Adaptive Multi-model Biometric Fusion for Digital Watermarking

P. Jidesh[1] and Santhosh George[2]

[1] Department of Mathematical and Computational Sciences, National Institute of Technology, Karnataka, 575025, India

[2] Department of Mathematical and Computational Sciences, National Institute of Technology, Karnataka, 575025, India

Abstract

In this work we propose to embed a fused biometric feature vector as a watermark into the sample image in spatial domain based on the fusion parameter (for embedding the watermark) which is chosen adaptively by using the structural similarity measure. Further the watermark is extracted from the subject image and verified with a considerably good accuracy. The results are demonstrated with substantial qualitative and quantitative measures to endorse on the effectiveness and efficiency of the proposed method.

Keywords: *Steganography, Structural similarity measure (SSIM), Adaptive Embedding Function, Spatial Domain Watermarking.*

1. Introduction

There has been a rapid growth in the area of steganography and watermarking during past couple of decades [3], [4], [5]. Various kinds of data are used as a stego-data or as a watermark for embedding into the image, it varies from mathematically formulated Gaussian random data to more general image data like symbols, logos etc. The possibility of using biometric features like fingerprint, face-image, palm prints etc. as watermarks were explored during recent years. There has been a considerable magnitude of woks that have proceeded in this direction see [6] for details. Spatial as well as frequency characteristics were explored for possible embedding of the stego-data see [1], [3]. Further the data was embedded in visible as well as in invisible forms [4], [8].

Using biometric information as a watermark was relatively new and captured the recent attention due to its uniqueness and credibility. The biometric data cannot be duplicated like other common watermarking data. Even the multi-model (fusing more than one kind of biological features) biometric data were used as a watermark in some recent works [6]. Multi model biometric data is considerably a robust stego-data due to its high sustainability towards normal attacks. The multi-model biometric data provide a high magnitude of protection for the data by reducing the risk of attacks. In this work we propose to use a multi-model biometric data (fingerprint and face) fused together to form a feature vector and embed in the spatial domain of the input image invisibly, based on the embedding parameters chosen adaptively.

This paper is organized into five sections. In section 2 we explain about the feature-vector (watermark) generation and embedding process. Section 3 explains about the watermark extraction and matching procedure. Section 4 will give an outline on the experimental woks carried-out to test the methods and the results. Section 5 concludes the work.

2. Watermark Generation Process

Generally a watermarking process can be mathematically modeled as:

$$\hat{f}(x,y) = f(x,y) \otimes \alpha\, \eta(x,y) \qquad (1)$$

where $\otimes$ denotes the kind of operation involved in watermarking, either additive or multiplicative operation is commonly employed due to its invertible nature. Here $\hat{f} : \mathrm{R}^2 \to \mathrm{R}^2$ denotes the watermarked image, $\eta(x,y)$ denotes the watermark, $\alpha \in [0,1]$ denotes the strength of the embedded watermark and $f : \mathrm{R}^2 \to \mathrm{R}^2$ is the original image with $x, y \in \rho$ where ρ denotes the pixels in the watermarking domain. If we have the information regarding the watermark η, the operation ($\otimes$) used for embedding and the strength parameter α, then we can retrieve the watermark from the affected image with a considerable amount of accuracy. But in many practical

scenarios these information may not be available, if we are to retrieve the data without a prior knowledge of these information then it is termed as blind-steganalysis. Blind-steganalysis is widely used in forensic applications.

In this work we assume that the watermark embedded, the strength of the watermark and the procedure followed (operator used) are known in advance to the intended receiver. Embedding the whole fingerprint or face image as a watermark into the desired data/image may results in a visible difference in the watermarked image, which is not desired. So we generate a feature vector which uniquely represents the input watermark and embed the same in the input image without making any noticeable difference in the watermarked image. Since the biometric features are used as a watermark there is an inherent binding from the source side. In other words the sender cannot deny the authenticity of the message sent.

In this paper we use a combination of Linear Discriminant Analysis (LDA) and Principle Component Analysis (PCA) to generate the feature vector as in [7]. Then we propose to evaluate the parameters α , β of the blending function in Eq. (11) and γ in Eq. (15) based on the *Mean Structural SIMilarity Index*(MSSIM) [10].

2.1 Feature Vector Generation

We generate the feature vector from face image and fingerprint image using Linear Discriminant Analysis and Principle Component Analysis [2], [7], [9] as the fundamental procedure. We assume that there are K fingerprint and face images in the test set and all are of size $N \times N$ pixels. We apply a LDA on the input face and fingerprint images to generate the feature vector; here PCA is applied as a pre-procedure, in order to ensure that the scatter matrix in Eq. (7) is non-singular. In addition to this PCA will considerably reduce the dimensionality of the matrix, which in turn will reduce the complexity of the calculations. Since in PCA we derive a set of eigenvectors corresponding to distinct non-zero eigenvalues, the matrix formed from these vectors will never be singular. Further these vectors form a basis because all the eigenvectors corresponding to distinct non-zero eigenvalues will be linearly independent and can span the space. This property of PCA can be exploited in obtaining an optimal non-singular scatter matrix. PCA can be summarized in following steps:

1. Let $f(x, y)$ denote the input matrix of size $N \times N$, and further assume Γ_i be a column vector corresponding to the i^{th} column of the input matrix with size $N \times 1$.
2. Compute the average Image vector ψ :

$$\psi = \frac{1}{N}\sum_{i=1}^{N}\Gamma_i \qquad (2)$$

3. Subtract the average vector from each of the N input column vectors:

$$\phi_i = \Gamma_i - \psi \quad i = 1, 2...N \qquad (3)$$

4. Compute the covariance matrix *C*:

$$C = \frac{1}{N}\sum_{n=1}^{N}\phi_n\phi_n^T = AA^T \quad (N \times N \ matrix) \qquad (4)$$

where $A = [\phi_1\ \phi_2\ ...\ \phi_N]$ is a $N \times N$ matrix.

5. Compute the eigenvectors u_i of AA^T , There can be only N eigenvectors. The eigenvectors corresponding to distinct non-zero eigenvalues will be linearly independent. Further note that the covariance matrix (used for calculating the eigenvectors) is symmetric, hence the eigenvalues will be real and positive. If we take only the eigenvectors corresponding to the distinct non-zero eigenvalues then, they can span an eigen-space (a sub-space spanned by eigenvectors) and further they can form a basis for the corresponding eigen-space (because these vectors are linearly independent).
6. We take some K dominant eigenvectors from *N* eigenvectors such that $K \square N$ and based on the non-zero distinct eigenvalues of the matrix AA^T . This eigenvectors span an eigen-space with a dimensionality much less than that of the input matrix. The decrease in dimensionality will affect the accuracy of the detection process, so it is advisable to set the dimensionality based on the desired accuracy.
7. Now project the input vectors (corresponding to the input image) on to this eigen-space spanned by the eigenvectors (u_i) 's. Let B represents a matrix formed by set of eigen-vectors of size $N \times K$ then the output matrix will be of size $K \times N$.

$$O = \sum\sum B^T(x, y)\, f(x, y) \qquad (5)$$

Since we have K eigenvectors of size $N \times 1$ and the input image is of size $N \times N$ the dimension of the output matrix O will be $K \times N$.

8. Since all the eigenvectors corresponding to distinct non-zero eigenvalues are ortho-normal (the eigenvectors are normalized) the inverse of such matrix is just transpose. So for the eigen-matrix (formed by a set of eigenvectors) the inverse will be just its transpose (since the eigenvalues are real and positive the matrix will

be positive definite). Therefore the data can be reconstructed using the following formula:

$$\hat{f} = \sum\sum B(x,y)O(x,y) \quad (6)$$

It is obvious from the above steps that when it comes to pattern classification PCA may not perform well, the whole training set is assumed to be from the same class in PCA. In this aspect LDA outperforms PCA in terms of pattern classification. Since face images and fingerprints can be classified based on the certain features, it will be highly beneficial to use LDA instead of PCA. When PCA is applied on the input face images the resulting images are called *Eigen-faces*. The steps in LDA can be summarized as follows:

1. Let L be a set of training images $\{Z_i\}_{i=1}^{L}$ each of dimension $n \times m$, each of which represents a vector of dimension $N \times 1$. In other words $z_i \in \mathrm{R}^N$ belongs to one of K classes of $\{Z_i\}_{i=1}^{K}$, where R^N is a real space of dimension N.
2. The main objective is to find a transformation ϕ based on optimization of certain class separability criteria, in order to yield a transform of the form $x_i = \phi(z_i)$ where $x_i \in \mathrm{R}^M$ where $M \square N$. The representation x_i is such that the separability criterion is optimal.
3. Now we define inter and intra class scatter matrices S_{inter} and S_{intra} respectively.
4. The intra-class scatter matrix is defined as :

$$S_{intra} = \sum_j (p_j \times (cov_j) \quad (7)$$

where p_j is the probability of j^{th} class and cov_j is the covariance of the j^{th} class, as defined in Eq. (9). Similarly inter-class scatter matrix is defined as :

$$S_{inter} = \sum_j (\mu_j - \mu_K) \times (\mu_j - \mu_K)^T \quad (8)$$

where μ_k denotes the mean of all the classes and μ_j is the mean of the j^{th} class. The cov is defined as:

$$cov = \sum_j (y_j - \mu_j) \times (y_j - \mu_j)^T \quad (9)$$

where y_j is the j^{th} input vector.

5. LDA uses Fisherface method in [2], [11] to find a set of basis vectors, denoted by x_i that maximizes the ratio between S_{intra}, S_{inter} :

$$\xi = \underset{\xi}{argmax} \frac{|\phi^T S_{inter}|}{|\phi^T S_{intra}|} \quad (10)$$

The basis vector ξ corresponds to first K eigenvectors of $(S_{intra}^{-1} S_{inter})$, hence K dimensional feature vector is obtained by projecting the input images into the subspace spanned by these K eigenvectors. However, we cannot guarantee that the scatter matrix S_{intra} to be always non-singular. Hence it is quite advisable to perform a PCA on this matrix, before proceeding with the LDA. By applying PCA the vectors will become linearly independent and the matrix will be positive definite [9], hence zero will never be an eigenvalue and the matrix will be non-singular. Further the vectors will be ortho-normal, hence the matrix is unitary and so the inverse of the matrix will be its transpose. Further one can notice that when LDA is applied of a set of test face images, will result in a set of *Fisher-faces.* The *Eigen-faces* and *Fisher-faces* corresponding to two test images are shown in Fig 5 in Section 4.

During the watermark embedding stage the fingerprint and the face image of the concerned authority (which is a source in this case) will be converted into feature vectors by the LDA procedure explained above. Hence two feature vectors will be generated by this process, one corresponding to the face and another corresponding to fingerprint. Let F be the feature vector generated for fingerprint and F' be the vector generated for face image then the multi-model biometric fusion can be expressed as a blending function of the form:

$$G = F\alpha + F'\beta \quad (11)$$

Where α and β are the parameters to the blending function which are determined empirically. The choice of α and β are done in an adaptive way making use of the quality metric MSSIM defined in [10]. This metric will provide necessary information to choose the blending function parameters α and β. Since we have to solve the equation Eq. (11) with two unknowns F and F' at the receiving end, we will require one more equation. We consider this equation as the sum of the two feature vectors:

$$G' = F + F' \quad (12)$$

From the two equations Eq. (11) and Eq. (12) we can uniquely derive the values F and F', by solving them simultaneously. The watermark embedding procedure is shown using a flowchart given in Fig. 1.

2.2 Structural SIMilarity Index (SSIM)

The motivation to use this approach is to find a more direct way to compare the structures of the reference and the distorted signals [10] . This new framework for the design of image quality measures was proposed, based on the assumption that the human visual system is highly adapted to extract structural information from the viewing field, the SSIM is formulated as:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C1)\times(2\sigma_{xy} + C2)}{(\mu_x^2\mu_y^2 + C1)} \qquad (13)$$

where x and y denotes the content of local windows in original and watermarked image respectively. The measure is applied for non-overlapping windows in both the images. In this paper we measure mean-SSIM (MSSIM)which is an index to evaluate the overall image quality. It is defined as:

$$MSSIM(X,Y) = \frac{1}{M}\sum_{j=1}^{M} SSIM(x_j, y_j) \qquad (14)$$

where X and Y are the original and watermarked image respectively; x_j and y_j denotes the content of the j^{th} local window and *M* is the number of local windows in the image; μ_x and μ_y are the mean of the two windows for which the measure is applied, C_1 and C_2 are constants.

2.3 Selection of Pixels to Embed the Watermark

Selection of pixels plays a crucial role in watermarking there are many strategies followed for selecting the pixels see [3], [4] for details.

In this work we choose pixels in such a way that adding the watermark into those pixels will not effectively make any noticeable difference in the input image (here we adopt an invisible watermarking scheme). This noticeable difference is quantified by the MSSIM. We choose the parameters α , β and γ based on the MSSIM, such that the MSSIM is within the desired limit. The flowchart given in Fig. 2 explains about the selection of pixels for embedding the watermark. The pixels are selected in such a way that when the watermark is added into these pixels the resulting image will not appear distorted or the embedded data will not be noticeable to the naked eyes. The watermark is embedded with the following embedding equation:

$$I(x,y)'_{x,y\in\rho} = I(x,y)_{x,y\in\rho}\gamma + (1-\gamma)G \qquad (15)$$

where *G* is an in Eq. (11). $I(x,y)'_{x,y\in\rho}$ denotes the set of pixels affected by the watermark and $I(x,y)_{x,y\in\rho}$ denotes the set of pixels selected for watermarking. Now the watermark is embedded in to the pixels that belong to the set ρ. The set ρ is selected based on the selection procedure explained in Fig. 2 The pixels with minimum gradient values (∇I) are selected as candidates for embedding the watermark because these pixels belong to the constant intensity areas and will not make any noticeable difference even after embedding the watermark. Since the selection of pixels is done globally the watermarking procedure is a global one and the embedded watermark remains un-noticeable to the naked eyes so it is invisible in nature. Further γ is a parameter to decide the strength of watermark to be embedded. If γ is a high value, then the strength of watermark embedded is less and vice-versa.

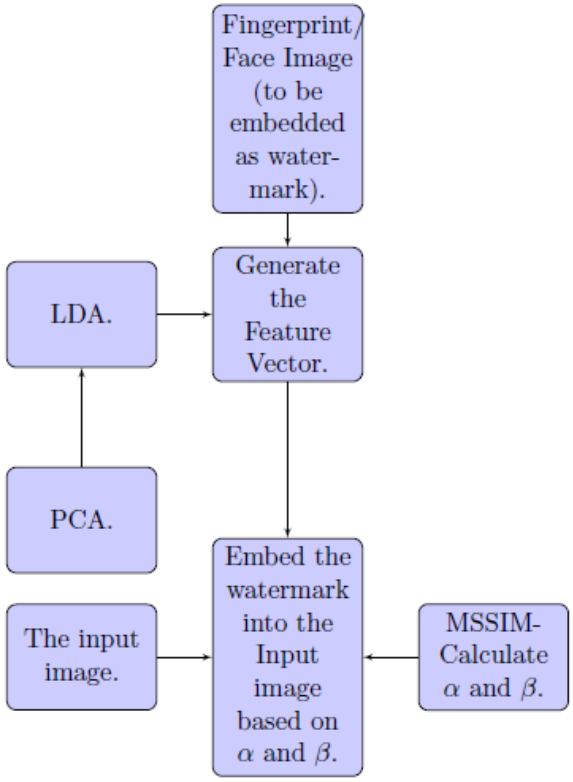

Fig. 1 The watermark embedding process in the Proposed Method.

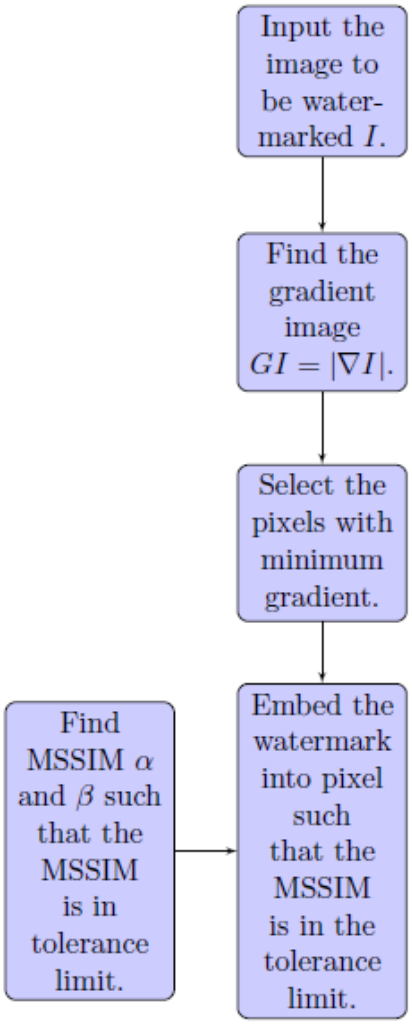

Fig. 2 Selection of pixels to embed the watermark.

3. Watermark Extraction and Comparison

Watermark extraction process is just a reverse of watermark embedding process. Here we assume that the receiver is completely aware of the watermark added into the image as well as the original image. Further the parameters α , β and γ are known to the receiver. From these information one can extract the watermark using the procedure explained in the flowchart in Fig. 3. Since the receiver has the information regarding the original image $I(x,y)$, the parameters viz. α, β *and* γ used for embedding the watermark and the watermarked image $I(x,y)'$, the primary task in extraction of the watermark will be to find out the pixels affected by the watermark. From Eq. (15) we have the set (watermarking domain) ρ, which denotes the set of pixels affected by the watermark, these pixels can be traced with the help of the gradient image. We have inserted the watermark into the pixels with minimum gradient values, so the watermarked pixels fall in the constant intensity areas. Hence we can find the domain ρ (set of pixels affected by the watermark) from this information. Once the set ρ is formed then the weighted embedded watermark G can be extracted by the inverse embedding equation:

$$G = \frac{I(x,y)'_{x,y\in\rho} - I(x,y)_{x,y\in\rho}\gamma}{(1-\gamma)} \qquad (16)$$

From Eq. (11), Eq. (12) and Eq. (16) , we can find the feature vectors $F(x,y)$ and $F'(x,y)$ corresponding to fingerprint and face image, by solving these equations.

Once the feature vector is extracted then the next step is to search database containing the feature vectors for a possible match, and extract the sender information. The matching process is just a match factor calculation and its comparison with a predefined *Threshold*. Let *F* denotes the feature vector extracted from the watermarked image and $\tilde{F}$ be one the feature vector stored in the database. The match factor (MF) is defined as:

$$MF = \| F - \tilde{F} \| \qquad (17)$$

where $\| \ \|$ denotes the Euclidean norm of the vector. If $MF \le Threshold$ then a possible match is found. Based on the *MF* the feature vector with least *MF* value is considered to be more similar. Hence the feature vector with minimum *MF* value is selected as the one that corresponds to the sender. If none of the feature vectors are falling under this criterion then a possible attack or an un-authorized watermark is alarmed.

4. Experimental Results

We used the fingerprint images from FVC-2000 (DB1, DB2, DB3, DB4) and Face images are taken from "Yale Face Database". We associated a fingerprint to a face image and carried out the testing. There are 11 face images of dimension (300×300) per subject in the database and five fingerprint images per subject in FVC-2000. We have chosen five face images and fingerprint images of size (300×300) per subject for our experiments. We have rescaled this image to (150×150) for making the feature vector small so that the embedded watermark remains

invisible. We have embedded the watermark in the test image "Lena" with dimension 512×512.

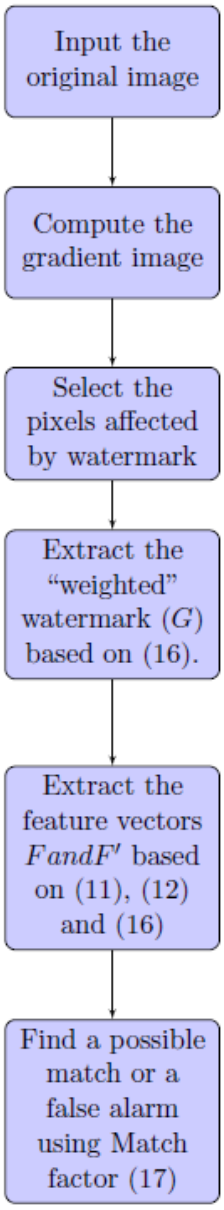

Fig. 3 The watermark extraction process in the Proposed Method.

There are 15 different classes of face and fingerprint images corresponding to different subjects. We have chosen the number of basis vectors as 15, making the feature vectors of size 15×150 for both face and fingerprint images. Then we apply the blending function with α, β *and* γ with values 0.36 , 0.47 and 0.42 respectively obtained by applying the MSSIM. The values of α, β *and* γ are the optimal values selected based on the MSSIM. If the values of α, β *and* γ are increased then the watermark will be visually distinguishable and if the values are decremented then the extracted watermark will not be prominently detected due to the week contribution of the watermark. Fig. 4 shows the test figure "Lena" before and after embedding the watermark. Figure 4(D), Fig. 4(E) and Fig. 4(F) shows the figure after embedding the watermark. The Fig. 4(E) and Fig. 4(F) show the results of applying the watermark with the parameter values α, β *and* γ other than the ones calculated using the MSSIM. It is clear from the images that when the parameter values are different from the calculated values (based on MSSIM) the watermarked image is visually distinguishable from the original one. The watermark is constructed from Fig. 4(A) and Fig. 4(B) by using multi-model biometric fusion with parameter α *and* β whose values are calculated based on MSSIM. Instead of replacing the pixels in the original image with the watermark components we use a regularization approach, in which a parameter γ determines the strength of contribution of the watermark component and the original pixel values. If $\gamma = 0$ then there will be contribution only from the watermark components, the original image pixel values will not have any role in watermarked image, whereas if $\gamma = 1$, then no watermark will be embedded into the input image. So the range of parameter γ is [0,1]. This value is also chosen based on the MSSIM. Table 1 shows the accuracy of the proposed method in terms of correctly identifying the embedded watermark at different *Threshold* values. It is quite evident from the table that the method has better accuracy when the *Threshold* value is 10. Table 2 shows the performance of the proposed method for different values of fusion parameters α, β and γ. It is clear from these values that the performance of the method is optimal for the values $\alpha = 0.36, \beta = 0.47$ *and* $\gamma = 0.42$.

Table 1: Performance of the proposed method for different Matching *Threshold* values.

Threshold	***False Acceptance Rate***	***Genuine Rejection Rate***	***Overall Performance (%)***
10	1.2	1.5	97.6
15	1.6	1.8	96.2
20	2.3	3.1	94.7

Table 2: Performance of the proposed method for different α, β, γ for *Threshold=10.*

α, β, γ	***MSSIM***	***Overall Performance (%)***
0.10, 0.30, 0.42	0.70	92.8
0.36, 0.47, 0.42	**0.80**	**97.6**
0.50, 0.61, 0.42	0.48	97.0
0.36, 0.47, 0.20	0.82	94.5
0.36, 0.47, 0.60	0.62	95.5

The eigen-faces and fisher-faces obtained after applying PCA and LDA respectively on the input face images are shown in Fig. 5.

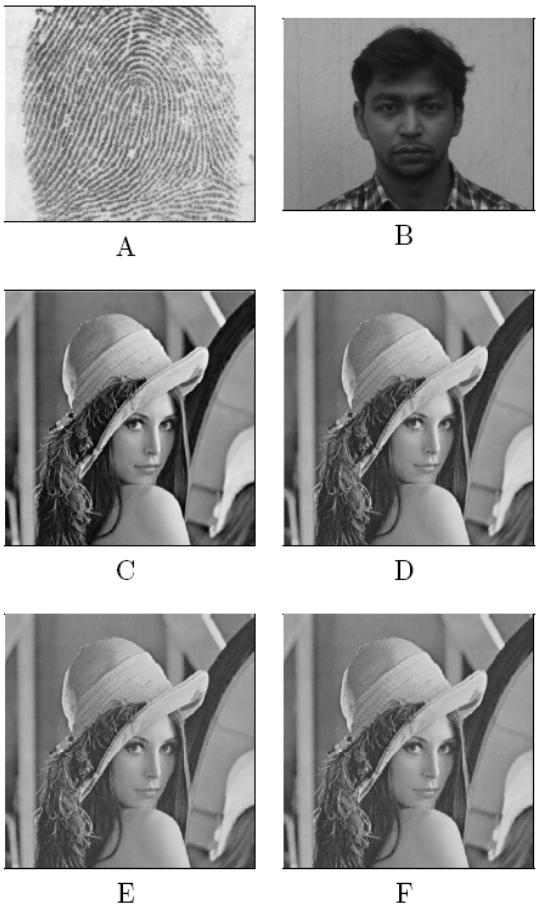

Fig. 4 Image "Lena``:(A) Fingerprint Image (Watermark) (B) Face Image (Watermark) (C) Image to be watermarked (D) Image After Watermarking (with parameters α=0.36, β=0.47, □ =0.42) (E) After applying watermark with parameters (α=0.36, β=0.47, □ =0.2) (F) After applying watermark with parameters (α =0.50, β=0.61, □ =0.2).

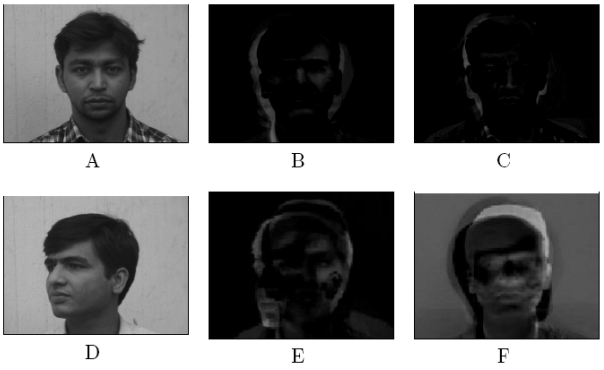

Fig. 5 (A) & (D) The input face images. (B) & (E) The *Eigen-faces* corresponding to input image (A) & (D) respectively. (C) & (F) The *Fisher-faces* corresponding to image (A) & (D) respectively.

5. Conclusion

In this paper we have proposed a method which combines two biometric features (fingerprint, face) to form a single feature vector and embedded into the image based on adaptive parameter selection. The fusion parameters α, β *and* γ are selected based on the structural similarity measure (MSSIM) which is close to human perception. The performance of the proposed method is quite evident from the results provided.

Acknowledgments

The Authors wish to thank NITK for the financial support under seed money grant.

References

[1] A.S. Madani, A.I. Hashad and A.E.M.A. Wahdan, "A robust steganography technique using discrete cosine transform insertion", IEEE/ITI 3rd International Conference on Information and communications Technology, 2005, pp. 255-264.

[2] Baback Moghaddam, Alex Pentland and Thad Starner, "View-based and modular eigenspaces for face recognition", IEEE Conf. on Computer Vision and Pattern Recognition, 1994, pp. 245-250.

[3] R. Anderson and F. Petitcolas, "On the limits of stehanography", IEEE Journal of Selected Areas in Communications, vol. 4, no. 16, 1998, pp. 474-481, 1998.

[4] Christian Cachin, "An information-theoretic model for Steganography", SANS Intrusion Detection and Response, 1999, pp. 295-305.

[5] Niel F Jhonson, "An introduction to watermark recovery from images", In Lecture Notes in Computer Science, 1998, pp. 306-318.

[6] Niel F Jhonson, "Biometric image authentication using watermarking", SICE-ICASE: International Joint Conference, 2006, pp. 3950-3953.

[7] K.N. Juwei Lu Plataniotis and VenetsanopoulosA.N. Biometric image authenticationusing watermarking. IEEE Transactions on Neural Networks, vo. 14, no. 1, 2003, pp. 195-200.
[8] O.C. Chi-Wang Ho Shu-Kei Yip, Au andHoi-Ming Wong. Lossless visible watermarking. In IEEE International Conference on Multimedia and Expo, 2006, pp. 853-856.
[9] M.A. Turk and A.P. Pentland, "Face recognition using eigenfaces", IEEE Conf. on Computer Vision and Pattern Recognition, 1991, pp. 586-591.
[10] Zhou Wang and Alan C Bovik, "Image quality assessment: From error visibility to structural similarity", IEEE Transactions on Image Processing, vol. 13, no. 4, 2004, pp. 1-14.
[11] Ming H. Yang, "Kernel eigenfaces vs. kernel fisher faces: Face recognition using kernel methods", Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition, 2002, pp. 215-220.

P. Jidesh received his B.Sc degree from University of Calicut in 1998, MCA degree from NIT Calicut 2001 and M.Tech Degree in Computer Science from University of Kerala in 2005. He worked as an Asst. Systems Engineer in TCS, Chennai and as a Subject Expert in AMDOCS DVCI Pune, India. Since 2009 January he is working as an Asst. Professor in the Department of Mathematical and Computational Sciences, National Institute of Technology, Karnataka, India. He has guided many M.Tech dissertation works. His research interests include PDE's/Variational methods in Image processing, Watermarking and Inverse problems in Imaging.

Santhosh George received his M.Sc degree in Mathematics from University of Calicut and PhD degree in Mathematics from University of Goa. Since 2008 he is working as an Associate Professor in the Department of Mathematical and Computational Sciences, National Institute of Technology, Karnataka. He has guided many M.Tech thesis works and two students have submitted their PhD thesis under his guidance. He has many International journal and conference papers to his credit. His research interests include Inverse problems in Science and Engineering and Functional Analysis.