Available online at www.sciencedirect.com
ScienceDirect
Pattern Recognition 41 (2008) 1030–1040
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/pr

Integration of local and global geometrical cues for 3D face recognition

F.R. Al-Osaimi*, M. Bennamoun, A. Mian

The University of Western Australia, 35 Stirling Highway, Crawley, WA 6009, Australia

Received 17 November 2006; received in revised form 16 May 2007; accepted 12 July 2007

Abstract

We present a unified feature representation of 2.5D pointclouds and apply it to face recognition. The representation integrates local and global geometrical cues in a single compact representation which makes matching a probe to a large database computationally efficient. The global cues provide geometrical coherence for the local cues resulting in better descriptiveness of the unified representation. Multiple rank-0 tensors (scalar features) are computed at each point from its local neighborhood and from the global structure of the 2.5D pointcloud, forming multiple rank-0 tensor fields. The pointcloud is then represented by the multiple rank-0 tensor fields which are invariant to rigid transformations. Each local tensor field is integrated with every global field in a 2D histogram which is indexed by a local field in one dimension and a global field in the other dimension. Finally, PCA coefficients of the 2D histograms are concatenated into a single feature vector. The representation was tested on FRGC V2.0 data set and achieved 93.78% identification and 95.37% verification rate at 0.1% FAR.

Keywords: 3D representation; Unified feature; Tensor field; Face recognition

1. Introduction

Face recognition is a fundamental problem in computer vision and has a wide range of vital applications in numerous and diverse domains e.g. security and human–robot interaction. Face recognition by matching 3D surfaces has shown better performance compared to 2D recognition [1]. This is an indication that the 3D shape has a significant amount of information.

The main difficulty in matching two 3D surfaces is that the surfaces are defined in different coordinate systems as the acquired surface data are defined in the coordinate system of the 3D sensor (viewer). Therefore, the 3D surface data depend on the location and orientation of the 3D sensor. There are three approaches to handle this difficulty (1) by registering the query surface to the reference surface [2,3] (2) by extracting and matching viewer-centric representations [4,5] (3) by extracting and matching object-centric representations of the surfaces [6–8].

The ICP algorithm is a well-known example of the first approach. ICP was initially introduced by Chen and Medioni [2] and Besl and McKay [9] for fine registration of two 3D pointclouds or meshes. Since then, it was widely used for both registering and matching surfaces. Starting from an initial coarse registration, it iteratively finds the rigid-body transformation that minimizes a metric error between pairs of closest points from the two meshes and applies it to one of the meshes until the metric error stabilizes. If ICP converges at the global minimum, the metric error is an indication of the similarity between the two surfaces. Although ICP does not loose any geometrical information (since it does not require any feature extraction stage), it does not always converge to the correct solution. In addition it is computationally expensive (especially if used with large databases) and requires an initial coarse registration. Another example of this approach is PCA [10] on range images which requires accurate registration of surfaces. Inaccurate surface registration adversely affects the recognition performance of PCA.

In the second approach of surface matching, viewer-centric representations, the 3D object is represented by a set of 2D images (views) that summarizes all its possible appearances. The object is then recognized by matching its representing views to the views of the reference object. Aspect graphs [4] and 2D silhouettes [5] of objects are fall in this approach of surface matching. In aspect graphs representation, topologically similar 2D views of an object are grouped and the neighboring groups

* Corresponding author. Tel.: +61 8 64883222.
E-mail address: faisal@csse.uwa.edu.au (F.R. Al-Osaimi).

0031-3203/$30.00
doi:10.1016/j.patcog.2007.07.009

in the viewpoint space are linked. Representing a 3D object using this approach requires a large number of views which does not only make it memory intensive but also complicates the object recognition task.

An object-centric representation attempts to represent 3D objects independently of the coordinate systems in which the acquired 3D surface data are defined. Thus, they are invariant to rigid transformations. Object-centric representations are usually more compact than viewer-centric representations. In addition, matching using object-centric representations is more computationally efficient. They achieves independence from the coordinate system of the data by deriving a new coordinate system from the 3D data as in the work by Mian et al. [8], decomposing the object into volumetric primitives as in the work by Greenspan et al. [7] or extracting rigid-transformation invariant surface signatures as in Ref. [6]. Our representation falls in this approach of 3D surface matching and achieves independence from the underlying coordinate system of the acquired 3D surface by utilizing rigid-transformation invariant rank-0 tensor fields (as explained in Section 3).

There are certain qualities which feature a good representation. These qualities are *unambiguity*, *conciseness* and *uniqueness* [11]. The representation is *unambiguous* if different objects have different representations. A *unique* representation does not have more than one representation for each object. Some other qualities are often considered such as representation *domain*, *completeness*, *sufficiency* and *stability* [12,13]. A *wide domain* representation can describe a large set of surfaces. For example, recognition of free-form surfaces requires representations that can describe surfaces with any arbitrary shape. A *complete* representation does not loose any surface information and the 3D surface can be reproduced from the representation. A *sufficient* representation captures enough surface information that serves the purpose of the representation. Minor changes in the acquired surface data such as that introduced by noise should not yield to a different representation (the representation is *stable*).

Trade-offs between these qualities are often involved. *Complete* representations rank low in some other qualities. For example, generalized cylinder (GR) representation [14] is complete but lacks *stability* and *uniqueness* as the object can be divided into general cylinders in multiple ways and small variations in object data can induce different divisions. B-spline representation [15,16] is also complete but not unique (nonuniqueness complicates matching or may affect the recognition performance). In addition, complete representations generally are not concise. *Conciseness* of the representation may compromise the *unambiguity* and *sufficiency* qualities. However, *conciseness*assists in improving the efficiency of the system. Despite that concise representations loose some information, they are desirable as they reduce the dimensionality of the surface data and facilitate efficient matching (given that they are *sufficiently* descriptive).

Surface representations can also be classified as local and global. Both local and global representations have their advantages and disadvantages. Global representations are extracted from the whole surface which usually makes them more concise and robust to noise. Therefore, matching global representations is computationally more efficient. However, they are sensitive to occlusions. On the other hand, local representations are extracted from local surface patches. Local representations are less sensitive to occlusions but they are computationally expensive as a large number of them need to be computed and matched. Moreover, local representations are extracted from localized surface patches which have small amounts of information. In the extreme case (very localized), the local surface has no information and coincides with the tangential plane (planar patch). Therefore, local representations are more sensitive to noise (in other words they have low signal to noise ratio). Consequently, they may easily be mis-matched with each other.

Psychological findings show that humans equally rely on both local and global visual information [17]. In this paper, we integrate both local and global geometrical cues into a *concise* object-centric representation. This representation is beneficial in two folds. Firstly, the whole surface is represented by a single feature vector which makes matching a surface to a large database computationally efficient as the computational cost is basically the computation of its representation and vector matching is computationally cheap. Therefore, as the size of the database increases, the computational cost of surface matching does not considerably increase. Secondly, the integration of local and global cues helps in increasing the *sufficiency* and simultaneously maintaining the *conciseness* of the representation. In addition, the proposed representation has shown *stability* and *robustness* to noise (see Section 6). If the global structure of the surfaces is similar (as it is the case for intra-class recognition problems such as face recognition), the local features play an important role in the representation since they are likely to be collectively dissimilar. However as previously mentioned, the local features may easily be mis-matched with each other. The integration of global geometric cues with the local ones can enhance the performance of the local cues by providing them with geometrical coherence. In other words, the local features are only matched to those which satisfy some global geometrical constraints, resulting in less confusion amongst the local features.

We devised such a hybrid local and global representation and applied it to face recognition. An earlier version of this work appeared in Ref. [18]. The 2.5D pointcloud of the face was cropped around the nose tip (the points which have distances more than a certain radius R_C from the tip of the nose were cropped off). Face cropping and nose tip detection was achieved according to Mian et al. [19]. The cropped facial pointcloud was then triangulated. From the triangulated mesh, we computed multiple local and global rank-0 tensor fields (or simply local and global scalar features at each mesh vertex), see Section 3. Unlike higher order tensor fields, it can be easily proven that the extracted rank-0 tensor fields are invariant to rigid transformations. Initially, we encoded geometric information from the triangulated mesh (which may have pose variations) in higher order tensor fields. Higher order tensor fields have the capacity to encode more surface information than rank-0 ones but

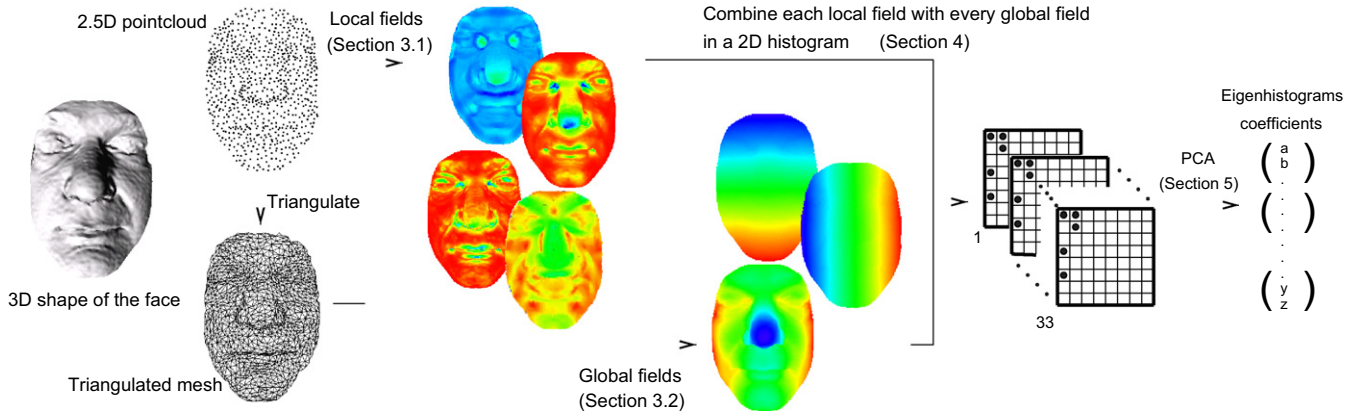

Fig. 1. Illustration of extraction of our unified local and global representation.

they vary with pose variations. Then, we computed rank-0 tensor fields from one or multiple higher order tensor fields. Thus, the geometric information is transferred from the higher order tensor fields to a larger number of rank-0 tensor fields. The extracted local and global rank-0 tensor fields are then used to integrate local and global geometrical information in multiple 2D histograms of the surface area (Section 4). Each 2D histogram is indexed by a local rank-0 tensor field in one direction and a global field in the other direction. In order to reduce the dimensionality of the representation, we performed PCA [10] on every histogram separately (Section 5). The eigenhistogram coefficients were then concatenated into a single feature vector (see Fig. 1).

2. Related data fusion work

Since our representation integrates local and global geometric cues, the representation is linked to the data fusion literature. In this section, we discuss the representation from this perspective and review related data fusion approaches. For more extensive reviews of the existing 3D surface representations that does not involve combination of local and global cues, the reader is referred to the surveys by Campbell and Flynn [13] and Mamic and Bennamoun [12].

Data fusion refers to the synergistic combination of data from multiple sources to provide more reliable and accurate information [20]. The aim of data fusion is to combine complementary and/or competing data to achieve better combined performance [21]. Inadequacies in the data can be complemented by data from other sources. For example, fusing data from multiple sources can reduce the uncertainty and increase reliability and robustness of the fused data. In pattern recognition systems, there are four fusion levels depending on the stage at which fusion takes place in the system [22], namely data level (fusion module produces a unified features from the multiple-source data), feature level (fusion module produces fused features from mono-source features), score level (fusion module produces a fused score from mono-source scores) and decision levels (fusion module produces a fused decision from mono-source decisions). It is believed that pattern recognition systems that fuse data at an earlier stage (data or feature level) will outperform those which fuse data at latter stages, because the data at lower levels have richer information about the class or the identity [23].

Our representation combines local and global information in two phases. This combination might be considered as a data level fusion in the first phase and a feature level fusion in the second phase. The first fusion phase is the integration of a local tensor field with a global field into a 2D histogram (from which unified features are extracted) and the second phase is when the unified features (eigenhistogram coefficients) of each histogram are combined into a single feature vector.

Despite the fact that there is a considerable amount of work on multi-modal data fusion (mostly texture and 3D shape at the score and the data levels) and on 2D local and global fusion, there is very limited and sparse research that addresses local and global fusion in 3D. Vandeborre et al. [24] fused local and global invariant descriptors for 3D model indexing at the rank (score) level. Their descriptors are 1D histograms of local surface curvatures, distance between mesh triangles, and volumes of the tetrahedrons formed by the triangles and the center of the 3D model. They have shown that combining the ranks of these three descriptors improves retrieval performance. Late fusion of their descriptors prevented the utilization of the important collocation information of the descriptors and their geometric relationships. Gokberk et al. [25] have performed decision and rank level fusions on four 3D face classifiers PCA, LDA, extended Gaussian images (EGI) [26] and facial profiles [27]. In the work by Xu et al. [28], a 3D face with a frontal view is uniformly triangulated. Then they used Gaussian–Hermite moments to quantify shape variations in four facial regions, namely the two eyes, nose and mouth. Finally, they performed PCA on the concatenation of the uniform range image and the Gaussian–Hermites moments. In their approach, simple concatenation of data which are of different natures (although redundant as the four facial regions are included in the range image) without normalization might impair PCA as the top principal components might entirely result from one data type that has the largest variations.

3. Facial 3D surface data and extraction of tensor fields

The representation was tested on the Face Recognition Grand Challenge (FRGC) data set V2.0 [29]. In that data set the 3D surface data were acquired using Minolta Vivid 900 range sensor which is based on the laser light-sectioning technique. The 3D facial surfaces are in the form of 3D pointclouds. Fig. 1 shows an example of a facial pointcloud. The basic data structure of a pointcloud is a $3 \times N$ matrix of x, y and z coordinates of all the points in the pointcloud.

Tensors are generalization of vectors. They vary with the transformations of their coordinate systems in which they are defined in such a way that the described mathematical or physical quantities are independent of such transformations [30]. A tensor field is a collection of tensors defined over a manifold (a tensor is attached to every point in the manifold). In our case the tensors are defined over the represented surface.

In computer vision, tensor fields have been used in the framework by Medioni et al. [31] which has been applied to many early vision problems. In their approach, they decompose a local surface patch into the basic components ball, plate and stick. Then these components (tokens) are communicated in the neighborhood and cast votes in a voting tensor field (rank-2) which is then decomposed again into ball, plate and stick with a magnitude measure (called saliency). Although their framework has worked successfully in applications like tracking [32], segmentation and surface extraction from noisy data [33], there is no evidence that the system can handle object recognition robustly as their voting scheme does not utilize sufficient global information and their local surface descriptors does not seem to be *sufficient* for such application.

In our approach, we used many rank-0 tensor fields which encode local and global information. Although, at a single surface point our rank-0 tensors are still weak descriptions of the surface's local and global geometric information, the tensor fields are synergistically combined into a *sufficient* representation.

3.1. Computation of local rank-0 tensor fields

We computed 11 local tensor fields over the triangulated mesh. For each field, a tensor is assigned to each mesh vertex. Fig. 2 shows the extracted tensor fields. These tensors are extracted from two local neighborhoods of the vertex. Each neighboring triangle which has a center of gravity $\mathbf{c}_i$ less than a certain distance threshold R_{t1} from the vertex is included in the first neighborhood $\mathscr{H}_1$. Similarly if $\mathbf{c}_i$ is less than another radius threshold R_{t2} the triangle is added to the second neighborhood $\mathscr{H}_2$. The values of R_{t1} and R_{t2} were chosen empirically to include surface patches that have sufficient geometric variations (as explained in Section 1, very localized surface patches have limited information). The difference between R_{t1} and R_{t2} should be sufficient so that the extracted tensor field captures different aspects of the local surface. In our experiments, we chose $R_{t1} = 9$ mm and $R_{t2} = 15$ mm (see Fig. 3). From each local neighborhood ($\mathscr{H}_1$ and $\mathscr{H}_2$), a rank-2 tensor field is extracted according to the following formula (we use matrix notation throughout the paper):

$$\mathbf{T}_j = \sum_{i=1}^{n_j} \frac{a_i \mathbf{r}_i \mathbf{r}_i^\top}{A_h \|\mathbf{r}_i\|^2}, \tag{1}$$

where n_j is the number of triangles in the vertex neighborhood $\mathscr{H}_j$. $\mathbf{r}_i$ represents the vector from the vertex to the centroid of the ith triangle (see Fig. 3) and a_i is the area of the ith triangle. A_h is the total area of the local neighborhood. Eq. (1) projects the surface patch onto the unit sphere and computes the covariance matrix of $\mathbf{r}$ which encodes variations of $\mathbf{r}$

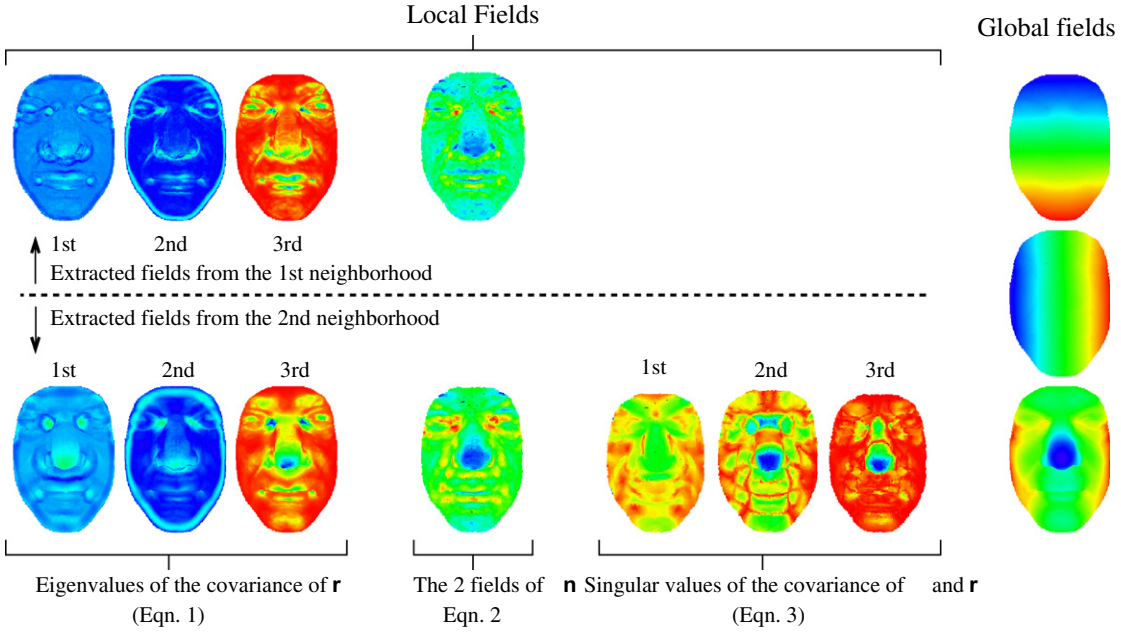

Fig. 2. Extracted local and global rank-0 tensor fields (best viewed in color).

in $\mathscr{H}_j$. The projection onto the unit sphere normalizes the contributions of the triangles in the covariance matrix so that the triangles with large $\mathbf{r}_i$ vector do not dominate others (see Fig. 4). Experimental tests have shown that this projection has improved the recognition accuracy (in comparison to the recognition accuracy when it is not used). The contribution of each triangle in the covariance matrix is weighted by its area a_i/A_h to make $\mathbf{T}_j$ less affected by the irregularities in mesh triangulation. Consequently, the representation can efficiently match decimated meshes which are usually irregular. It also makes the representation robust to variations in mesh resolutions. In our case, the mesh resolution was about 12 000 points on the face. When the facial meshes are heavily decimated to about 5000 points, a small degradation in recognition rate is noticed (about 4% from the demonstrated rate in Section 6).

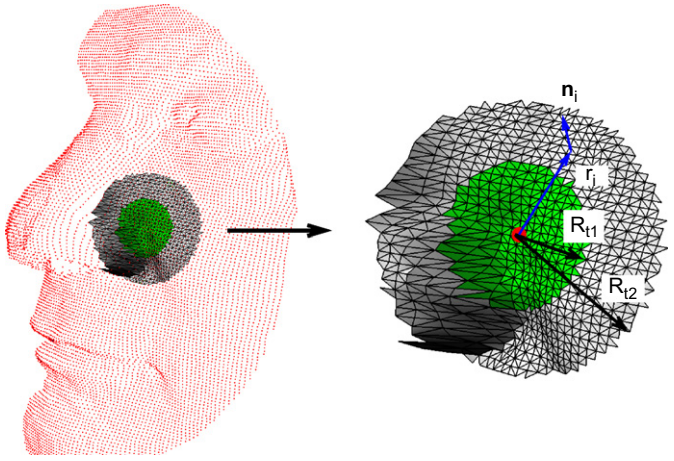

Fig. 3. Two threshold values (R_{t1} and R_{t2}) are used select two local neighborhoods of a vertex.

From each of the two rank-2 tensors, three rank-0 tensors are extracted. The three eigenvalues of each $\mathbf{T}_j$ are sorted in descending order ($\lambda_1 \geqslant \lambda_2 \geqslant \lambda_3$) and considered three rank-0 local tensor fields (L_{f1}, L_{f2} and L_{f3}). Since the rank of $\mathbf{T}_j$ is greater than zero, it varies with rigid transformations. $\mathbf{T}_j$ is decomposed into rigid-transformation independent components (eigenvalues) and dependent components (eigenvectors). If the local surface is planar, its projection onto the unit sphere is a circular line with equal density. In this case $\lambda_1 = \lambda_2$ and $\lambda_3 = 0$. However, if the local neighborhood has variations, the eigenvalues will vary according to the distribution of the local surface patch on the unit sphere. Fig. 4 shows surface projections onto the unit sphere.

Five additional rank-0 tensor fields are extracted from the normals of the triangles $\mathbf{n}_i$ and their $\mathbf{r}_i$ vectors. Two of these are from Eq. (2) and the remaining ones are from Eq. (3):

$$\mathbf{l}_j = \sum_{i=1}^{n_j} \frac{a_i \mathbf{n}_i \cdot \mathbf{r}_i}{A_h \|\mathbf{r}_i\|}, \tag{2}$$

$$\mathbf{Q} = \sum_{i=1}^{n_2} \frac{a_i (\mathbf{n}_i - \bar{\mathbf{n}}) \mathbf{r}_i^{\top}}{A_h \|\mathbf{r}_i\|}. \tag{3}$$

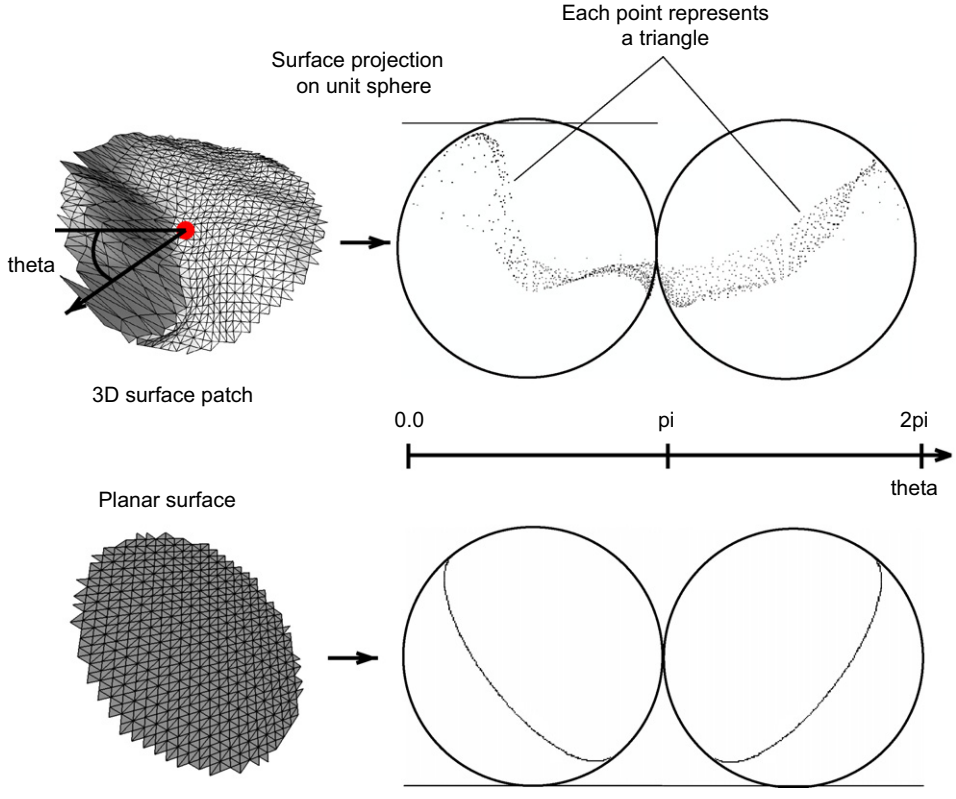

Fig. 4. Projection of an arbitrary local surface patch onto the unit sphere (top). The projection of a planar surface onto the unit sphere is a circle (bottom).

In Eq. (2), $\mathbf{n}_i \cdot \mathbf{r}_i$ is the dot product between the normal of the ith triangle and its vector $\mathbf{r}_i$ (see Fig. 3). From each neighborhood ($\mathscr{H}_1$ and $\mathscr{H}_2$ correspond to the same local neighborhoods used in Eq. (1)), we get one rank-0 tensor field. These two tensor fields provide measures of how the normals in the neighborhood are oriented with respect to the vector $\mathbf{r}_i$. For a planar surface the value of this tensor field is zero because the normals of the triangles are always perpendicular to the $\mathbf{r}_i$ vectors. However, if the normals in the local neighborhood are tilted away from the vertex (e.g. in a convex surface), the field will be positive. The field is negative if the normals are tilted towards the vertex (e.g. in a concave surface). In these three special cases, the field resembles the mean curvature of the surface. However, the tensor field is different for arbitrary surface variations in the local neighborhood.

The two local fields which are extracted from Eq. (2) are sensitive to the variations in normals relative to their $\mathbf{r}_i$ vectors regardless of the $\mathbf{r}_i$ directions. In contrast, the tree rank-0 local fields which are extracted from Eq. (3) relate directional quantities derived from $\mathbf{r}$ and $\mathbf{n}$. The tensor computed in Eq. (3) ($\mathbf{Q}$) is the covariance matrix between the normals of the triangles $\mathbf{n}_i$ and their $\mathbf{r}_i$ vectors (after projection on the unit sphere). $\mathbf{Q}$ was computed only from the large neighborhood ($\mathscr{H}_1$). Unlike the rank-2 tensor in Eq. (1), the tensor in Eq. (3) is unsymmetrical and may have complex eigenvalues. Therefore, instead of using the eigenvalues as rank-0 tensor fields, the tensor is decomposed into its singular values ($\mathbf{Q}_k = \mathbf{USV}^\top$). As the previously used eigenvalue decomposition, singular value decomposition also decomposes the rank-2 tensor into rigid-transformations invariant components (singular values) and directional components (the orthonormal columns of $\mathbf{V}$ and the orthonormal columns of $\mathbf{U}$). The singular values (s_1, s_2 and s_3) are considered rank-0 tensor fields. Nishida et al. [34] have shown that the singular values of the matrix of x, y and z coordinates of a pointcloud are invariant to rotations. In a similar way, it can be shown that the singular values of the covariance matrix of $\mathbf{n}$ and $\mathbf{r}$ are invariant to rigid transformations. The singular values relate columns of $\mathbf{V}$ to the columns of $\mathbf{U}$ according to $\mathbf{Qv}_i = s_i \mathbf{u}_i$. Hence, the counterpart of $\mathbf{U}$ and $\mathbf{V}$ in $\mathbf{Q}$ are $\mathbf{n}$ and $\mathbf{r}$, respectively.

The normals of a 3D surface have more variations along directions with more curvatures. The first singular value could be an indication (to some extent) of the absolute value of the maximum principal curvature. The tensor field of the first singular value in Fig. 2 shows that the facial regions with more absolute maximum principal curvature have more tensor values. For example, at the nose which is cylindrically shaped and has a large maximum principal curvature but a low minimum curvature, the values of this field are high. However, the field of the second singular value is very low at the center of the nose. At the two ends of the nose (nose tip and the region between the eyes) both fields are high because the surface is highly curved in different directions. The tip of the nose is convex and its normals vary in all directions but the other region is saddle shaped and highly curved in two perpendicular directions, the horizontal direction (positively curved) and the vertical direction (negatively curved).

Based on the described similarities between the principal curvatures and these singular values, these singular values are potentially applicable in approaches that are based on the principal curvatures, e.g. the shape index by Lu et al. [35] and approaches that segment the 3D surface according to its principal curvatures (e.g. Ref. [36]). The shape index represents distinct nine local shapes based on the principal curvatures and is used to detect certain points on the 3D face such as eye corners and nose tip. We expect these singular values to be more robust to noise compared to the principal curvatures as they are extracted from much larger neighborhoods. On the other hand, the principal curvatures are mostly computed from the second derivatives of the 3D surface (which are to some extent sensitive to noise) or fitting a paraboloid to a small local neighborhood then the principal curvatures are analytically computed from the paraboloid [37].

3.2. Computation of global rank-0 tensor fields

Three rank-0 global fields are extracted from the cropped face. The centroid of the face mesh is computed according to

$$\mathbf{g} = \frac{1}{A_c} \sum_{i=1}^{n} a_i \mathbf{c}_i, \tag{4}$$

$$\mathbf{M} = \sum_{i=1}^{n} a_i (\mathbf{c}_i - \mathbf{g})(\mathbf{c}_i - \mathbf{g})^\top, \tag{5}$$

where n is the number of the triangles in the mesh. a_i and $\mathbf{c}_i$ are the area and the centroid of the ith triangle, respectively. A_c is the total area of the cropped face. Then the covariance matrix of the whole cropped surface $\mathbf{M}$ is computed. Its three eigenvectors $\mathbf{p}_1$, $\mathbf{p}_2$ and $\mathbf{p}_3$ (principle directions of the cropped face) which correspond to the eigenvalues sorted in decreasing order are assumed uniform rank-1 tensors at every vertex. Since the negative of an eigenvector is also an eigenvector as $\mathbf{M}(-\mathbf{p}) = \lambda(-\mathbf{p})$, the principal directions $\mathbf{p}_1, \mathbf{p}_2$ and $\mathbf{p}_3$ may have 180° *ambiguity*. The eigenvectors are checked against reference vectors limiting the permissible pose variations to less than ±90°. In this range, the principle directions behave like rank-1 tensors with regard to geometric transformations. The dot product $\mathbf{p}_j \cdot (\mathbf{c}_i - \mathbf{g})$ produces a global rank-0 tensor field from each principle direction. These global fields redefine the 3D surface in the coordinate system determined by the three principal components as its directed axes and the global centroid as its origin. The three global fields are independent of the pose of the surface.

In fact, the three principal components are not *uniquely* definable for every surface. For example, when two eigenvalues of $\mathbf{M}$ are equal, it is not possible to order the eigenvalues uniquely. Very close eigenvalues may affect the *stability* of the representation as noise may influence the ordering of eigenvalues. However, the representation is applicable to a large *domain* of surfaces including the human face for which the principal directions are *robustly* definable. In addition, principal directions are not the only way to extract global fields. Another possible way is to compute several vectors $\mathbf{k}_j$ from the whole surface

such as those defined in Eq. (6). Different values of m_j yield to different vectors (e.g. $\frac{1}{2}$ and 2, the value of 1 should be avoided as it gives $\mathbf{k}_j = [000]^\top$). The dot product can also be used to define global fields, $(\mathbf{c}_i - \mathbf{g}) \cdot (\mathbf{k}_j/\|\mathbf{k}_j\| - \mathbf{k}_{j+1}/\|\mathbf{k}_{j+1}\|)$

$$\mathbf{k}_j = \frac{1}{A_c} \sum_{i=1}^{n} a_i \|\mathbf{c}_i - \mathbf{g}\|^{m_j} \frac{\mathbf{c}_i - \mathbf{g}}{\|\mathbf{c}_i - \mathbf{g}\|}. \quad (6)$$

4. Fusion of local and global tensor fields

In Section 3, the local and global rank-0 tensors are extracted at every mesh vertex (tensor fields). Values of the various fields are estimated for mesh triangles by averaging field values at the vertices of the triangle. Each rank-0 local tensor field is integrated with each global rank-0 tensor field in a 2D histogram. The histogram is indexed by a local field in one dimension and a global field in the other dimension. The area of every triangle in the mesh is added to the bin that is indexed by its local and global fields. The co-location of the local and global fields at the mesh triangles determines to which 2D histogram bins their areas are added. Since all the 11 local fields and the 3 global fields are invariant to rigid transformations the 2D histograms are also invariant (as long the face is not largely rotated resulting in self-occlusion of some points).

In order to utilize the bins of the histogram efficiently, nonuniform griding was used. Grid sizes are chosen so that the histogram bins have equal probability as illustrated in Fig. 5. The grids are spaced according to the mean μ_{f_i} and the standard deviation σ_{f_i} of every indexing field f_i. μ_{f_i} and σ_{f_i} are computed offline from all the gallery faces. The grids D_j of each field are placed around the mean field according to

$$D_i = \mu_{f_j} + k_i \sigma_{f_j}, \quad (7)$$

where k_i is the nonuniform griding that yields equal density distribution for the normal probability density function $\mathcal{N}(\mu = 0, \sigma = 1)$ (see Table 1 for the used μ, σ and k_i values). Fig. 5 shows a nonuniform histogram griding example of a global tensor field with $\sigma = 2$ and a local field with $\sigma = 1$. The choice of the dimension of the histograms is critical. Too coarse

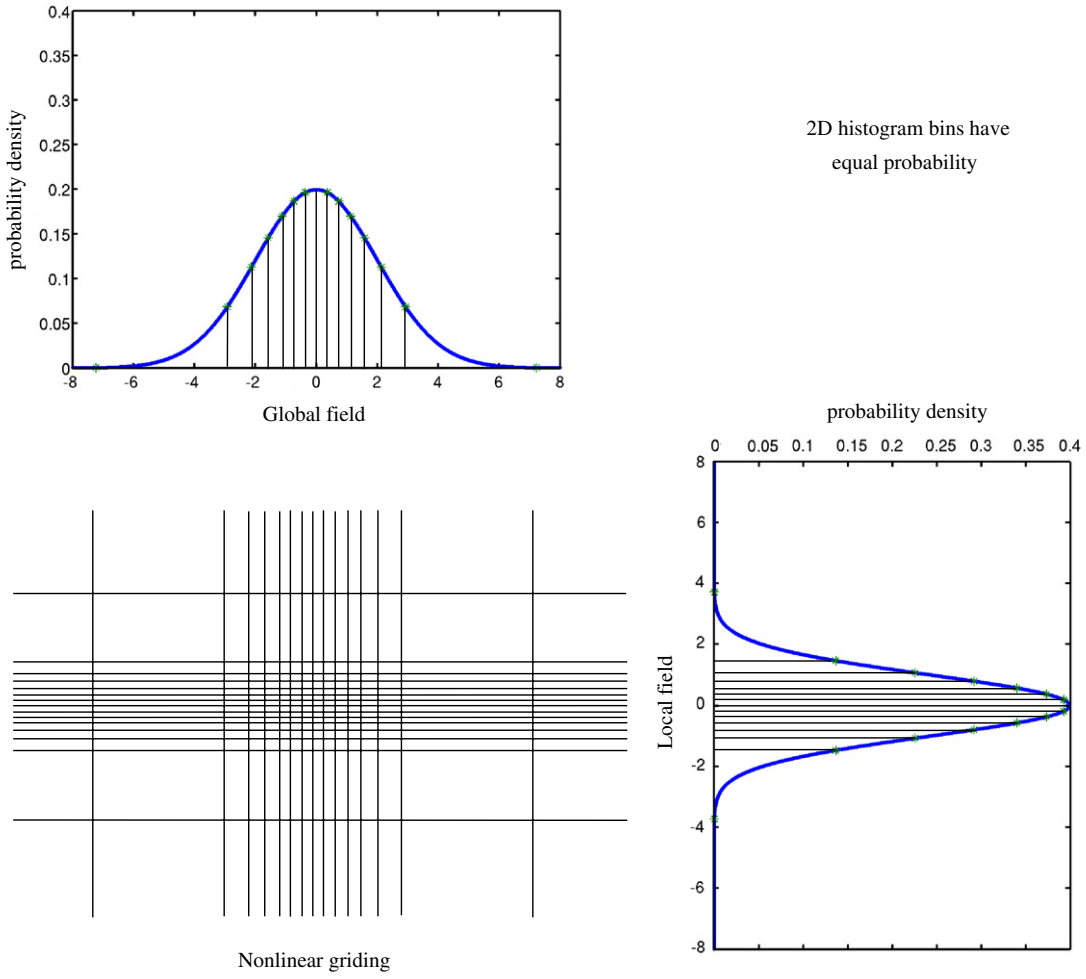

Fig. 5. Illustrating example of the non-linear histogram griding. The 2D histogram is indexed by a local field with standard deviation $\sigma = 1$ and mean $\mu = 0$ in one dimension and a global field with $\sigma = 2$ and $\mu = 0$ in the other dimension.

Table 1
Mean and standard deviation values of the tensor fields (top)

Local fields											Global fields		
	1	2	3	4	5	6	7	8	9	10	11	12	13
μ_{f_j}	0.05	20.8	17.5	1.82	0.11	56.7	48.9	6.02	1.93	0.71	2.45	−0.1	0.44
σ_{f_j}	0.88	2.7	2.65	2.29	1.55	5.44	6.03	6.81	1.06	0.34	37.3	29.7	10.7
k_i	−1.46	−1.07	−0.79	−0.57	−0.37	−0.18	0.0	0.18	0.37	0.57	0.79	1.07	1.46

k_i factors which were used to define the non-linear griding of the 2D histograms (bottom).

histogram griding results in an *insufficient* and *ambiguous* representation because significant field variations may not bring about a change in bin indexing. On the other hand, too fine griding adversely affects the *stability* of the representation and its *robustness* to noise. Minor changes in the indexing fields, caused by noise or variation in mesh triangulation, may be sufficient to make the fields index another histogram bin. The dimension of the field histograms used in the representation was 14×14. A total number of 33 field histograms are used to represent a surface (11×3).

5. Histograms compression and matching

The principal component analysis (PCA) [10] algorithm has been widely used in data compression and pattern recognition. The higher principal components account for most of the data variations and dropping the lower principal components may not cause significant data loss. The PCA algorithm was applied to each 2D field histogram individually. PCA represents a field histogram in lower dimensional space. The 2D histogram is vectorized into $m \times 1$ vector $\mathbf{h}$, where m is the number of bins of the histogram. The covariance matrix $\mathbf{\Omega}$ of the corresponding fields in the face gallery is computed as

$$\mathbf{\Omega} = \sum_{i=1}^{n} (\mathbf{h}_i - \bar{\mathbf{h}})(\mathbf{h}_i - \bar{\mathbf{h}})^{\top}, \tag{8}$$

$\bar{\mathbf{h}}$ is the average histogram. The eigenvectors $\mathbf{e}$ of $\mathbf{\Omega}$ that have the top k eigenvalues λ are sorted by their eigenvalues in decreasing order in the matrix $\mathbf{E}$. Instead of projecting the histogram difference $\mathbf{h}_{d_i} = \mathbf{h}_i - \bar{\mathbf{h}}$ onto the orthonormal subspace $\mathbf{E}$ as in the case of the PCA algorithm ($\mathbf{E}^{\top}\mathbf{h}_{d_i}$), $\mathbf{h}_{d_i}$ is projected on the eigenvectors after dividing each eigenvector by the square root of its eigenvalue which implies that the subspace is transformed (equivalent to the statistical whitening transform [38]) so that the variance along all the principle directions are equal ($w = (\mathbf{ET})^{\top}\mathbf{h}_{d_i}$, where the $\mathbf{T}$ matrix is diagonal and has $\{1/\sqrt{\lambda_1}, \ldots, 1/\sqrt{\lambda_k}\}$ diagonal entries). An eigenvalue of the $\mathbf{\Omega}$ is proportional to the squared magnitude of the projection of $\mathbf{h}_{d_i}$ on the corresponding principal direction (see Eq. (9)):

$$\mathbf{\Omega} \approx \sum_{j=1}^{k} \lambda_j \mathbf{e}_j \mathbf{e}_j^{\top} = \sum_{j=1}^{k} \sum_{i=1}^{n} (\mathbf{h}_{d_i} \cdot \mathbf{e}_j)^2 \mathbf{e}_j \mathbf{e}_j^{\top}. \tag{9}$$

Thus, dividing the projection of $\mathbf{h}_{d_i}$ by $\sqrt{\lambda_j}$ produces a covariance matrix with eigenvalues equal to one. Consequently, the selected principle components contribute equally to the histogram matching. Our tests revealed that this subspace transformation has considerably improved the recognition accuracy. The eigen-histogram coefficients w of all the field histograms are concatenated in a single feature vector $\mathbf{w}$. The feature vectors are then matched using Euclidean distance. In addition to normalization along the top principal direction within each individual histogram, the transformation also achieves normalization among all histograms. The variations in some histograms may be significantly larger than the variations in other histograms. Consequently, the encoded geometric information in the histograms with small variations may not translate into proportional distance in histogram matching. This also explains why the application of PCA on each histogram individually outperforms PCA on the concatenated histograms as the top selected principal directions might belong to the histograms with the highest variations and the geometric information in the other histograms might be lost.

6. Results and discussions

The representation was tested on neutral expression faces of the FRGC v2.0 data set. This data set was chosen for testing the representation mainly for two reasons. (1) Face recognition is a subclass of object recognition and possibly more challenging because the shape of the face is deformable and varies with expression (even the neutral expression faces have mild variations) and aging. In addition, it is an intra-class recognition problem which requires highly descriptive representations for the classification into subclasses. (2) In contrast to object recognition databases, the FRGC v2.0 data set is considerably large and has 466 subjects and about 2410 facial pointclouds with neutral expressions (Fig. 6).

Out of the 2410 facial pointclouds, 466 ones were considered as the gallery faces (one training pointcloud per subject). The remaining 1944 facial pointclouds were used to test the representations. Our proposed representation achieved 95.37% verification rate at 0.1% false accept rate (FAR) on the 466 galary faces. Fig. 7(a) shows the verification performance in an ROC curve and Fig. 7(b) shows the recognition rates for the first 20 ranks. An identification rate of 93.78% was achieved. The curve indicates that the recognition rate rapidly increases from the first rank to the 10th rank. This indicates that recognition will significantly improve when 2D information is used or another classifier is integrated with this classifier.

Fig. 6. Some probe faces (1st line) that were mis-recognized as gallery faces shown in the 3rd line. The correct gallery faces are shown in the middle line.

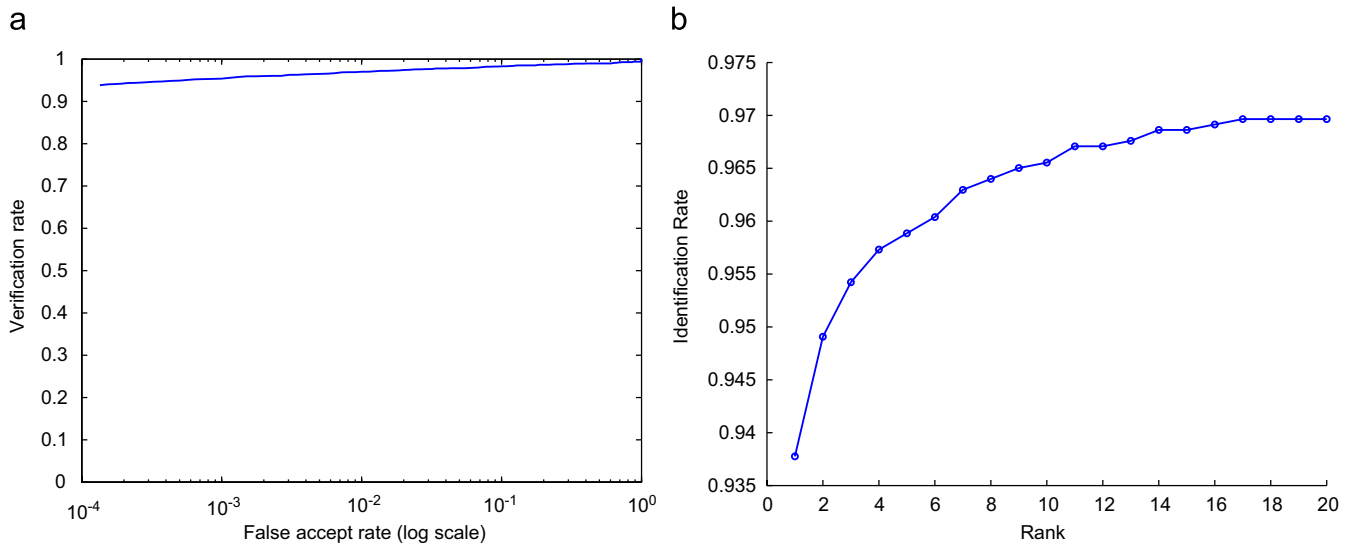

Fig. 7. Recognition performance of 466 gallery faces and 1944 probes.

To test how the global fields contribute to the representation, these fields were discarded from the 2D histograms. Instead of indexing each 2D histogram by a global field and a local field, it was indexed by two different local fields. The recognition performance in this case decreased significantly to less than 50%. Matching using only the global fields is not possible because they are not sufficiently descriptive by themselves. We conclude from this that the fusion of the local and global fields increases the descriptiveness of the representation.

Failures in the recognition were mainly due to large surface variations between the probes and the gallery faces (See Fig. 6). The main sources of these variations are hair and other surface artifacts caused by noise in the pointcloud. The faces were automatically cropped and the cropping errors also affected the representation. The local fields are robust to surface cropping errors and artifacts as they are locally extracted. If these factors introduce large changes in the global structure, the global tensor fields may change. Recall from Section 4 that large field variations can introduce a change in histogram bin indexing. In such a situation, an offset between the local fields and the global fields might happens and result in surface mis-recognition. The tolerance of the global field to cropping errors and artifacts is related to the number of histogram grids. Histograms with less number of bins are more tolerant to cropping errors and artifacts but as mentioned in Section 4, the *sufficiency* of the representation will decrease too. In some cases, mis-recognized gallery faces have very similar 3D shapes even to the human eye. In this case, if the face is mis-recognized in the first rank, it is still within the top few ranks.

7. Conclusion

A surface representation that fuses local and global geometrical cues at the data level in a compact representation is presented. The local geometric cues are encoded in multiple rank-0 tenor fields and similarly the global cues are encoded in other rank-0 tensor fields. The local and global fields are integrated into 2D histograms. Then PCA is performed separately on every histogram. The fusion of the local and the global fields (in our representation) has shown better performance than when

relying only on local or global tensor fields. Our tests have shown that applying PCA on multiple-source data individually outperforms PCA on the concatenation of the multiple-source data (in our case the 2D histograms) because PCA effectively may discard data that has smaller variances than the others. Also, our results have shown that the statistical whitening transform of the subspaces of the multiple PCAs normalizes their features and consequently improves the recognition performance.

References

[1] K. Bowyer, K. Chang, P. Flynn, A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition, Comput. Vision Image Understanding 101 (2006) 1–15.

[2] Y. Chen, G. Medioni, Object modeling by registration of multiple range images, IEEE Trans. Pattern Anal. Mach. Intell. 3 (1991) 2724–2729.

[3] C. Hesher, A. Srivastava, G. Erlebacher, A novel technique for face recognition using range imaging, in: Seventh International Symposium on Signal Processing and Its Applications, 2003, pp. 201–204.

[4] J. Koenderink, A. van Doorn, Surface shape and curvature scales, Image Vision Comput. 10 (1992) 557–565.

[5] S. Ullman, R. Basri, Recognition by linear combination of models, IEEE Trans. Pattern Anal. Mach. Intell. 13 (1991) 992–1006.

[6] A. Johnson, M. Hebert, Using spin images for efficient object recognition in cluttered 3D scenes, IEEE Trans. Pattern Anal. Mach. Intell. 21 (1999) 433–449.

[7] M. Greenspan, P. Boulanger, Efficient and reliable template set matching for 3d object recognition, in: Proceedings of the Second International Conference on 3D Digital Imaging and Modelling, 1999, pp. 230–237.

[8] A. Mian, M. Bennamoun, R. Owens, Three-dimensional model-based object recognition and segmentation in cluttered scenes, IEEE Trans. Pattern Anal. Mach. Intell. 10 (2006) 1584–1601.

[9] P. Besl, N. McKay, A method for registration of 3-D shapes, IEEE Trans. Pattern Anal. Mach. Intell. 14 (1992) 239–256.

[10] I. Jollife, Principal Component Analysis, Springer, Berlin, 1986.

[11] C. Brown, Some mathematical and representational aspects of solid modeling, IEEE Trans. Pattern Anal. Mach. Intell. 3 (1981) 444–453.

[12] G. Mamic, M. Bennamoun, Representation and recognition of 3D free-form objects, Digital Signal Process. 12 (2002) 47–76.

[13] R. Campbell, P. Flynn, A survey of free-form object representation and recognition techniques, Comput. Vision Image Understanding 81 (2001) 166–210.

[14] R. Nevatia, T. Binford, Description and recognition of curved objects, Artif. Intell. 8 (1977) 69–76.

[15] P. Dierckx, Curve and Surface Fitting with Splines, Oxford Science, New York, 1993.

[16] V. Koivunen, R. Bajcsy, Spline representations in 3-D vision, in: Object Representation in Computer Vision, Object Representation in Computer Vision, 1995, pp. 177–190.

[17] J. Vogel, A. Schwaninger, C. Wallraven, H. Blthoff, Categorization of natural scenes: local vs. global information, in: Symposium on Applied Perception in Graphics and Visualization APGV, 2006.

[18] F. Al-Osaimi, M. Bennamoun, A. Mian, 3D shape representation by fusing local and global information, in: IEEE Symposium on Signal Processing and Its Applications, 2007.

[19] A. Mian, M. Bennamoun, R. Owens, Automatic 3D face detection, normalization and recognition, in: 3DPVT, 2006.

[20] R. Luo, C. Yih, K. Su, Multisensor fusion and integration: approaches, applications, and future research directions, IEEE Sensors J. 2 (2002) 107–119.

[21] J. Hackett, M. Shah, Multi-sensor fusion: a perspective, in: IEEE International Conference on Robotics and Automation, 1990, pp. 1324–1330.

[22] A. Ross, A. Jain, J. Qian, Information fusion in biometrics, Lecture Notes in Computer Science, Springer, Berlin, 2001, pp. 354–359.

[23] A. Ross, A. Jain, Multimodal biometrics: an overview, in: Proceedings of 12th Signal Processing Conference (EUSIPCO), 2004, pp. 1221–1224.

[24] J. Vandeborre, V. Couillet, M. Daoudi, A practical approach for 3D model indexing by combining local and global invariants, in: 3DPVT, vol. 1, 2002, pp. 644–647.

[25] B. Gokberk, A. Salah, L. Akarun, Rank-based decision fusion for 3D shape-based face recognition, Lecture Notes in Computer Science, Springer, Berlin, 2005, pp. 1019–1028.

[26] S. Kang, K. Ikeuchi, The complex EGI: new representation for 3-D pose determination, IEEE Trans. Pattern Anal. Mach. Intell. 15 (1997) 707–721.

[27] Y. Lee, K. Park, J. Shim, T. Yi, Face recognition using statistical multiple features for the local depth information, in: 16th International Conference on Vision Interface, 2003.

[28] C. Xu, Y. Wang, T. Tan, L. Qusn, Automatic 3D face recognition combining global geometric features with local shape variation information, in: IEEE International Conference on Automatic Face and Gesture Recognition, vol. 6, 2004, pp. 308–313.

[29] P. Phillips, P. Flynn, T. Scruggs, K. Bowyer, J. Chang, K. Hoffman, J. Marques, M. Jaesik, W. Worek, in: IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, 2005, pp. 947–954.

[30] J. Heinbockel, Introduction to Tensor Calculus and Continuum Mechanics, Trafford, 1995.

[31] G. Medioni, M. Lee, C. Tang, A Computational Framework for Feature Extraction and Segmentation, Elsevier, Amsterdam, 2000.

[32] P. Kornprobst, G. Medioni, Tracking segmented objects using tensor voting, in: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, 2000, pp. 118–125.

[33] C. Tang, G. Medioni, Curvature-augmented tensor voting for shape inference from noisy 3Ddata, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2002) 858–864.

[34] T. Nishida, S. Yamada, A. Eguchi, Y. Fuchikawa, S. Kurogi, Range data matching for object recognition using singular value decomposition, in: Proceedings of SCI2004, vol. 5, 2004, pp. 29–34.

[35] X. Lu, A. Jain, D. Colbry, Matching 2.5D face scans to 3D models, IEEE Trans. Pattern Anal. Mach. Intell. 28 (2006) 31–43.

[36] A. Moreno, A. Sanchez, J. Fco, V. Fco, J. Diaz, Face recognition using 3D surface-extracted descriptors, in: IMVIP, 2003.

[37] P. Krsek, G. Lukas, R. Martin, Algorithms for computing curvatures from range data, Math. Surf. 3 (1998) 1–16.

[38] R. Duda, P. Hart, D. Stork, Pattern Classification, Wiley, New York, 2001 p. 34.

About the Author—FAISAL R. AL-OSAIMI received his BSc degree in electrical and computer engineering from The University of Umm Al-Qura, Saudi Arabia, in 2000. After that, he worked in the field of data communication. In 2005, he received ME degree in computer systems engineering from Queensland University, Australia, in 2005. Currently he is working towards his PhD degree in the field of computer vision in The University of Western Australia. His current research interests include mobile robotics, pattern recognition and biometrics.

About the Author—MOHAMMED BENNAMOUN received the MSc degree from Queen's University, Kingston, Canada, in the area of control theory, and the PhD degree from Queen's/QUT in Brisbane, Australia, in the area of computer vision. He lectured in robotics at Queen's, and then joined QUT in 1993 as an associate lecturer. He then became a lecturer in 1996 and a senior lecturer in 1998 at QUT. In January 2003, he joined the School of Computer Science and Software Engineering at The University of Western Australia as an associate professor. He was also the director of a research center from 1998 to 2002.

He is the coauthor of the book Object Recognition: Fundamentals and Case Studies (Springer-Verlag, 2001). He has published more than 100 journal and conference publications. He served as a guest editor for a couple of special issues in international journals, such as the International Journal of Pattern Recognition and Artificial Intelligence. He was selected to give conference tutorials at the European Conference on Computer Vision 2002 and the International Conference on Acoustics Speech and Signal Processing (ICASSP) in 2003. He organized several special sessions for conferences; the latest was for the IEEE International Conference in Image Processing (ICIP) held in Singapore in 2004. He also contributed in the organization of many local and international conferences. His areas of interest include control theory, robotics, obstacle avoidance, object recognition, artificial neural networks, signal/image processing, and computer vision.

About the Author—AJMAL S. MIAN received the BE degree in avionics from the College of Aeronautical Engineering, NED University, Pakistan, in 1993. He worked on a number of engineering and R&D projects related to radar data processing, communication jamming, and antijamming techniques before he was nominated for a masters degree. He received the MS degree in information security from the National University of Sciences and Technology, Pakistan, in 2003 and was awarded a PhD scholarship. He received the PhD degree in computer science from The University of Western Australia in 2006. He is currently a research fellow at the School of Computer Science and Software Engineering at The University of Western Australia. His research interests include computer vision, pattern recognition, multimodal biometrics, and information security.