Proceedings of the
2007 IEEE Intelligent Vehicles Symposium
Istanbul, Turkey, June 13-15, 2007

Driver Face Recognition by Using 3D Discrete Cosine Transform

Goksel Gunlu, Hasan Sakir Bilge

***Abstract*—Next generation vehicles are foreseen to use biometric recognition systems for person authentication. One of them is face recognition. In this study, a different approach of feature vector generation is presented for a face recognition system. To generate the feature vectors, DCTfaces, which are very similar to Eigenfaces, are used. DCTfaces are obtained by using 3D-DCT of face images after an image synthesis and 2D-Inverse DCT of the 3D DCT coefficient layers. Taking the advantage of DCT image compression capability, very few terms are used for these DCTfaces. Experimental results show that, DCTfaces can be successfully used for feature vector generation in face recognition.**

I. Introduction

To make the vehicles safer and comfortable, several driving assistant methods are suggested [1]-[3]. For vehicle safety, driver should be authorized by biometric person recognition systems, such as fingerprint, iris or face recognition.

Face recognition attracts more and more attention for access control applications due to not requiring user interaction. However current state of art in face recognition is not yet sufficient for the more demanding applications [4]. The most important parameter for biometric recognition systems is the recognition performance. Many researchers are working for improving the recognition performance. Some of them use 2D intensity images or 3D data, while others use both of them. Detailed description can be found in related papers [4]. Due to its correspondence to our study, we only mention below the Eigenface method, which is one of the most widely used.

Eigenface method is suggested by Turk and Pentland [5]. Eigenfaces, which are the eigenvectors of the covariance matrix of training images, are used to project each face image onto eigen space. So each face image is represented as a feature vector in the face space. Then, a distance metric is just used to classify the feature vector of a test image to its closest feature vector of training images.

Our method transforms faces into characteristic feature images that we call them as DCTfaces, which are 2D-Inverse Discrete Cosine Transforms (IDCT) of each layer obtained from 3D-DCT of training images. Inspiring by Eigenface method, we process DCTfaces in the face recognition process with the same manner that Eigenfaces are used in recognition. A given test image is projected into face space by using DCTfaces. So a feature point is produced for each DCTfaces. Feature vectors, which are combinations of feature points, are used for recognition.

Face recognition includes several scenarios [4], [9]. One of them is recognition or identification and another is authentication or verification.

In this study, recognition scenario is applied for vehicle applications. Limited numbers of drivers are permitted to drive a vehicle; therefore there are a few of training face images of authorized drivers. When an unknown person requests to access to service in vehicle, identity is determined by suggested method. If the identity is an authorized person, his access to vehicle services is granted; if not, rejected.

Manuscript received January 15, 2007.

Goksel Gunlu is with the Electrical-Electronics Engineering Department, Gazi University, Turkey (e-mail: goksel@gazi.edu.tr).
Hasan Sakir Bilge is with the Computer Engineering Department, Gazi University, Turkey (e-mail: bilge@gazi.edu.tr).

II. Eigenfaces and DCT

A. Eigenface

In Eigenface method, the original image with NxN dimension is firstly converted to a one-dimensional vector (N^2x1) [5]. A series of vectors of different images are subject to Principal Component Analysis (PCA) that finds the vectors which best represent the distribution of the faces. A small number ($M \ll N^2$) of eigenvectors with largest eigenvalues are selected to project face images onto eigen space. Each face image is then represented by a linear combination of these eigenvectors, also named as Eigenfaces. Thus each NxN dimensional face image is described with Mx1 dimensional feature vector. These feature vectors are used for matching. Distance metric is used to classify a feature vector of a test image to its closest feature vector of training images. Minimum distance between the test vector and other vectors in the database gives the correct correspondence.

B. 2D DCT and Face Recognition

2D-DCT has been used in several face recognition applications for dimension reduction and feature extraction, because of its compression capability [6], [7], [10]. Compressed images preserve perceptually relevant information at small size. So complexity is enormously reduced in face image without losing any crucial information. Some of the methods use these low dimensional data as feature vectors in classification [8], [10], [11].

C. PCA and DCT

There are several studies about relation between Principal Component Analysis (PCA) and DCT [12], [13].

For dimension reduction, PCA is the most optimal solution. On the other hand, DCT is asymptotically equivalent to the PCA (or Karhunen–Loeve Transform) for Markov-1 signals with an inter-element correlation coefficient close to one [12]. DCT performs very close to the statistically optimum Karhunen–Loeve transform in terms of compression performance. However DCT has some advantages over PCA such as input independency of transform matrix and lower computational complexity.

In this study, face recognition performances of both methods have been experimentally evaluated.

D. 3D DCT

3D DCT is the extension of 2D DCT and recently used for stereo or video image compression, and motion estimation [14], [15]. It exploits inter-image correlation in addition to inter-pixel correlation.

3D-DCT of a data volume is defined as [15]:

$$C(u,v,w) = \frac{2\sqrt{2}}{\sqrt{MNL}}.\alpha(u)\alpha(v)\alpha(w) \sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\sum_{z=0}^{L-1} f(x,y,z)g(x,u,M)g(y,v,N)g(z,w,L) \quad (1)$$

where *f(x,y,z)* is the voxel intensity value at position *x,y,z* of the image data volume, *C(u,v,w)* is the DCT coefficient value at position *u,v,w* of the transformed cube, M,N,L are dimensions of the cube, and

$$\alpha(p) = \begin{cases} \frac{1}{\sqrt{2}}, & p = 0 \\ 1, & others \end{cases} \text{ and } g(r,t,K) = \cos\left[\frac{(2r+1)t\pi}{2K}\right].$$

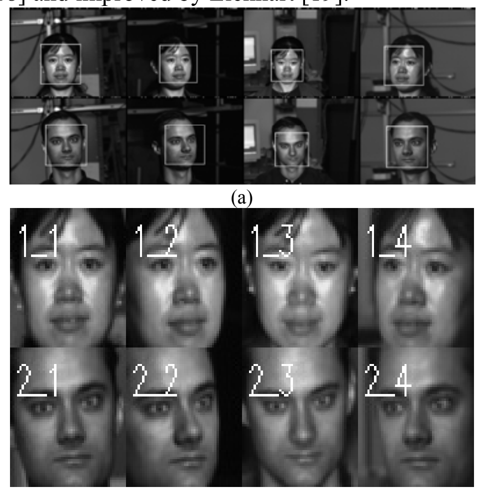
Since 3D-DCT is separable, it can be calculated by row-column decomposition. Some faster algorithms have been suggested such as 3D vector-radix decimation-in-frequency (3D VR DIF) algorithm [16]. In this study, 3D-DCT is calculated by getting firstly 2D-DCT of layers and then by applying 1D-DCT, due to having fast 2D-DCT algorithms.

III. Proposed Method

A. Face Detection

The first step of any face processing system is detecting the face locations in the images [17]. Face detection affects highly the recognition performance. Given a single image, the goal of face detection is to identify all sub image regions that contain a face regardless of its position, orientation, and lighting conditions [17].

Although skin-tone based face detection is the fastest, they are not suitable for variant illumination conditions such as vehicle interior [2]. Face detection is out of scope of this study. Therefore, a previously proposed face detection algorithm is used; this effective method based on a boosted cascade of simple Haar-feature classifier is introduced by Viola [18] and improved by Lienhart [19].



(a)

(b)

Fig. 1. (a) Some of CMU_PIE face database used for performance test. (b) Detected and scaled face images for face recognition stage.

As shown on the Fig. 1, after getting a scene image from camera, the location of the faces are firstly determined. And only face region is selected. Face region is scaled to a predetermined size, because all input images have to be in the same dimension for the proposed recognition process. All the faces in this study are firstly detected in an image and then appropriately scaled.

B. Calculation of the DCTfaces

For the face recognition, we should firstly model the known faces in a special feature space. In our method, we put L faces with size MxN of authorized drivers together by placing each of them on top of the other. Hence we obtain a 3D data structure, whose each layer is a face image (Fig. 2.a). This 3D matrix (MxNxL) has the same width (M) and length (N) with input images, and its height (L) is equal to the number of input images.

Then, 3D DCT of this 3D matrix is calculated. The output is a 3D form of real DCT coefficients, with the same dimension (MxNxL) of the input. Each layer of the output has a different characteristic about the faces. For example, the first layers (MxN) correspond to lower frequencies. By taking the 3D inverse DCT of the 3D DCT matrix, original images can be correctly obtained. At this point, each 2D layer of the 3D DCT matrix contributes to each face in some extent. To see this contribution, we apply 2D Inverse DCT to each 2D layer of the 3D DCT matrix. At the end of this process, we obtain L images. When we look carefully these semi-original images, we conclude that these are very similar to Eigenfaces. And it is obvious that each original image is reconstructed from these semi-original images. We call these images as DCTfaces. The similarity of these DCTfaces to Eigenfaces gives us the idea of processing DCTfaces like Eigenfaces. The following operations are the same as in the recognition by Eigenfaces.

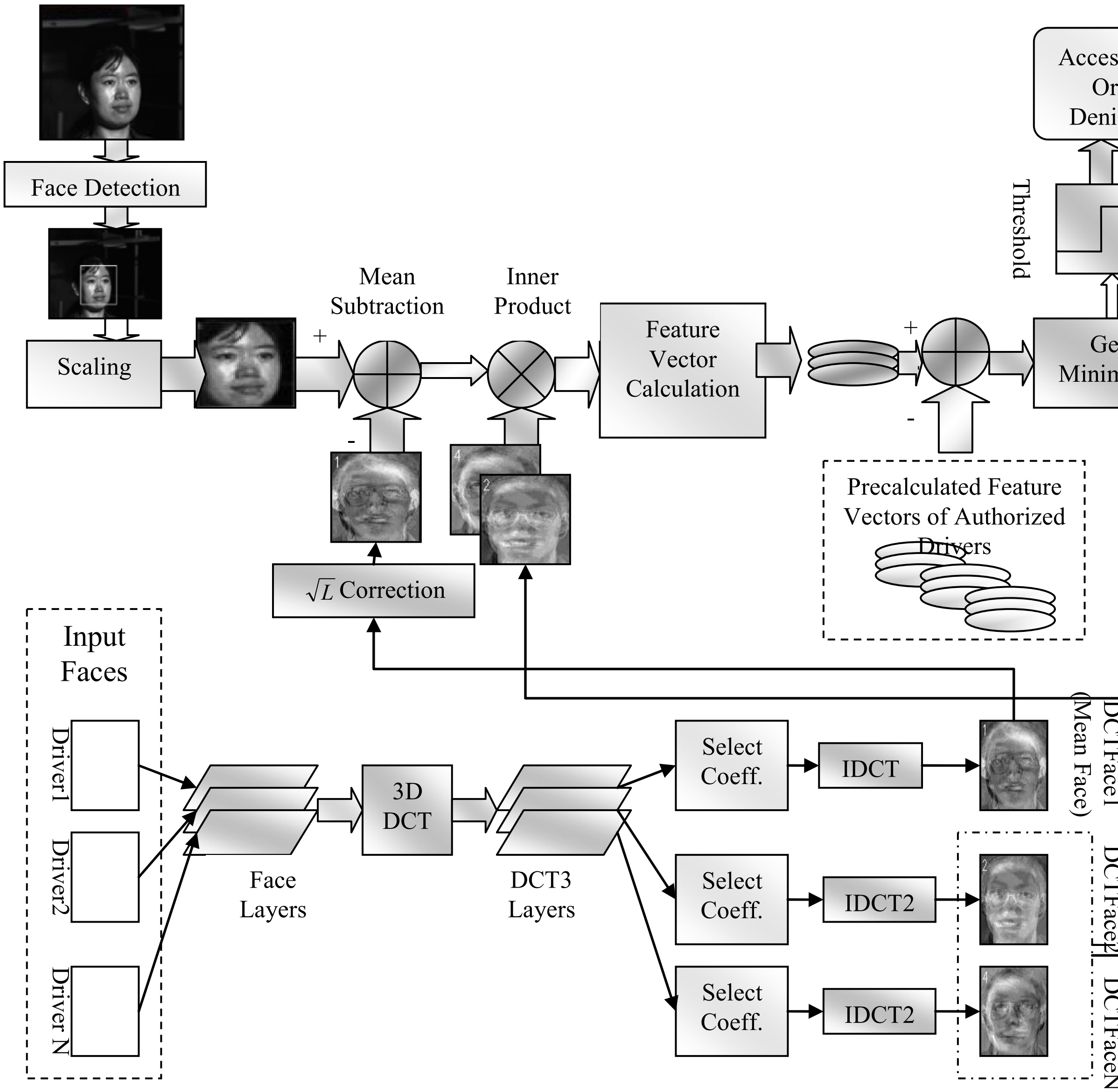


Fig. 2. (a) Calculation of DCTfaces (bottom part). (b) Face recognition process (top part).

C. Calculation of the Feature Vectors

Feature vectors for each original face are calculated by inner product of the face image and a series of DCTfaces. Thus, each face is described as a weighted combination of DCTfaces. All feature vectors of authorized drivers are stored in a database in order to be used in subsequent recognition process.

When a new face image comes, firstly the mean image of the all authorized drivers is subtracted from the new image. In fact, this mean image is the first DCTface image. This can be easily seen from the below calculations.

3D DCT formula is given in (1); here if we take w = 0, first layer of the 3D DCT matrix can be obtained as (2);

$$C(u,v,w=0)=\frac{2\sqrt{2}}{\sqrt{MNL}}.\alpha(u)\alpha(v)\left(\frac{1}{\sqrt{2}}\right) \sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\sum_{z=0}^{L-1} f(x,y,z)\cos(x,u,M)\cos(y,v,N)\cos(z,w=0,L) \tag{2}$$

$$C(u,v,w=0)=\frac{2}{\sqrt{MN}}.\alpha(u)\alpha(v) \sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\cos(x,u,M)\cos(y,v,N)\frac{1}{\sqrt{L}}\sum_{z=0}^{L-1} f(x,y,z) \tag{3}$$

using (3),

$$C(u,v,w=0)=\frac{2}{\sqrt{MN}}.\alpha(u)\alpha(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f_z'(x,y)\cos(x,u,M)\cos(y,v,N) \quad (4)$$

2D DCT and 2D Inverse DCT formula are given as;

$$C(u,v)=\frac{2}{\sqrt{MN}}.\alpha(u)\alpha(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\cos(x,u,M)\cos(y,v,N) \quad (5)$$

$$f(x,y)=\frac{2}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\alpha(u)\alpha(v)C(u,v)\cos(x,u,M)\cos(y,v,N) \quad (6)$$

From equation (4) and (5) we get,

$$C(u,v,w=0)=C_z'(u,v) \quad (7)$$

Take 2D IDCT of equation (7),

$$\begin{aligned} idct2\{C(u,v,w=0)\} &= idct2\{C_z'(u,v)\} \\ idct2\{C(u,v,w=0)\} &= f_z'(x,y) \\ idct2\{C(u,v,w=0)\} &= \frac{1}{\sqrt{L}}\sum_{z=0}^{L-1}f(x,y,z) \end{aligned} \quad (8)$$

If both side of the equation (8) is multiplied by $\frac{1}{\sqrt{L}}$,

$$\frac{1}{\sqrt{L}}idct2\{C(u,v,w=0)\}=\frac{1}{\sqrt{L}}\frac{1}{\sqrt{L}}\sum_{z=0}^{L-1}f(x,y,z) \quad (9)$$

$$=\frac{1}{L}\sum_{z=0}^{L-1}f(x,y,z)$$

It can be seen that, the first DCTface image is the mean of the L images in z direction, scaled by a factor of $\sqrt{L}$.

After the mean (first DCTface scaled by a constant) is subtracted from new face image, the difference image is multiplied with each DCTface point by point and summed (inner product). This results in test feature vector,

$\Omega^T = \{w_1, w_2, \ldots, w_{L-1}\}$. (10)

This test feature vector should be compared with feature vectors in the database in order to find any match.

D. Matching

Matching is another important stage in the face recognition. For this purpose, three different distance metrics may be used, formulas for these norm types are given as;

$$\text{Norm 1: } \frac{\|fv_a - fv_b\|_C}{\|fv_b\|_C}=\frac{\max(|fv_{ai}-fv_{bi}|)}{\max(|fv_{bi}|)} \quad (11)$$

$$\text{Norm 2: } \frac{\|fv_{a1} - fv_{b1}\|_{L1}}{\|fv_{b1}\|_{L1}}=\frac{\sum_{i=1}^{N}|fv_{ai}-fv_{bi}|}{\sum_{i=1}^{N}|fv_{bi}|} \quad (12)$$

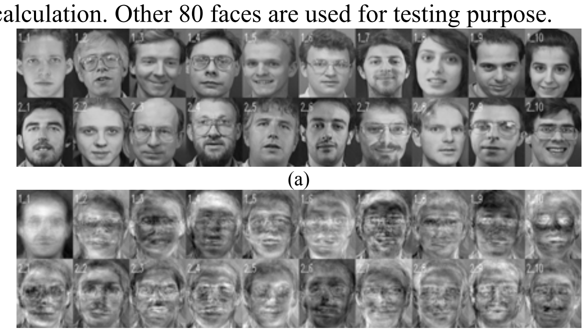
$$\text{Norm 3: } \frac{\|fv_{a1} - fv_{b1}\|_{L2}}{\|fv_{b1}\|_{L2}}=\frac{\sqrt{\sum_{i=1}^{N}(fv_{ai}-fv_{bi})^2}}{\sqrt{\sum_{i=1}^{N}(fv_{bi})^2}} \quad (13)$$

Minimum distances between the feature vector of the test image and other feature vectors in the database are calculated according to these formulas. In other words, the closest face of authorized drivers to the test face is searched in the database. If the minimum distance is smaller than a predefined threshold, this means that test image is found in the database. Then, this driver is granted to access any service. If the minimum distance is greater than the threshold, we assume that test image isn't found in the database. Thus, this driver is rejected. If the threshold is taken very small, then some faces are mistakenly rejected. If the threshold is taken large, then the recognition ratio is decreased. As a result, the amplitude of the threshold determines the trade-off between false rejection ratio and recognition ratio.

IV. Experimental Study and Results

In this study, the proposed method is evaluated by using CMU PIE and ORL face databases.

In ORL face database, for some subjects, the images were taken at different times, varying lighting slightly, facial expressions (open/closed eyes, smiling/non-smiling) and facial details (glasses/no-glasses). Faces are already aligned and scaled, which generally don't require any preprocessing. These face images are directly used for recognition. 100 face images of 20 people are used for our experimental study (Fig. 3). Only one face of each person is used for DCTface calculation. Other 80 faces are used for testing purpose.



(a)

(b)

Fig. 3. (a) ORL face database (b) DCTFaces obtained from 20 faces.

Faces in CMU database are not aligned. So, images in CMU face database are preprocessed before the recognition. For this purpose, face detection and scaling processes are applied. Fig. 1 shows some CMU face images. Frontal face images for each of 20 people are used for training (1_1, 2_1 in Fig. 1 (b)). One right (1_2), one upper-frontal (1_3) and one left (1_4) face are used as test faces. 20 training images are used for generation of 20 DCTfaces. Each test feature vector is compared to feature vectors of 20 training faces by using 3 distance metrics given in Eq. (11), Eq. (12), and Eq. (13).

Face images with similar information can be represented with less DCT coefficients. 3D-DCT exploits not only inter-pixel correlation, but also inter-image correlation [15]. For this reason in 3D-DCT compressed domain, high frequency coefficients with low information can be discarded. We used only low frequency coefficients with small indices. As shown in Fig.2, before 2D IDCT calculation some DCT coefficients are selected for subsequent operations. DCT coefficients with summation of indices greater than a constant number N are taken as zero (pruning). Only N(N+1)/2 coefficients are inversely transformed for DCTfaces.

We investigate the relation between the number of selected coefficients and performance of the recognition in Table I. As a result, selecting more than 465 of DCT coefficients has no effect on the recognition performance (Fig. 4).

TABLE I
RECOGNITION RESULTS FOR SELECTED COEFFICIENTS

N	N(N+1)/2	Recognized Face number	Recognition Ratio (%)
10	55	32	80
15	120	33	82,5
20	210	34	85
25	325	34	85
30	465	35	87,5
50	1275	35	87,5
100	5050	35	87,5
200	20100	35	87,5

Fig. 4. Authorized face image, only 465 DCT coefficients are used for reconstruction.

After getting feature vectors by means of proposed method, distances of the test images to each authorized driver feature vector are calculated according to 3 different norm types. As seen from Table II and Table III, proposed method has recognition ratios nearly equal to Eigenface method for CMU_PIE and ORL face database.

TABLE II
RECOGNITION RATIOS (%) FOR DCTFACE AND EIGENFACE
(CMU PIE FACE DATABASE)

People (4 face for each)	DCTface			Eigenface		
	Norm1	Norm2	Norm3	Norm1	Norm2	Norm3
10	82,5	90	87,5	80	90	92,5
20	68,75	70	75	62,5	82,5	76,25

TABLE III
RECOGNITION RATIOS (%) FOR DCTFACE AND EIGENFACE
(ORL FACE DATABASE)

People (5 face for each)	DCTFace			Eigenface		
	Norm1	Norm2	Norm3	Norm1	Norm2	Norm3
10	90	90	94	88	92	92
20	79	81	83	80	86	87

As mentioned before, difference between the Eigenface and DCTFace method is at the projection face calculation. All other processes are the same. So we give only that part of computational complexity calculations. Complexity for 3D DCT is $O(N_1 N_2 M \log(N_1 N_2 M))$ where N_1 and N_2 are the dimensions of the 2D images, M is the number of the images, so we can write complexity in terms of N and M as $O(NM \log(NM))$.

In PCA, images are converted into 1D vectors, so we used N for $N_1 N_2$. Determining covariance matrix of very large vectors is an intractable task. So a computationally feasible method is used for eigenvalue and eigenvector calculation [5]. PCA complexity is computed at 4 different stage [20],

1-) Finding the mean vector: $O(NM)$

2-) Calculation of covariance matrix: $O(NM^2)$

3-) Performing Eigen analysis on reduced matrix: $O(M^3)$

4-) Calculation of M eigenvectors by using reduced matrix: $O(NM^2)$.

Total complexity is then $O(NM(1+2M)+M^3)$, and that can be expressed as $O(NM^2+M^3)$ asymptotically.

As seen from this complexity analysis, DCTFace method is computationally more efficient than the Eigenfaces method.

V. Conclusion

Both DCTfaces and Eigenfaces have similar ghost like images. The only difference is the calculation method of these faces. Here 3D-DCT is used, which has input independency and fast algorithms for calculation unlike Eigenfaces. Due to the inter-image correlation reduction capability of the 3D-DCT, DCTfaces can be represented by a small number of terms. Recognition performance of the proposed method is very close to that of Eigenface method, whereas the DCTface method has a lower computational complexity. Experimental results show that DCTfaces may be easily used instead of Eigenfaces.

References

[1] K. Takemura, J. Ido, Y. Matsumoto, T. Ogasawara, "Drive monitoring system based on non-contact measurement system of driver's focus of visual attention" in *Proc. 2003 IEEE Intelligent Vehicles Symposium*, 2003, pp. 581 – 586.

[2] K.S. Huang, M.M. Trivedi, T. Gandhi, "Driver's view and vehicle surround estimation using omni directional video stream", in *Proc. 2003 Intelligent Vehicles Symposium*, 2003, pp. 444 – 449.

[3] K.H. Rho, K.H. Park, M.H. Han, "Automatic mirror adjustment system using a driver's pupils", in *Proc. 2002 Intelligent Vehicles Symposium*, 2002, vol.1, pp. 251 – 258.

[4] K.W. Bowyer, K. Chang, P. Flynn, "A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition", *Computer Vision and Image Understanding*, vol. 101, no. 1, pp 1–15, 2006.

[5] M.A. Turk, A.P. Pentland, "Face recognition using Eigenfaces" in *Proc. CVPR 1991*, pp. 586 – 591.

[6] V.V. Kohir, U.B. Desai, "Face recognition using a DCT-HMM approach" in *Proc. IEEE WACV* ,1998, pp.226 – 231.

[7] W. Chen, M.J. Er, S. Wu, "Illumination compensation and normalization for robust face recognition using discrete cosine transform in logarithm domain", *IEEE Transactions on Systems, Man and Cybernetics*, Part B, vol. 36, issue 2, pp. 458 – 466, 2006.

[8] G. Gunlu, "Improving DSP performance for artificial neural networks based face recognition" in *Proceedings of the 2005 IEEE Signal Processing and Communications Applications Conference*, pp. 551 – 556, May 2005.

[9] E. Erzin, Y. Yemez, A.M. Tekalp, A. Ercil, H. Erdogan, H. Abut, "Multimodal person recognition for human-vehicle interaction", *Multimedia IEEE*, vol. 13, issue 2, pp. 18 – 31, 2006.

[10] C. Podilchuk, X. Zang, "Face recognition using DCT-based feature vectors" in *Conference Proceedings IEEE ICASSP-96*, vol. 4, 1996, pp. 2144 – 2147.

[11] Y. Chen, S. Jiang, A. Abraham, "Face Recognition Using DCT and Hybrid Flexible Neural Tree" in *Neural Networks and Brain Conf.*, vol. 3, 2005, pp. 1459 - 1463.

[12] R. J. Clark, "Relation between the Karhunen-Loeve and cosine transform" in *Proc. IEEE*, vol. 128, 1981, pp. 359–360.

[13] M. Uenohara, T. Kanade, "Optimal approximation of uniformly rotated images: relationship between Karhunen-Loeve expansion and discrete cosine transform", *IEEE Transactions on Image Processing*, vol. 7, issue 1, pp.116 – 119, 1998.

[14] N. Božinović and J. Konrad, "Motion analysis in 3D DCT domain and its application to video coding", *Sig. Process.: Image Communication*, Elsevier Science, vol. 20, iss. 6, pp. 510 – 528, July 2005.

[15] N.P. Sgouros, S.S. Athineos, P.E. Mardaki, P.E.; A.P. Sarantidou, M.S. Sangriotis, P.G. Papageorgas, N.G. Theofanous, "Use of an adaptive 3D-DCT scheme for coding multiview stereo images" in *Proc. 2005 IEEE Signal Processing and Information Technology*, pp. 180 – 185.

[16] S. Boussakta, H.O. Alshibami, "Fast algorithm for the 3-D DCT-II", *IEEE Trans. on Signal Processing*, vol. 52, issue 4, pp. 992 – 1001, 2004.

[17] M.-H. Yang; D.J. Kriegman, N. Ahuja, "Detecting faces in images: a survey", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, issue 1, pp. 34 – 58, 2002.

[18] P. Viola, M. Jones, "Rapid object detection using a boosted cascade of simple features" in *IEEE Computer Society Conf. CVPR 2001*, vol. 1, 2001, pp. I-511 – I-518.

[19] R. Lienhart, J. Maydt, "An extended set of Haar-like features for rapid object detection" in *Proc. 2002 Conf. Image Processing*, vol. 1, 2002, pp. I-900 – I-903.

[20] S. Kaewpijit, J. Le Moigne, and T. El-Ghazawi, "Automatic reduction of hyperspectral imagery using wavelet spectral analysis", *IEEE Transactions On Geoscience And Remote Sensing*, vol. 41, no. 4, April 2003.