3D Face Recognition by Projection Based Methods

Helin Dutağacı [(1)], Bülent Sankur [(1)], Yücel Yemez [(2)]

[(1)] Electrical and Electronic Engineering Department, Boğaziçi University, Bebek, İstanbul, Turkey
[dutagach, bulent.sankur]@boun.edu.tr
Telephone: (90) 212 359 6414, Fax: (90) 212 287 2465
[(2)] Computer Engineering Department, Koç University, Bebek, İstanbul, Turkey
yyemez@ku.edu.tr
Corresponding author: Bülent Sankur

ABSTRACT

In this paper, we investigate recognition performances of various projection-based features applied on registered 3D scans of faces. Some features are data driven, such as ICA-based features or NNMF-based features. Other features are obtained using DFT or DCT-based schemes. We apply the feature extraction techniques to three different representations of registered faces, namely, 3D point clouds, 2D depth images and 3D voxel. We consider both global and local features. Global features are extracted from the whole face data, whereas local features are computed over the blocks partitioned from 2D depth images. The block-based local features are fused both at feature level and at decision level. The resulting feature vectors are matched using Linear Discriminant Analysis. Experiments using different combinations of representation types and feature vectors are conducted on the 3D-RMA dataset.

Keywords: Face biometry, 3D face recognition, Independent Component Analysis, Nonnegative Matrix Factorization

1. INTRODUCTION

There are a number of challenges encountered with face recognition from 2D intensity images. In intensity images, faces acquired from the same person show high variability due to lighting conditions. Face segmentation from a cluttered background is another problem. Since 3D acquisition devices measure shape information, 3D face models are independent of lighting conditions. In addition, segmentation of 3D faces from background is relatively an easy task, for range images, as far as the face is within the range of the scanner. Furthermore, 3D face information can model small pose variations as opposed to intensity images.

The shape information of 3D faces is descriptive enough to distinguish between people. This information can either be used alone, or can be fused with 2D intensity information to increase recognition performance. In this work, we have used only 3D range images for identification purposes.

We can summarize the work on 3D face identification as follows: Phillips[1, 2, 3] et al. proposed a 3D face recognition system based on curvature calculation on range data. Tanaka et al.[4] utilize Extended Gaussian Image, which includes information of principal curvatures and their directions. Different EGIs are compared using Fisher's spherical correlation. Another work based on Extended Gaussian Image can be found in the paper of Lee et al[5]. Gordon[6], proposed a template-based recognition system, which again involves curvature calculation. Chua et al.[7], have used point signatures, a free form surface representation technique. Beumier et al.[8] proposed the use of face profiles for identification. They extracted central profile and lateral profile, and compared curvature values along these profiles. Chang et al.[9] applied Principal Component Analysis on 3D range data, together with the intensity images.

In this work we have utilized 3D face data registered by the algorithm described by Akarun et al.[10, 11, 12]. After registration, they have used the following methods for matching faces: Euclidean distance between point clouds, matching surface normals, principal component analysis, linear discriminant analysis and matching central and lateral profiles.

* *This work was partially supported by TÜBİTAK project 103E038, TÜBİTAK project 104E080 and BU Research Fund 05HA203.*

Security, Steganography, and Watermarking of Multimedia Contents VIII, edited by Edward J. Delp III, Ping Wah Wong,
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 6072, 60720I, © 2006 SPIE-IS&T · 0277-786X/06/$15

We propose three different representation schemes of 3D face information and a number of projection-based features, and compare their recognition performance. The representation schemes are 3D point cloud, 2D depth image and 3D voxel representation. Table 1, summarizes the proposed features extracted from these representations.

Table 1: Representation schemes and features used for 3D face recognition

Representation	Features
3D Point Cloud	▪ 2D DFT (Discrete Fourier Transform) ▪ ICA (Independent Component Analysis) ▪ NNMF (Nonnegative Matrix Factorization)
2D Depth Image	▪ Global DFT ▪ Global DCT ▪ Block-based DFT (Fusion at feature level) ▪ Block-based DCT (Fusion at feature level) ▪ Block-based DFT (Fusion at decision level) ▪ Block-based DCT (Fusion at decision level) ▪ ICA (Independent Component Analysis) ▪ NNMF (Nonnegative Matrix Factorization)
3D Voxel Representation	▪ 3D DFT (Discrete Fourier Transform)

The paper is organized as follows: Section 2 introduces the representation types of 3D face data. Section 3 describes the projection-based features and their extraction from different representation types. Section 4, briefly explains the distance measure between feature vectors. In Section 5, we give the experimental results. Finally we conclude with section 6.

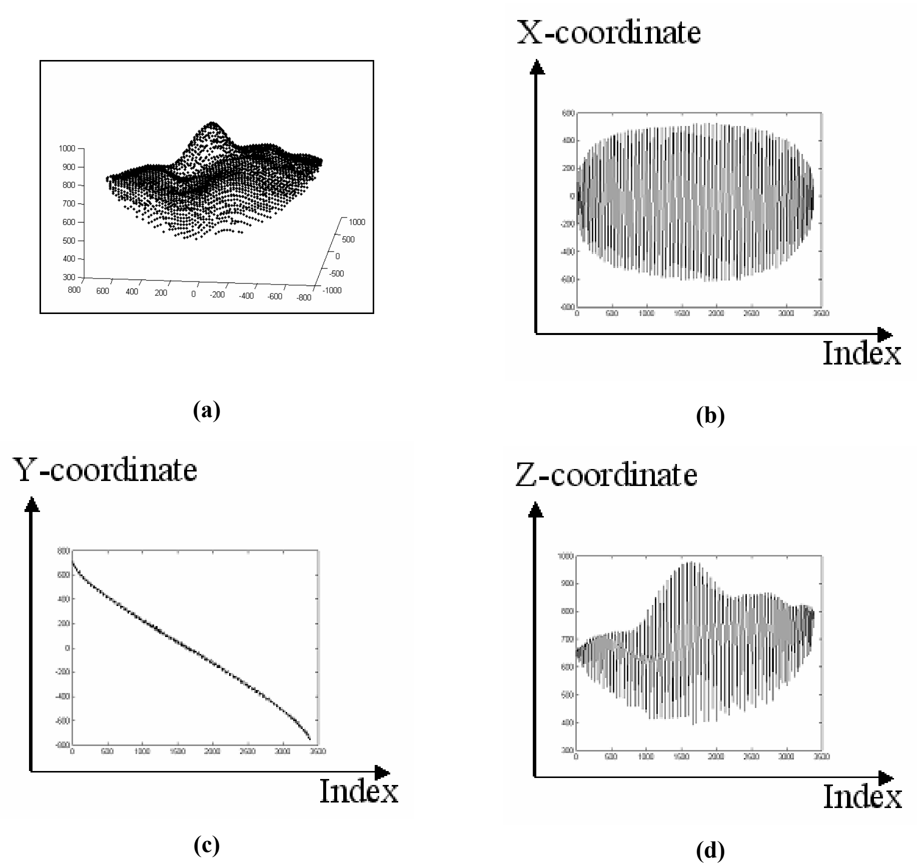
2. REPRESENTATION TYPES OF FACE DATA

In this work, we have compared three different representation schemes and extracted the features from these representations. These representation types are 3D point cloud, 2D depth image and 3D voxel representation. All these representations are derived from registered and cropped face data. The faces are registered using the ICP algorithm described by Akarun et al. [10, 11 ,12]

2.1. 3D point cloud

The 3D point cloud representation is the set of 3D coordinates, $\{x, y, z\}$ of the range data, obtained after registration. We have all the correspondences, defined at the registration process. Thus we can threat the ordered set of the coordinates as the signal describing the face. Figure 1.a shows a sample point cloud representation, while Figure 1.b, c and d show the x, y and z vectors respectively, as a function of the vector index.

If we have N points in the face data, we can concatenate the x, y and z coordinates and obtain a one-dimensional signal of length 3N. Another way of arranging the set of coordinates is to form an Nx3 matrix, where each dimension is placed into one of the columns. We choose one of these two arrangements of the data, depending on the feature type we would like to estimate.



(a)

(b)

(c)

(d)

Figure 1: (a) Point cloud representation. **(b, c, d)** x , y and z vectors respectively, as a function of the vector index.

2.2. 2D depth image

2D depth image is a commonly used representation type for face recognition. It is sometimes called as 2 ½ D image since it encodes 3D information. The point cloud is placed onto a regular X-Y grid, and the Z coordinates are mapped onto this grid to form the depth image $I(x, y)$ (Figure 2). This representation type is similar to intensity images by structure, therefore many techniques applied on intensity images can be also applied to $I(x, y)$. We have tested the following descriptors, which were previously applied to 2D intensity images, with the depth images: DFT, DCT, block-based versions of DFT and DCT, Independent Component Analysis (ICA) and Nonnegative Matrix Factorization (NNMF).

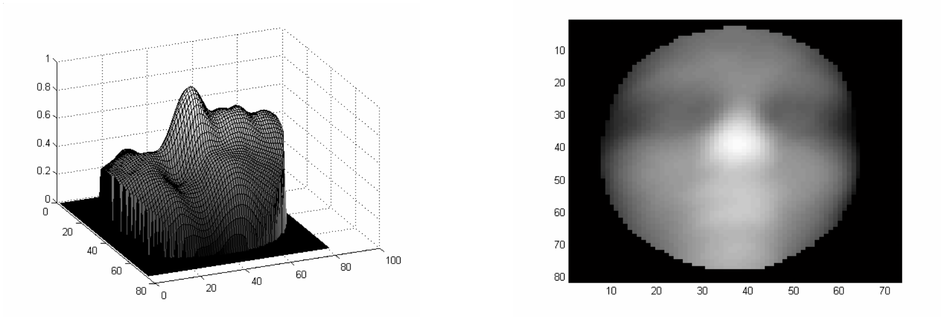


Figure 2: 2D depth images from side and from top.

2.3. 3D voxel representation

3D voxel representation can be regarded as a function $V_d(x,y,z)$, filling the 3D space. To obtain this function, we implement the following steps: The point cloud is placed in an NxNxN voxel grid. The center of this voxel grid should coincide with the center of mass of the point cloud. Then we define a binary function $V(x,y,z)$ on the voxel grid. If, in a particular voxel at location (x,y,z), there does not exist any point from the face, then $V(x,y,z)$ at that voxel is set to zero. Otherwise, $V(x,y,z)$ gets a value of one. Figure 3 shows a sample point cloud, and the corresponding binary $V(x,y,z)$ displayed as a negative image.

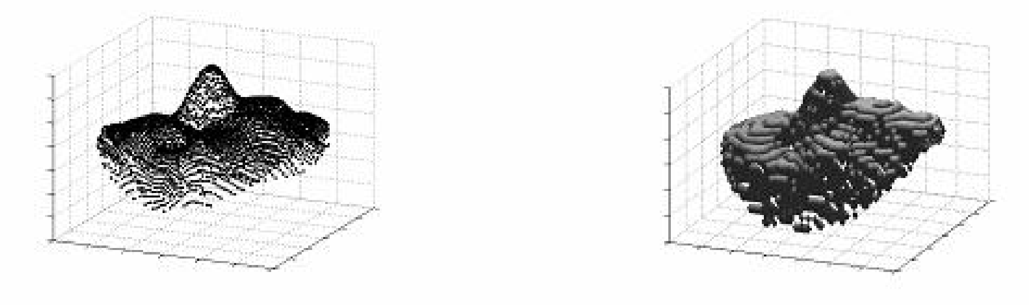


Figure 3: Point cloud and its binary voxel representation.

After the 3D binary function is obtained, we apply 3D distance transform on this binary function to get $V_d(x,y,z)$. This function gets a value of zero on the face surface, and the value increases as we get further away the surface. By using the distance transform we distribute the shape information of the surface throughout the 3D space and obtain a richer representation. Figure 4 gives slices from the voxel representation based on the distance transform.

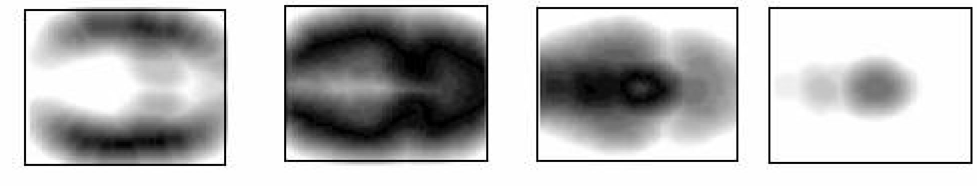


Figure 4: Slices from the voxel representation based on the distance transform.

3. FEATURES FOR FACE RECOGNITION

In Table 1, we have summarized the features to be compared with respect to their face recognition performance. These features can be grouped into four categories: Global DFT/DCT-based features block DFT/DCT-based features, ICA coefficients and NNMF-based features.

3.1. Global DFT/DCT

In order to compute DFT coefficients from the point cloud of N points, we first define an Nx3 matrix, **P**, and replace coordinates of each dimension to one column of **P**:

$$\mathbf{P}_{Nx3} = [\mathbf{X}_{Nx1} \quad \mathbf{Y}_{Nx1} \quad \mathbf{Z}_{Nx1}]$$

This matrix can be regarded as a 2D function, and we apply 2D DFT on this function. We could have concatenated the X, Y and Z coordinates and computed the one-dimensional DFT, however, then we would lose the inherent relation within the coordinates of a point in the face. DFT coefficients are strongly dependent on the order of the data, and we intended to keep the X, Y and Z coordinates of a point, close in the data structure. The 2D-DFT coefficients of **P** are then computed as follows:

$$\mathbf{FP}_{ij} = \mathbf{DFT}\{\mathbf{P}\}_{ij} = \sum_{n=1}^{N} \sum_{d=1}^{3} \mathbf{P}_{nd} \exp(\frac{2\pi ni}{N}) \exp(\frac{2\pi dj}{3})$$

FP is a matrix of size Nx3. We take the first K coefficients of the first column of this matrix, and obtain a feature vector of size 2K – 1 by concatenating the real and imaginary parts of the K complex coefficients. Figure 5 shows a sample DFT-based feature vector of the point cloud. One should note that, most of the energy is concentrated in the band-pass region due to the zigzag scan of the face as can be observed from the plots of the coordinates in Figure 1. One future work can be investigation of the appropriate band that will give superior recognition results.

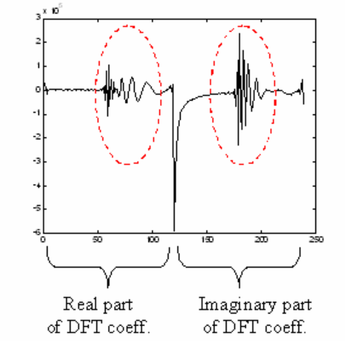


Figure 5: Sample DFT-based feature vector obtained from point cloud.

In order to obtain global DFT-based features from the depth image, we apply 2D-DFT to the function $I(x, y)$. The resulting DFT coefficients are of the same size with the depth image. We extract the first KxK coefficients of this matrix and obtain a feature vector of size 2K^2 – 1, by concatenating the real and imaginary parts (Figure 6). Likewise, we get the global DCT-based features; however, in this case we obtain a feature vector of size K^2 since DCT coefficients are real.

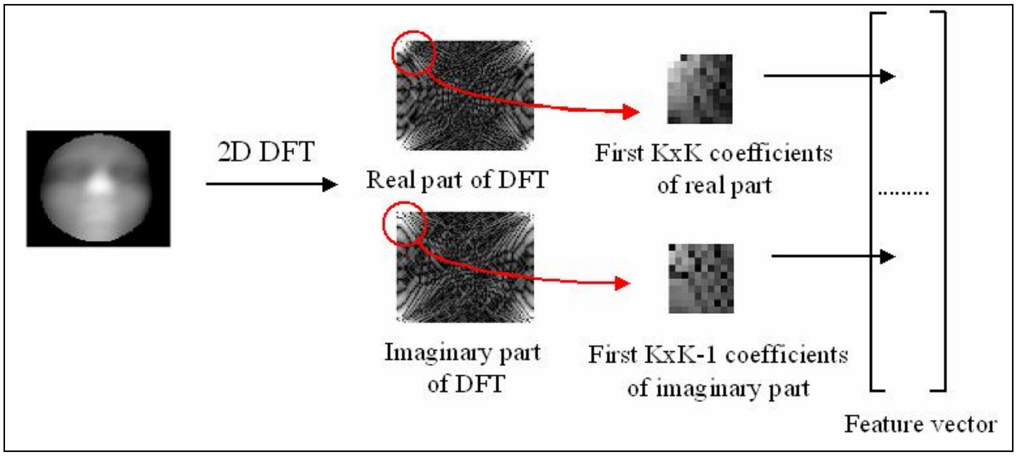

2D DFT
Real part of DFT
Imaginary part of DFT
First KxK coefficients of real part
First KxK-1 coefficients of imaginary part
....
Feature vector

Figure 6: Extraction of global DFT-based features from depth image.

We also extract DFT-based descriptors from the voxel representation. We compute the 3D-DFT coefficients of the distance transform function $V_d(x, y, z)$, and extract the first KxKxK coefficients to form the feature vector. By concatenating the real and imaginary parts, we obtain a feature vector of size 2K^3 – 1 (Figure 7).

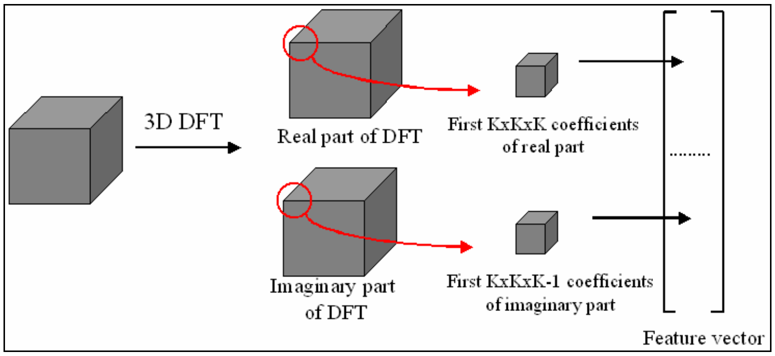

3D DFT
Real part of DFT
Imaginary part of DFT
First KxKxK coefficients of real part
First KxKxK-1 coefficients of imaginary part
....
Feature vector

Figure 7: Extraction of global DFT-based features from voxel representation.

3.2. Block-based DFT/DCT

In addition to the global DFT/DCT-based techniques, we also extract local features, based on the calculation of DFT coefficients on blocks. The depth images are partitioned into blocks of size MxM and 2D-DFT is applied separately to each block. Then we take the first KxK DFT coefficients to form the feature vector special to a particular block. We can then fuse this data either at feature level, or at decision level.

Fusion at feature level is performed by concatenating the DFT coefficients coming from the blocks in a single vector. Figure 8 gives a sample feature vector obtained by fusion at feature level and Figure 9 explains the procedure.

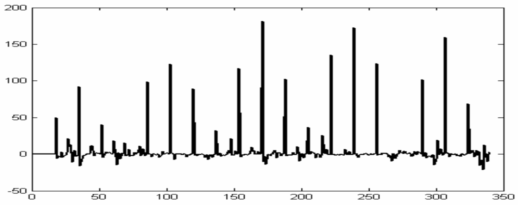


Figure 8: Sample block-based feature vector obtained from fusion at feature level.

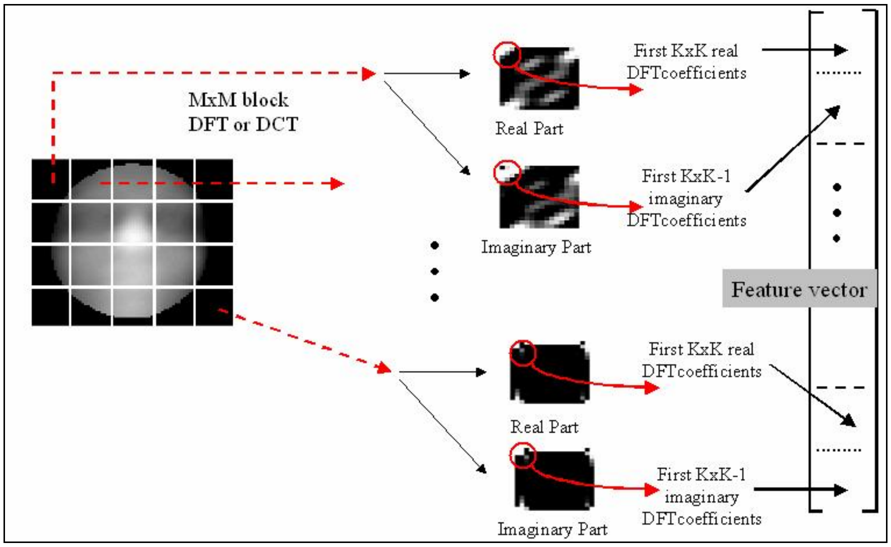

MxM block
DFT or DCT
Real Part
Imaginary Part
First KxK real DFTcoefficients
First KxK-1 imaginary DFTcoefficients
Real Part
Imaginary Part
First KxK real DFTcoefficients
First KxK-1 imaginary DFTcoefficients
....
Feature vector

Figure 9: Procedure for fusion at feature level.

We perform fusion at decision level by using the sum rule. The depth image of an input face to be recognized is partitioned into blocks and each block is matched with the corresponding blocks of the depth images in the database. From this comparison, each face in the database gets a rank. A face in the database, thus obtains rank values as many as the number of blocks. When we sum up the ranks, we obtain the final rank for the face, and choose the identity of the face with the lowest final rank. Figure 10 summarizes this procedure.

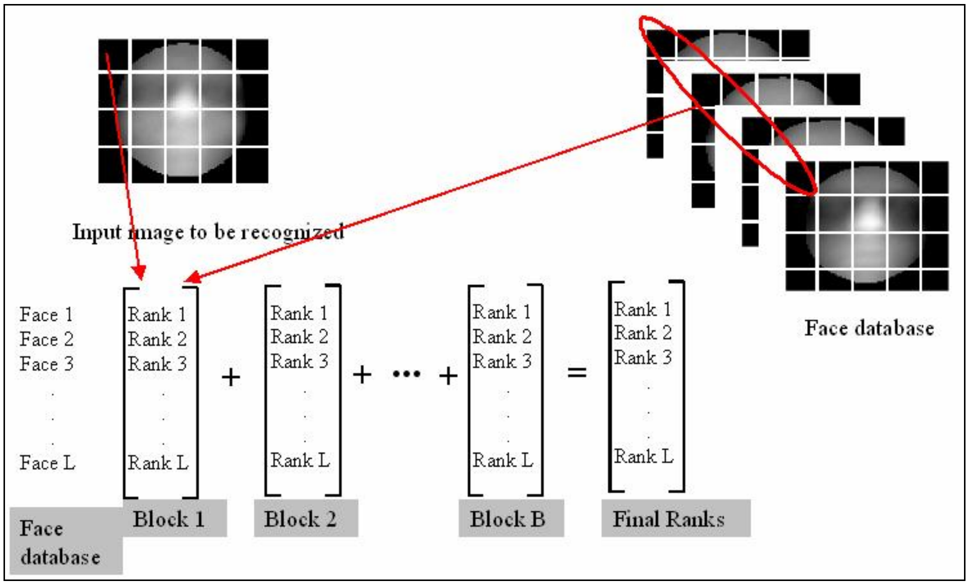


Figure 10: Procedure for fusion at decision level.

3.3. Independent Component Analysis (ICA)

Independent component analysis is a technique to extract independent variables from a mixture of them. In this work, we assume that each point of a face is a mixture of variables from independent sources. Let $\mathbf{X}$ be the data matrix, where each column includes the data from one face, then we can represent $\mathbf{X}$ as follows:

$$\mathbf{X} = \mathbf{AS}$$

where $\mathbf{A}$ is the mixing matrix. The columns of $\mathbf{A}$ form a basis for the face database. On the other hand, the columns of $\mathbf{S}$ are independent coefficients and they form feature vectors for the corresponding faces. We have utilized the FastICA algorithm described by Hyvarinen and Oja[13] to extract the matrices $\mathbf{A}$ and $\mathbf{S}$.

We extracted ICA-based features from two different representations. For the point cloud, the (x, y, z) coordinates are concatenated to form a one-dimensional vector. The dimensionality of these vectors is then reduced using Principal Component Analysis. The data matrix $\mathbf{X}$ is constructed by placing the PCA coefficients of each face to the columns. Figure 11 shows the first three basis faces obtained by ICA algorithm, back projected to principal components. The basis faces are plotted on top of the mean face.

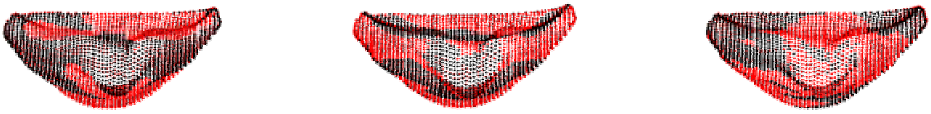
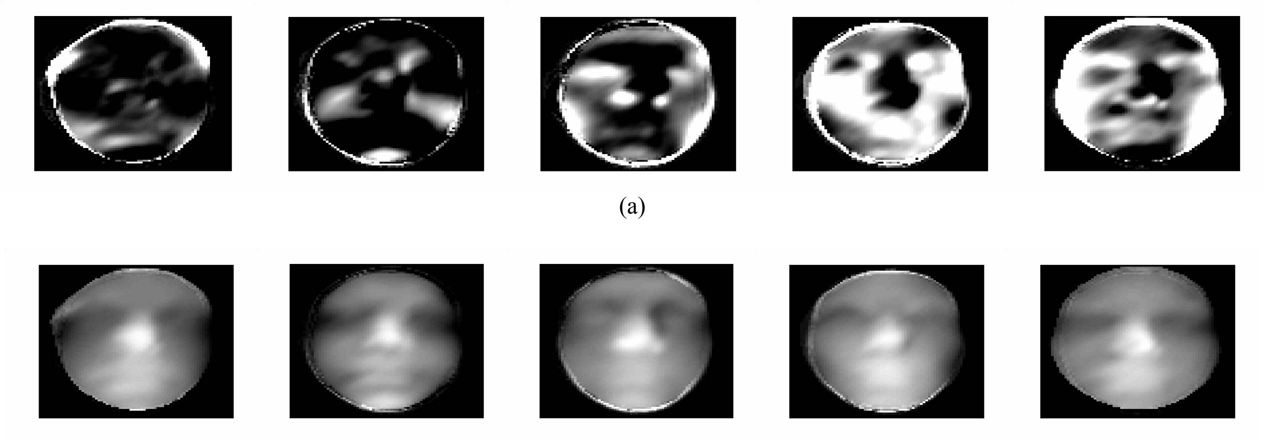


Figure 11: First three basis faces obtained from ICA on point cloud.

For depth images, we follow a similar procedure. The columns of a depth image are concatenated to form a single one-dimensional vector. Then PCA is applied to the face database and ICA-based features are derived from the PCA coefficients of the faces. Figure 12 (a) shows the first five basis faces obtained from ICA of the depth images, and Figure 12 (b) illustrates the same basis functions summed with the mean face.


(a)

(b)

Figure 12: (a) First five basis faces obtained from ICA on depth images. **(b)** Basis faces plotted on top of mean face.

3.4. Nonnegative Matrix Factorization (NNMF)

Given a non-negative matrix $\mathbf{X}$ of size NxL, nonnegative matrix factorization tries to obtain two non-negative matrices $\mathbf{W}$ and $\mathbf{H}$ such that,

$$\mathbf{X} \approx \mathbf{WH}\,,$$

where $\mathbf{W}$ is of size NxR and $\mathbf{H}$ is of size RxL. Since we force the two matrices to be non-negative, we can only get an approximation of $\mathbf{X}$ from the product of them. The columns of $\mathbf{W}$ can be regarded as basis vectors and the columns of $\mathbf{H}$ are utilized as feature vectors of the corresponding faces. $\mathbf{W}$ and $\mathbf{H}$ are obtained using the multiplicative update rules described by Lee and Seung[14]. The update rules that minimize:

$$\|\mathbf{X}-\mathbf{WH}\| = \sum_{il}(\mathbf{X}_{il}-[\mathbf{WH}]_{il})^2$$

are

$$\mathbf{H}_{a\mu} \longleftarrow \mathbf{H}_{a\mu}\frac{(\mathbf{W}^{\mathbf{T}}\mathbf{X})_{a\mu}}{(\mathbf{W}^{\mathbf{T}}\mathbf{WH})_{a\mu}}\,, \text{ and} \qquad \mathbf{W}_{ia} \longleftarrow \mathbf{W}_{ia}\frac{(\mathbf{XH}^{\mathbf{T}})_{ia}}{(\mathbf{WHH}^{\mathbf{T}})_{a\mu}}\,.$$

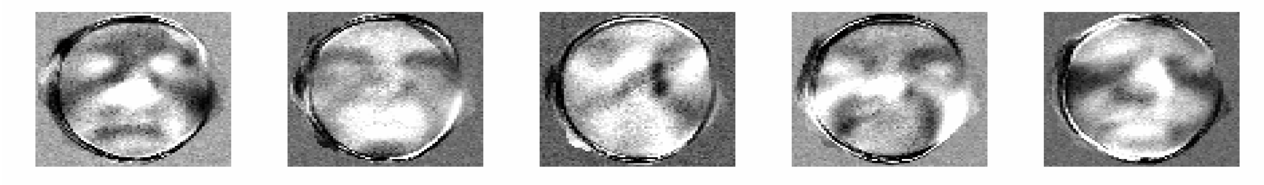


Figure 13: First five basis faces obtained from NNMF on depth images.

To construct the data matrix $\mathbf{X}$, we either use the point cloud representations or the depth images. In either case, we place the data of each face into the columns of the data matrix. Figure 13 shows the first five basis faces obtained from NNMF of the depth images. We can observe that local features are encoded by the basis faces, which is a basic characterization of NNMF. Since the nonnegativity constraints only allow additive combinations, NNMF provides a parts-based representation.[15]

4. MATCHING FEATURES

We use linear discrimination for classifying an input feature vector. We estimate the covariance matrix of the feature vectors in the training set and fit a multivariate normal density to each class (person) using this global covariance matrix. When there is an input face to be recognized, the feature vector of the face is extracted and the Mahalanobis distances of the input feature vector to the class centers are calculated. The class giving the smallest Mahalanobis distance is chosen as the identity of the input face.

5. EXPERIMENTAL RESULTS

We have used the 3D-RMA face database[16] for comparing the schemes discussed above. The 3D-RMA database contains face scans of 106 subjects. The total number of faces is 617 and there are 5 to 6 sessions per person. We have used 4 sessions for training (424 face scans) and utilized the rest 193 faces for test. We have conducted 5 experiments by selecting different combinations of the sessions. Table 2 gives the identification results of all the schemes, averaged over the 5 experiments. Table 2 also provides information about the number of features selected for each scheme.

Table 2: Recognition performances and number of features

Representation	Features	Number of features	Performance
3D Point Cloud	2D DFT	2x400-1 **(799)**	95.86
	ICA	**50**	**99.79**
	NNMF	**50**	**99.79**
2D Depth Image	Global DFT	2x8x8-1 **(127)**	98.24
	Global DCT	11x11 **(121)**	96.58
	Block-based DCT (Fusion at feature level)	20x20 blocks (12 blocks), 2x2x2-1 for each block **(84)**	98.76
	Block-based DCT (Fusion at feature level)	20x20 blocks (12 blocks) 3x3 for each block **(108)**	98.24
	Block-based DFT (Fusion at decision level)	20x20 blocks (12 blocks), 4x4-1 for each block **(180)**	98.13
	Block-based DCT (Fusion at decision level)	20x20 blocks (12 blocks) 6x6 for each block **(432)**	97.82
	ICA	**50**	96.79
	NNMF	**50**	94.43
3D Voxel Representation	3D DFT	2x4x4x4-1 **(127)**	98.34

As can be observed from Table 2, ICA and NNMF-based features extracted from the point cloud representations of faces gave superior results with smallest number of features. They gave 100 per cent recognition performance for the three experiments and missed only one face for the two experiments. The missed face is plotted on top of another face of the same person in Figure 14. The misclassification is due to the inaccurate registration of this particular face.

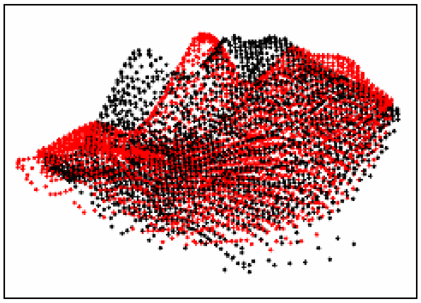


Figure 14: The misclassified face plotted on top of another face of the same person (misclassified by ICA and NNMF – based features computed on point cloud representation).

6. CONCLUSION

In this work, several feature types are proposed for the recognition of pre-registered 3D face data. The features are extracted from three different face representations of the face data. Experimental results show that the point cloud representation along with the ICA-based or NNMF-based features gave superior results, 99.8 per cent recognition performance. This is due to the fact that with point cloud representation we have the minimum amount of data loss; however, to obtain the depth images or voxel representations we quantize the 3D coordinates of the faces. On the other hand, ICA and NNMF have the ability to extract the essence of the information present in the large data matrices.

Since all the schemes described in this work yielded recognition performances higher than 94 per cent, all of them are worth further improvement. Several fusion methods at both feature and decision levels can be applied for block-based DFT-DCT methods. The appropriate frequency bands can be investigated to obtain better performance with DFT-based schemes. Finally, these features should be tested on larger databases for both recognition and verification problems.

REFERENCES

1. Phillips, P., Grother, P., Micheals, R., Blackburn, D., Tabassi, E. and Bone M., "Face Recogntion Vendor Test 2002", NIST Technical Report, NIST IR 6965, March 2003.
2. Rizvi, S., Phillips, P. and Moon, H., "The FERET Verification Testing Protocol for Face", NIST Technical Report, NIST IR 6281, October 1998.
3. Phillips, P., Moon, H., Rizvi, S. and Rauss, P., "The FERET Evaluation Methodology for Face-Recognition Algorithms", IEEE Trans. Pattern Analysis and Machine Intelligence, 22:1090-1103, 2000.
4. H. T. Tanaka, M. Ikeda, and H. Chiaki. Curvature based face surface recognition using spherical correlation - principal directions for curved object recognition. In Proceedings of Int. Conf. on Automatic Faceand Gesture Recognition, pages 372–377, 1998.
5. Lee, J. C. and Milios, E. "Matching Range Images of Human Faces", In Proc. IEEE International Conference on Computer Vision, Page 722-726, 1990.
6. Gordon, G., Vincent, L., "Application of Morphology to Feature Extraction for Face Recognition", In Proc. of SPIE, Nonlinear Image Processing, San Jose, Feb. 1992. Vol. 1658.
7. Chua, C.S. Han, F. and Ho, Y. K., "3D Face Recognition Using Point Signature", In Proc. IEEE International Conference on Automatic Face and Gesture Recognition, Pages 233-238, March 2000.
8. Beumier, C., Acheroy, M., "Face Verification from 3D and grey-level Clues", Pattern Recognition Letters, Vol. 22, 2001, pp 1321-1329.
9. Chang, K., Bowyer, K., Flynn, P., "Face Recognition Using 2D and 3D Facial Data", In IEEE International Workshop on Analysis and Modeling of Faces and Gestures. Nice, France. Oct. 2003.
10. Akarun, L., Gökberk, B., Salah A.A., "3D Face Recognition for Biometric Applications", European Signal Processing Conference, EUSIPCO, Antalya, September 2005.
11. Gökberk, B., Salah A.A., Akarun, L., "Rank-based Decision Fusion for 3D Shape-based Face Recognition", Audio- and Video-based Biometric Person Authentication (AVBPA), Tarrytown, New York, July 2005.

12. İrfanoğlu, M. O., Gökberk, B., Akarun, L., "3D Shape-based Face Recognition using Automatically Registered Facial Surfaces", Proceedings of the 17th International Conference on Pattern Recognition (ICPR2004), 2004, Cambridge.
13. Hyvarinen A. and Oja E., "Independent Component Analysis: Algorithms and Applications," Neural Networks 13 (4-5), 411-430, 2000.
14. Lee, D.D. and Seung H. S., "Algorithms for Nonnegative Matrix Factorization", Advances in Neural Information Processing Systems 13. 2001.
15. Lee, D.D. and Seung H. S., "Learning the Parts of Objects by Non-Negative Matrix Factorization", Nature, 401:788, 1999.
16. http://www.sic.rma.ac.be/~beumier/DB/3d_rma.html