scans. The scans have been slightly modified to remove unnecessary information from it. In total we received 14 ranked lists from 4 different groups. Since the Average Recall alone is not enough to evaluate a single run due to the small class size, the Mean Average Precision is also calculated. These two, in combination with the Recognition Rate, give a fairly good view on how well a specific method works. The two highest Mean Average Precision were both obtained by the Robust PCA fit method (run1: 0.7029; run2: 0.6190), followed by the "Salient Points" method (run3: 0.4758).

References

[ACVC10] AERIA G., CLAES P., VANDERMEULEN D., CLEMENT J. G.: Targeting specific facial variation for different identification tasks. *Forensic Science International 201* (2010), 118–124. 6

[Cla07] CLAES P.: *A robust statistical surface registration framework using implicit function representations: Application in craniofacial reconstruction*. Phd, 2007. 6

[CTK*10] C.MAES, T.FABRY, KEUSTERMANS J., SMEETS D., SUETENS P., VANDERMEULEN D.: Feature detection on 3d face surfaces for pose normalisation and recognition. In *International Conference on Biometrics: Theory, Applications and Systems, BTAS* (2010). 2

[dZ14] DE ZWAAN J. P. K.: *Die Insel Nias Bei Sumatra - Antrhopologische Untersuchungen über die Niasser*. Martinus Nijhof, 1914. 1

[GI04] GOLDFEATHER J., INTERRANTE V.: A novel cubic-order algorithm for approximating principal direction vectors. *ACM Trans. Graph 23* (2004), 45ÂIC63. 3

[HAWC10] HUANG D., ARDABILIAN M., WANG Y., CHEN L.: 3d face recognition using distinctiveness enhanced facial representations and local feature hybrid matching. In *International Conference on Biometrics: Theory, Applications and Systems, BTAS* (2010). 2

[JKSJ07] JOSHI S. H., KLASSEN E., SRIVASTAVA A., JERMYN I.: A novel representation for riemannian analysis of elastic curves in r^n. In *CVPR* (2007). 4

[Low04] LOWE D. G.: Distinctive image features from scale invariant keypoints. *International Journal of Computer Vision 60*, 2 (November 2004), 91–110. 2

[MEM01] MEYER T. H., ERIKSSON M., MAGGIO R. C.: Gradient estimation from irregularly spaced data sets. *Mathematical Geology 33* (2001), 693ÂIC717. 3

[MFK*10] MAES C., FABRY T., KEUSTERMANS J., SMEETS D., SUETENS P., VANDERMEULEN D.: Feature detection on 3D face surfaces for pose normalisation and recognition. In *BTAS '10: Proceedings of the IEEE Third International Conference on Biometrics: Theory, Applications and Systems* (Washington DC, USA, September 2010). 6

[SS07] SKELLY L. J., SCLAROFF S.: Improved feature descriptors for 3d surface matching. In *SPIE Conf. on Two- and Three-Dimensional Methods for Inspection and Metrology, SPIE* (2007). 2

[TLF08] TOLA E., LEPETIT V., FUA P.: A fast local descriptor for dense matching. In *International Conference on Computer Vision and Pattern Recognition* (2008). 2

[VLMV*01] VAN LEEMPUT K., MAES F., VANDERMEULEN D., COLCHESTER A., SUETENS P.: Automated segmentation of multiple sclerosis lesions by model outlier detection. *IEEE Transactions on Medical Imaging 20*, 8 (2001), 677–688. 6

[ZBVH09] ZAHARESCU A., BOYER E., VARANASI K., HORAUD R.: Surface feature detection and description with applications to mesh matching. In *International Conference on Computer Vision and Pattern Recognition* (2009). 2