http://jecei.srttu.edu

SRTTU

Journal of Electrical and Computer Engineering Innovations

JECEI, Vol. 5, No. 2, 2017

Regular Paper

Combining and Steganography of 3-D Face Textures

Mohsen Moradi[1,*] and Mohammad-Reza Sadeghi[1]

[1] Department of Mathematics and Computer Science, Amirkabir University of Technology – Tehran Polytechnic, Tehran, Iran.

[*]Corresponding Author's Information: m.moradi8111@aut.ac.ir

ARTICLE INFO

ARTICLE HISTORY:
Received 14 February 2017
Revised 12 March 2017
Accepted 20 April 2017

KEYWORDS:
Steganography
Shape
Texture
Face image
Combining images

ABSTRACT

One of the serious issues in communication between people is hiding information from the others, and the best way for this, is to deceive them. Since nowadays face images are mostly used in three dimensional format, in this paper we are going to steganography 3-D face images and detecting which by curious people will be impossible. As in detecting face only, its texture is important, we separate texture from shape matrices, for eliminating half of the extra information. Steganography is done only for face texture, and for reconstructing a 3-D face, we can use any other shape. Moreover, we will indicate that, by using two textures, how two 3-D faces can be combined. For a complete description of the process, first, 2-D faces are used as an input for building 3-D faces, and then 3-D face and texture matrices are extracted separately from the constructed 3-D face. Finally, 3-D textures are hidden within the other images.

1. Introduction

In cryptography, encrypted message is at the center of people's attention, and its security is based on the difficulties on accessing the message key. In steganography, other's unawareness is used for hiding the message to send it in the safest way. For this reason, first, the essential information of one image which we want to send to the others will be embedded in another host image in a way that the others cannot understand outward discrepancy of the initial host image and the embedded host image, therefore legal receiver can extract and reconstruct the initial embedded image. The most important research works conducted in this area are in [1], [2], [3], [4], and [5] .

Embedding capacity and visual quality are the two essential parameters in stego or cover images [6]. Embedding capacity refers to the maximum amount of secretive message which can be embedded in the host image, and visual quality is the embedding message in the host image in a way that human eye cannot notice any difference between the new form and the original one. One criterion which is usually used for evaluating visual quality is the peak of signal to noise ratio (PSNR) between the stego image and the original host image and is expressed in dB unit. The bigger the PSNR, the higher is the visual quality of the stego image. In other words, it is more difficult for an human eye to detect stego image than to do so for the host image [7].

In this paper, to construct a 3-D image of the intended face for steganography within other images, first a colored 2-D face image will be processed using Basel database as in [8] and is then fitted to a 3-D model. 3-D face and texture matrices are extracted separately from the constructed 3-D face. Now, if this 3-D texture is used for any other 3-D shapes, it is possible to recognize the person as well. For this reason, in steganography, the shape matrix is not so important and hence we only use texture matrix. A wavelet-based watermarking algorithm is used to enhance the secrecy. By using singular value decomposition (SVD) and discrete wavelet transform (DWT), the information of a 3-D texture of an image will be hidden in a label image. This label image can be any 2-D image e.g., a fingerprint, a shape, or a texture of the other 3-D images. First, label image is converted into frequency domain, and SVD is used on both of the original cover image and 3-D texture. Then, the two obtained singular values will be replaced with each other.

2. Morphable Model

3-D morphable models were first introduced by Blanz and Vetter [2], [9] in late 90s. They were applied successfully to computer images and graphics. A 3-DMM includes separate shape and texture models which by themselves can build each person's shape and texture's changes, respectively. A 3-DMM is used for one group of 3-D scanned images.

Constructing 3-D models is very hard and time consuming and Paysan and his colleagues provided their 3-DMM for public usage. Basel model uses 3-D models of 100 men and 100 women whose ages ranged from 8 to 64 [8].

An iterative multiresolution dense 3-D registration which has been introduced by Rodriguez [10] is studied in this section. Suppose that the ith vertex of the registered image is in (x_i, y_i, z_i) point and its RGB color is (R_i, G_i, B_i) . It will be assumed that for any face, N 3-D points are registered. Therefore, one registered face in face and texture language can be shown as:

$$\boldsymbol{S}' = \begin{pmatrix} x_1 . \dots . x_N \\ y_1 . \dots . y_N \\ z_1 . \dots . z_N \end{pmatrix} \tag{1}$$

$$\boldsymbol{T}' = \begin{pmatrix} R_1 . \dots . R_N \\ G_1 . \dots . G_N \\ B_1 . \dots . B_N \end{pmatrix} \tag{2}$$

Also, these points can be shown in one row or column vector of length 3N. For example, the vertical vectors of shape and texture are:

$$\boldsymbol{S}' = (x_1 . \dots . x_N . y_1 . \dots . y_N . z_1 . \dots . z_N)^T, \tag{3}$$

$$\boldsymbol{T}' = (R_1 . \dots . R_N . G_1 . \dots . G_N . B_1 . \dots . B_N)^T \tag{4}$$

which N is the number of registered faces. We suppose that these two kinds of point display are equivalent, and we will use any of these two formats when needed.

Now, for a database which has 200 shapes and texture matrices, a linear combination for shapes and textures is used. These linear combinations can be formulated as follows:

$$\boldsymbol{S} = \sum_{i=1}^{200} \alpha_i . \boldsymbol{S}_i . \qquad \boldsymbol{T} = \sum_{i=1}^{200} \beta_i . \boldsymbol{T}_i . \tag{5}$$

It is almost impossible that these combinations, despite the fact that they include all the possible faces, are similar to a real face. If a convex combination $(\sum \alpha_i = 1 . \sum \beta_i = 1 . \alpha_i . \beta_i \in [0.1])$ of faces is supposed, hence, a face can also be obtained, but again, it is not possible that the points which are far from the convex area can form the actual face points. Therefore, for any vector, a coefficient is needed to be allocated to a probability distribution for the description of a face. This probability is modeled by a Gaussian distribution, in which shapes and textures are decorrelated, with a diagonal matrix. Suppose a Gaussian distribution allows that subspace of a face to be estimated by a smaller set of orthogonal basis vectors which is calculated using the principle component analysis (PCA) of the test samples.

Principle component analysis is a statistical tool which transform shape or texture so that the covariance matrix will be diagonal (it means that data are decorrelated). In this section, using PCA for the shape is studied. Using PCA for the texture is also done in a similar way. PCA is a transform in the vector space which is used mostly for decreasing the dimensions of the data sets. Originally, the principle component analysis was used by Karl Pearson in 1901 [11]. This analysis includes the decomposition of the eigenvalues of the covariance matrix.

The average of the shapes is calculated as:

$$\overline{\boldsymbol{S}} = \frac{1}{200} \sum_{i=1}^{200} \boldsymbol{S}_i . \tag{6}$$

By subtracting each sample shape from the average shape matrix, the vertical vector a_i can be calculated as:

$$a_i = vec(\boldsymbol{S}_i - \overline{\boldsymbol{S}}). \tag{7}$$

This vertical vectors are used as the columns of matrix **A**, and the eigenvalue vectors of covariance **C** are calculated by a singular value decomposition [12] . So, we have:

$$\boldsymbol{A} = (a_1 . a_2 . \dots . a_{200}) = \boldsymbol{UWU}^T \tag{8}$$

and

$$\boldsymbol{C} = \frac{1}{200} \boldsymbol{AA}^T = \frac{1}{200} \boldsymbol{UW}^2 \boldsymbol{U}^T . \tag{9}$$

$Vec(\boldsymbol{S})$ will change matrix $\boldsymbol{S}$ to a column vector by concatenating its columns into a vertical order. 200

columns of orthogonal matrix $\boldsymbol{U}$ are eigenvectors of covariance matrix $\boldsymbol{C}$ and $\sigma_i^2 = \frac{\lambda_i^2}{200}$ is its eigenvalues in which λ_is are diagonal elements of matrix W that are arranged in a decreasing order. The ith column of $\boldsymbol{U}$ is shown by $\boldsymbol{U}_{0.i}$, and the principle component of i will be changed to a $3 \times n$ matrix using $\boldsymbol{S}^i = \boldsymbol{U}_{0.i}^{(3)}$. The notation $a_{m\times 1}^{(n)}$, changes the $m \times 1$ vector a to a $n \times (m/n)$ matrix [13].

Now, instead of describing a new shape or texture as the linear combination of the samples as in (5), they can be expressed by the linear combination of n_s shapes and n_t textures as the principle components:

$$\boldsymbol{S} = \overline{\boldsymbol{S}} + \sum_{i=1}^{n_s} \alpha_i . \boldsymbol{S}^i \tag{10}$$

and

$$\boldsymbol{T} = \overline{\boldsymbol{T}} + \sum_{i=1}^{n_t} \beta_i . \boldsymbol{T}^i \tag{11}$$

Therefore, a combination of an arbitrary number of shapes and textures to construct 3-D morphable faces is used.

The 3-DMMs should be so limited that improbable faces could rarely be sampled.

Experiments on the real data show that by supposing a Gaussian distribution on the face when sufficient information in the extraction is available, the result is satisfying. However, to assume a Gaussian distribution is not always practical, so Patel and Smith presented another assumption [4] . They observed that the lengths of the vectors of the parameters have Chebyshev distributions. In other words, the real faces are on the spherical manifolds. In this case, the average model, with its vector length being zero, is improbable, and the faces should have a certain level of distinction.

3. Extracting 3-D Shapes and Textures From 2-D Face Images

In this section, extracting three dimensional shapes and textures of 2-D colored images is explained and we show that how well we can combine two of them. The flowchart of this procedure is shown in Fig. 1.

Bas [1] used Basel database for fitting 2-D images on to a 3-D model. This fits N 2-D $\boldsymbol{X}_i = \binom{x_i}{y_i}$ points for $i = 1, \dots, N$ to the corresponding 3-D $\boldsymbol{V} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}$ vectors on the model in order to get the best shape and pose which can minimize the Euclidean distance of the N $\boldsymbol{X}_i$ points from the scaled orthogonal projection of four components $(\boldsymbol{V}, \boldsymbol{R}, t, s)$ in which $\boldsymbol{R}$ is a 3 by 3 rotation matrix of real numbers, t is a translation of 2-tuple of real numbers and s scale is a positive real number. These are the three pose parameters. By fitting a 2-D image onto a 3-D image, the 3-D shape $\boldsymbol{S}$ and texture $\boldsymbol{T}$ of the 3-D face are obtained, as (1) and (2), and then by using these shapes and textures, 3-D faces can be constructed. Fig. 2 shows two 2-D face images, which can be chosen completely by accident, and their respective constructed 3-D image.

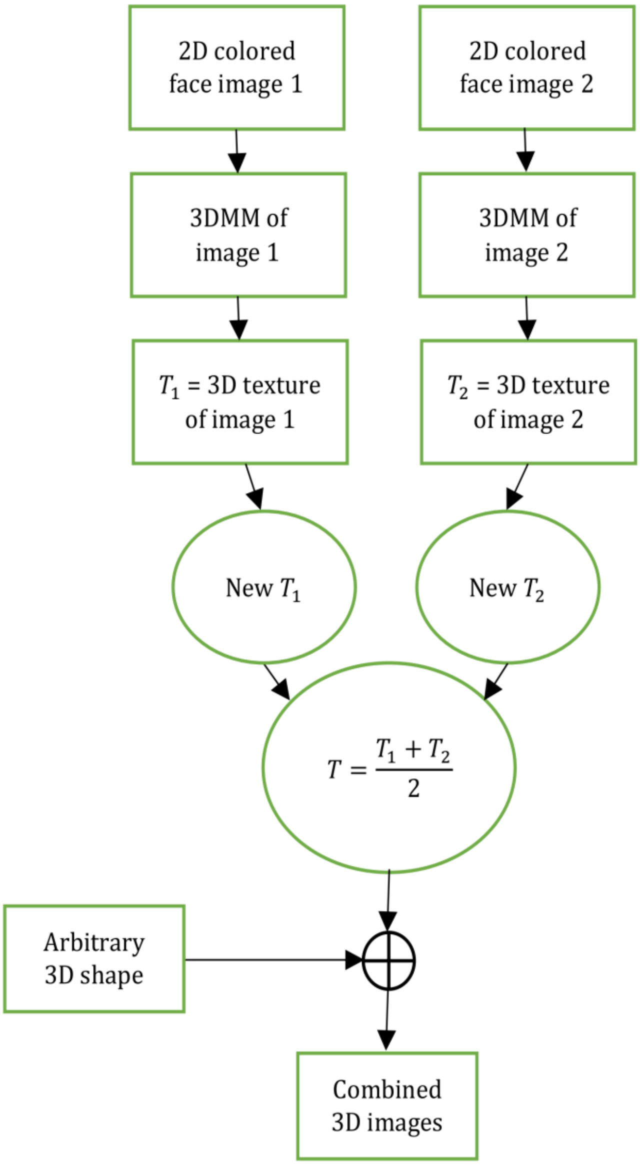

Figure 1: Flowchart of extracting 3-D shapes and textures and combining them.

Figure 2: 2-D face images and their obtained 3-D models.

The shape and texture of Obama's face are $\boldsymbol{S}_1, \boldsymbol{T}_1$, respectively, and the shape and texture of Trump's image are $\boldsymbol{S}_2, \boldsymbol{T}_2$, respectively.

Now, if we use Obama's texture which is a 3-D matrix $\boldsymbol{T}_1$, and Trump's 3-D shape matrix which is shown by $\boldsymbol{S}_2$, Fig. 3 will be obtained.

Figure 3: Obama's texture on Trump's shape matrix.

As it can be seen in Fig. 3, the 3-D shape for detecting a person's face is not so important, and only face texture for reconstructing a human face is sufficient. For this reason, if we want to apply steganography onto a 3-D face image, it would be better to ignore half of the information, which is the shape matrix, and only use the texture matrix which is described in section 4.

Moreover, if we want to combine the obtained 3-D images from the 2-D images, with each other, an appropriate way is to use coefficients of principle components. For 53490×3 dimentional shape matrix $\boldsymbol{S}_1$, by using PCA with alternating least squares ALS algorithm [14] we have:

$$[coeff\boldsymbol{S}_1 . score\boldsymbol{S}_1 . \mu] = pca\ (\boldsymbol{S}_1) \tag{12}$$

where 3×3 dimensional coefficient $coeff\boldsymbol{S}_1$ is obtained which its columns arranged in an increasing order and also include any of the 3 principle component coefficients. The 53490×3 dimensional score matrix is also shown by $score\boldsymbol{S}_1$. The 1×3 dimensional average of any variable matrix is shown by μ_{S_1}.

The same thing can be done on $\boldsymbol{S}_2, \boldsymbol{T}_1, \boldsymbol{T}_2$, again by using the principle component analysis together with ALS algorithm. Now, if the new shapes and textures are obtained using (13), the 3-D face models of Fig. 4 can be then constructed.

$$\begin{cases} \boldsymbol{T}_1 = score\boldsymbol{T}_1 * coeff\boldsymbol{T}_2' + repmat(\mu_2, 53490, 1) \\ \boldsymbol{T}_2 = score\boldsymbol{T}_2 * coeff\boldsymbol{T}_2' + repmat(\mu_2, 53490, 1) \\ \boldsymbol{S}_1 = score\boldsymbol{S}_1 * coeff\boldsymbol{S}_2' + repmat(\mu\boldsymbol{S}_2, 53490, 1) \\ \boldsymbol{S}_2 = score\boldsymbol{S}_2 * coeff\boldsymbol{S}_2' + repmat(\mu\boldsymbol{S}_2, 53490, 1) \end{cases} \tag{13}$$

Figure 4: Obama's and Trump's transformed shapes and textures as in (13).

To construct the left face in Fig. 4, the new shape and texture matrices $\boldsymbol{S}_1, \boldsymbol{T}_1$ have been used, and to construct the right face in Fig. 4, the new shape and texture matrices $\boldsymbol{S}_2, \boldsymbol{T}_2$ have been used.

Now, by the transform which has been applied in (13), only by taking a simple average in (14), we can see a satisfactory behavior of two politicians as the face in the figure Fig. 5.

$$\begin{cases} \boldsymbol{S}_m = (\boldsymbol{S}_1 + \boldsymbol{S}_2)/2 \\ \boldsymbol{T}_m = (\boldsymbol{T}_1 + \boldsymbol{T}_2)/2 \end{cases} \tag{14}$$

Figure 5: Combined Obama and Trump 3-D models.

In Fig. 6 some other input 2-D colored face images and their combined 3-D models are shown.

4. 3-D FACE IMAGE STEGANOGRAPHY

As explained before, if we want to apply steganography onto a 3-D face image, it would be better to ignore half of the information, which is the shape matrix, and only use the texture matrix, which is described in this section.

As it was mentioned previously, a texture matrix is a 53490×3 dimensional matrix.

To hide this 3-D image, we add some zeros to the end of this matrix to obtain a 53824×3 dimension matrix, hence we have $232 \times 232 = 53824$. By using the s-econd dimension of the resulted matrix, we change it to three vertical 53824×1 vectors, and after that we reshape each of these vectors to a 232×232 square matrix as the symbols for red, green and blue colors, respectively.

The values of these matrices which might not be a number, will be replaced by zero, (also for better results, other techniques could be used, such using the adjacent points to recover the missing texture points). Now, by using singular value decomposition, any of the colors is decomposed to a 3-tumples like $(\boldsymbol{U}_{t_i}, \boldsymbol{S}_{t_i}, \boldsymbol{V}_{t_i})$, by subindices $i = r, g, b$ which are related to their respective colors.

Also, a colored image is used as the cover image and its size will be changed into to a $464 \times 464 \times 3$ dimensional matrix that we represent by matrix $\boldsymbol{C}$ (the cover matrix can also be any other 3-D texture matrix and in this way, we also do the same thing as mentioned above). Matrix $\boldsymbol{C}$, by using a single-level discrete 2-D wavelet transform, is decomposed to a 4-tuple$(\boldsymbol{CA}, \boldsymbol{CH}, \boldsymbol{CV}, \boldsymbol{CD})$, all with the same $232 \times 232 \times 3$ dimensions. $\boldsymbol{CA}$ is the approximation coefficient matrix which has a low frequency, and three so called detailed coefficient matrices by the names of horizontal matrix $\boldsymbol{CH}$ and vertical matrix $\boldsymbol{CV}$ which both have medium frequencies, and diagonal matrix $\boldsymbol{CD}$ which has a high frequency.

Now, because of the third dimension of $\boldsymbol{CD}$, we decompose it to 3 $\boldsymbol{CD}_r, \boldsymbol{CD}_g, \boldsymbol{CD}_b$ matrices each by 232×232 dimensions for the symbol of red, green, and blue colors, respectively. In the following, by using singular decomposition, each color matrix, like the matrices which were obtained by texture, is decomposed to 3-tuples $(\boldsymbol{U}_{c_i}, \boldsymbol{S}_{c_i}, \boldsymbol{V}_{c_i})$ using subindices which belong to the three colors.

Now, for red, green, and blue colors, the obtained singular value matrix from the cover image is replaced by $\boldsymbol{S}_{c_i}$ of color i, and then added to a tenth of $\boldsymbol{S}_{t_i}$. For example for the red color, we can obtain a new value of

$$\boldsymbol{CD}_{new_r} = \boldsymbol{U}_{c_r} \times (\boldsymbol{S}_{c_r} + 0.1 * \boldsymbol{S}_{t_r}) \times \boldsymbol{V}_{c_r}^T. \quad (15)$$

Then, by concatenation of these 3 red, green, and blue matrices, the new value of $\boldsymbol{CD}_{new}$ will be obtained and then by using inverse wavelet decomposition on the 4-tuple $(\boldsymbol{CA}, \boldsymbol{CH}, \boldsymbol{CV}, \boldsymbol{CD}_{new})$, the stego image will be constructed. The procedure of steganography which is described in this section can be seen in Fig. 7.

To extract a hidden image, we should do the above mentioned process in reverse. At first, on the stego image, a 2-D wavelet decomposition for rebuilding 4-tuple $(\boldsymbol{wA}, \boldsymbol{wH}, \boldsymbol{wV}, \boldsymbol{wD})$ is performed.

Because of the third dimension of $\boldsymbol{wD}$, we decompose $\boldsymbol{wD}$ to three red, green and blue color images, and also on any color by using singular value decomposition, the 3-tuple $(\boldsymbol{U}_{e_i}, \boldsymbol{S}_{e_i}, \boldsymbol{V}_{e_i})$ with three subindices $i = r, g, b$ which belong to any color, is obtained.

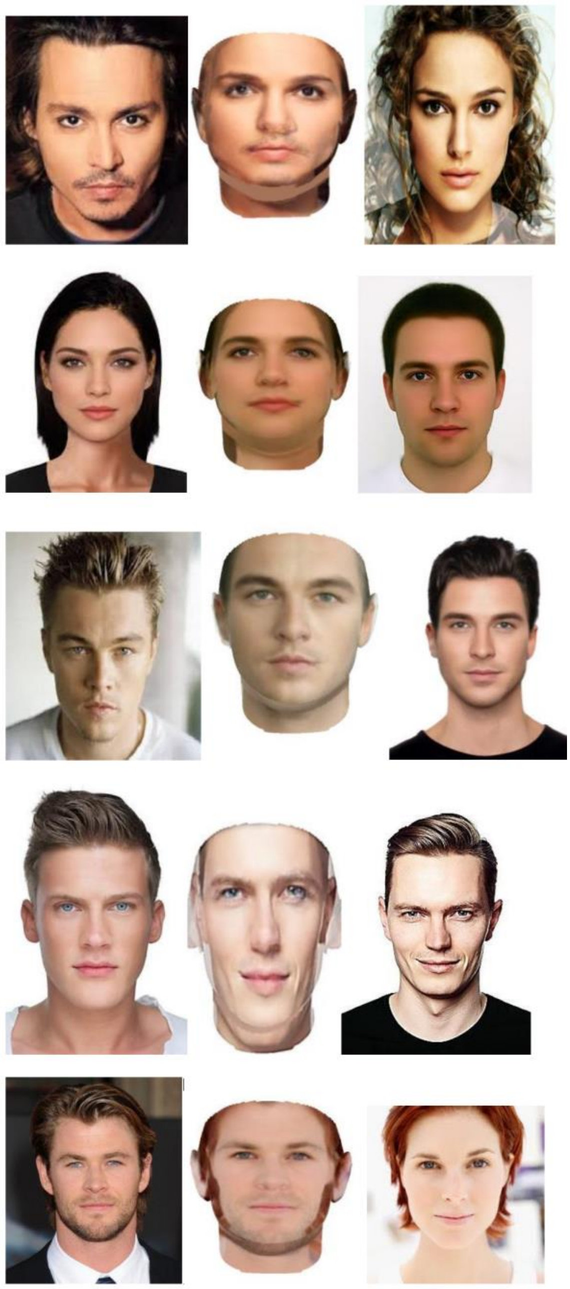

Figure 6: 3-D combining some arbitrary face images.

Also, for calculating $\boldsymbol{S}_{t_i}$, we should subtract the value of $\boldsymbol{S}_{c_i}$ from obtained singular value, $\boldsymbol{S}_{e_i}$ and multiply it by 10 (the inverse of what was mentioned before).

Next, the new texture column components from the result of $\boldsymbol{U}_{e_i} \times \boldsymbol{S}_{t_i} \times \boldsymbol{V}_{e_i}^T$ for each color is calculated. This procedure is shown in Fig. 8. We can implement the resulted texture on any shape to reconstruct the desired 3-D face image .

As an example, the images for the mentioned process are as in Fig. 8, which from left to right are, the original cover image, the 3-D used original image, the stego image and the desired extracted image.

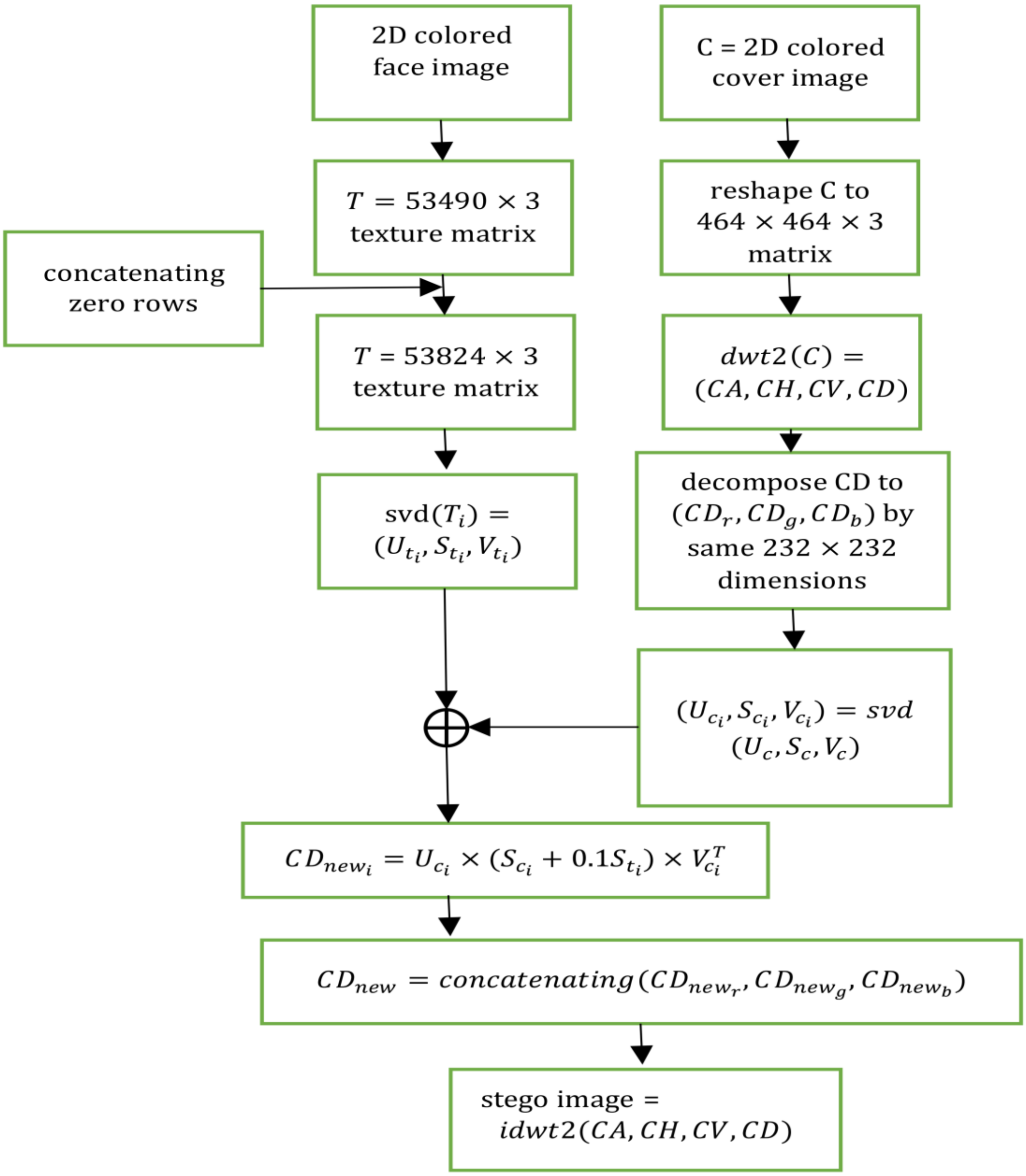

Figure 7: Steganography of 3-D texture of face images, where $\boldsymbol{i}$ is equal to three red, green, and blue colors.

A known used criterion for measuring the image quality is the peak of signal to noise ratio [15]. The peak of signal to noise ratio of the two original cover and stego images of Fig. 9 is 81.51 (dB) using equation (16) below, where MSE is the mean square error and S is the maximum pixel value.

$$PSNR = -10 \log_{10} \frac{MSE}{S^2}. \tag{16}$$

Also, some other cover and hidden images with their PSNR values are shown in Fig. 10.

In all results, the destruction of cover image by hiding texture in it using our method is very insignificant and all the PSNR values are close to 80. A similar result is also achieved by Chao [16].

5. CONCLUSION

Hiding the secretive information from the enemies' view in a way that they do not be at all skeptical to the existence of hidden messages helps to a safe communication. In this paper, by using the idea of extracting 3-D face texture and embedding it in other images has proved that how by using only half of the face information, which is texture, we can appropriately reconstruct face images. In addition, it has been shown that how by combining two textures and without any stricture on shape matrix, we can combine two faces pleasantly.

$$(wA, wH, wV, wD) = dwt2(stego\ image)$$

$$\downarrow$$

$$\left(wD_{e_r}, wD_{e_g}, wD_{e_b}\right) = decompose(wD)$$

$$\downarrow$$

$$(U_{e_i}, S_{e_i}, V_{e_i}) = svd(wD_{e_i})$$

$$\downarrow$$

$$S_{t_i} = (S_{e_i} - S_{c_i}) \times 10$$

$$\downarrow$$

$$T_i = U_{t_i} \times S_{t_i} \times V_{t_i}^T$$

$$\downarrow$$

$$T = (T_r, T_g, T_b)$$

Figure 8: Extracting hidden texture for constructing 3-D face image where i is equal to three red, green and blue colors.

Figure 9: Steganography of 3-D texture which from left to right are original cover image, 3-D used original image, the stego image, and the desired extracted image.

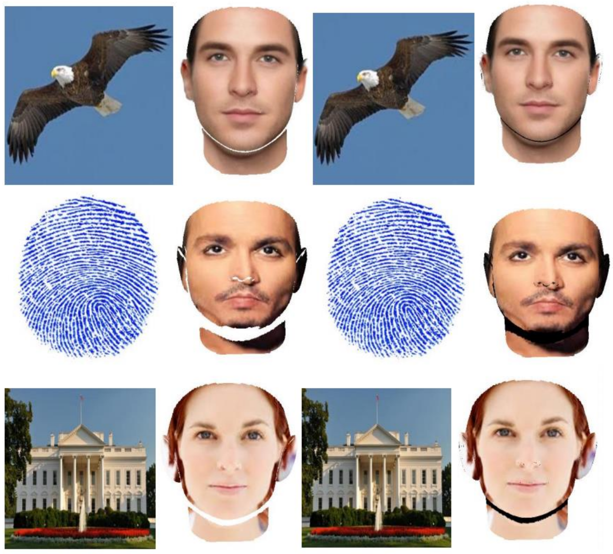

Figure 10: Steganography and extracting three other textures with PSNR equal to 79.4391, 81.3087, and 80.9286 dB, respectively.

REFERENCES

[1] A. Bas, W. A. Smith, T. Bolkart, and S. Wuhrer, "Fitting a 3-D morphable model to edges: A comparison between hard and soft correspondences," arXiv preprint arXiv: 1602.01125, 2016.

[2] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3-D faces," *in Proc. the 26th annual conference on Computer Graphics and Interactive Techniques*, Los Angeles CA, USA, 1999.

[3] J. Kittler, P. Huber, Z. H. Feng, G. Hu, and W. Christmas, "3-D morphable face models and their applications," *in Proc. International Conference on Articulated Motion and Deformable Objects*, Palma de Mallorca, Spain 2016.

[4] A. Patel and W. Smith. "3d morphable face models revisited. In *in Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, pp. 1327–1334, Florida, USA, 2009.

[5] X. Zhu and D. Ramanan. "Face detection, pose estimation, and landmark localization in the wild," *in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Rhode Island, 2012.

[6] C. C. Chang, T. S. Chen, and K. F. Hwang, "Electronic image techniques," Taipei: Unalis, 2000.

[7] N. F. Johnson and S. Jajodia, "Exploring steganography: Seeing the unseen," *IEEE Computer*, vol. 31, no. 2, pp. 26-34, Feb. 1998.

[8] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3-D face model for pose and illumination invariant face recognition*," in Proc. IEEE Intl. Conf. on Advanced Video and Signal based Surveillance*, Genova, Italy, 2009.

[9] V. Blanz and T. Vetter, "Face recognition based on fitting a 3-D morphable model," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1063-1074, 2003.

[10] J. R. T. Rodríguez, "3D face modelling for 2D+3D face recognition," Ph.D. dissertation, Dept. Electron. Eng., Univ. Surrey, Guildford, U.K., 2007.

[11] K. Pearson, "On lines and planes of closest fit to systems of points in space," *Philosophical Magazine,* vol. 2, no. 11, pp. 559-572, 1901.

[12] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B.P. Flanner, "Singular value decomposition," *Solution of Linear Algebraic Equation from Numerical Recipes in C*, pp. 59-70, 1992.

[13] T. P. Minka, "Automatic choice of dimensionality for PCA," M.I.T Media Laboratory Perceptual Computing Section Technical Report, Cambridge, 2000.

[14] H. A. Kiers, "Setting up alternating least squares and iterative majorization algorithms for solving various matrix optimization problems," *Computational Statistics and Data Analysis*, vol. 41, no. 1, pp. 157-170, 2002.

[15] C. G. Rafael and E. W. Richard. Digital Image Processing. Prentice-Hall, Upper Saddle River, NJ, USA, 3rd edition, 2006.

[16] M. W. Chao, et al., "A high capacity 3D steganography algorithm," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 2, pp. 274-284, 2009.

BIOGRAPHIES

Mohsen Moradi received the B.Sc. degree from the University of Isfahan, Isfahan, Iran, in 2011, and the M.Sc. degree from Amirkabir University of Technology, Tehran, Iran in 2013, both in pure mathematics. His research interests are in coding theory with focus on lattice codes, lattice-based cryptography, information theory, image processing, and artificial intelligent algorithms.

Mohammad Reza Sadeghi received the Ph.D. degree from Carleton University, Canada, in 2003. In 2004, he was a Post-Doctoral Fellow with Carleton University. He was given the Marwah award for outstanding Ph.D. studies and received a scholarship from the Fields Institute of Mathematical Studies. He is currently an Associate Professor with the Department of Mathematics and Computer Science, Amirkabir University of Technology, Tehran, Iran, as well as, adjunct research professor at School of Mathematics and Statistics, Carleton University, Ottawa, Canada. His field of interests are coding theory, lattice codes, and lattice-based cryptography.

How to cite this paper:

M. Moradi and M. R. Sadeghi, "Combining and steganography of 3-D face textures," Journal of Electrical and Computer Engineering Innovations, vol. 5, no. 2, pp. 93-100, 2017.

DOI: 10.22061/JECEI.2017.690

URL: http://jecei.sru.ac.ir/article_690.html