International Journal of Electrical Electronics & Computer Science Engineering
Volume 5, Issue 1 (February, 2018) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

Scale Factor Optimization for a Robust Multi-Biometric Watermarking

Aysun Tutak Erözen[1], Nihan Kahraman[2]
[1]Central R&D Department, Arçelik A.Ş., İstanbul, Turkey
[2]Electronics & Communication Engineering Department, Yildiz Technical University, İstanbul, Turkey
[1]aysun.erozen@arcelik.com, [2]nicoskun@yildiz.edu.tr

Abstract: ***In this study multi-biometric images are embedded into a host image that includes RGB channels. The watermarking method is based on to decompose both host image and watermark images into singular values and mixing these singular values with a scale factor. At first, scale factor is selected randomly and same scale factor is applied for all channels. From the experiments, it is seen that choosing scale factor has essential role for performance. There are two performance criteria in watermarking. First criterion is invisibility which means that the watermark must not be detected with eye without some operations. Second criterion is robustness. If a watermarking is robust, it means that the watermarks can be retrieved back successfully even though there are some attacks to the watermarked image. Optimization of scale factor balances both performance criteria. In this study two different multi-dimensional iterative optimization methods are used for same fitness function. These methods are Spiral optimization and Particle swarm optimization. Additional to that, polynomial regression is performed to optimize scale factors of each channel separately for three different fitness functions.***

Keywords: Multi-Biometric Watermarking, Particle Swarm Optimization, Polynomial Regression, Spiral Optimization, Singular Value Decomposition.

I. INTRODUCTION

With widespread use of internet, one of the most important needs is to secure digital information. Encryption is generally used for securing data by making the secret messages unreadable and or meaningless. However, these irregular messages usually attract some unintended observers' attention. Therefore, watermarking techniques are used for embedding data into an image, video or voice in literature [1-14].

As biometrics becomes more prevalent in everyday life, such as bank operations, passports even some national identity cards, there is much need in securing the data associated with the identification of each individual. Furthermore, biometric databases are usually stored in data centers where they are sometimes distributed over insecure public access networks and problems may occur such as stealing, copying and duplication, distribution or imitation of this information by malicious people.

In order to hide the information that is used to enter a system, watermarking techniques can be used. Because to extract the hidden information it is a must to know the key and the way how this key is used. It is harder to get necessary information in this way. In case of being stolen, to prevent imitation of the information multi-biometry is used. Because biometric information is personal and cannot be imitated. Even though trying to simulate it would be very difficult to repeat three biometric data at the same time. Conventional watermarking process can be shown as Fig. 1.

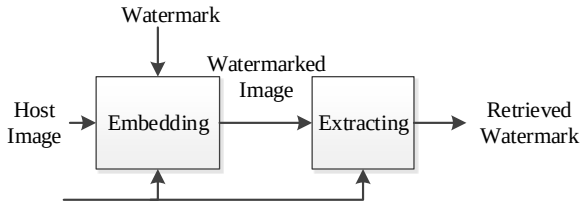

Fig. 1. Conventional Watermarking Process

Extracting part in this watermarking process is the inverse of embedding where the recovered watermark and the initial watermark are expected to be same. But there may be some attacks and this case the recovered watermark and the initial watermark may not be same. Possible attack implementation is shown in Fig. 2.

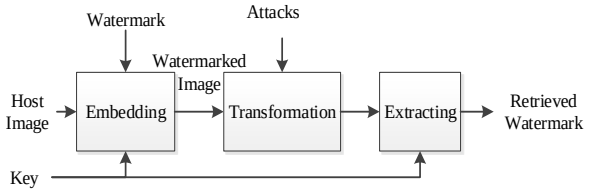

Fig. 2. Attack Implementation to Typical Watermarking Process

The performance of watermarking is analysed with robustness and invisibility parameters. Invisibility, which is measured by Peak Signal to Noise Ratio (PSNR), means that a person couldn't detect watermark without extraction process. Robustness, which is measured by Correlation Coefficient (CC) means that the watermark should be detected with extraction process even though there is an attack to the watermarked image.

There are various studies in the literature for watermarking, some are in spatial domain and some are in frequency domain. For robust application singular Value Decomposition (SVD) is a favourite method and there are various applications using SVD with optimization methods [15- 24]. In [15-16] a grayscale watermark is embedded in grayscale host. Host image is decomposed into four sub bands using Discrete Wavelet Transform (DWT) and sub band images singular values are modified with the singular values of watermark. Same watermark is embedded in four sub band images.

International Journal of Electrical Electronics & Computer Science Engineering
Volume 5, Issue 1 (February, 2018) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

In [15], Particle Swarm Optimization (PSO) is used for 400 iterations to choose scale factor. In [16], multi-objective evolutionary algorithms are used for optimization and 90 iterations have been applied. In [17], a blind watermarking is applied for binary watermark image. The host image is separated in blocks. For each block the first element is modified according to watermark bit value. The parameter used in modification process is optimized with genetic algorithm. In [18], firstly Discrete Cosines Transform (DCT) is applied to the gray scale host image then DWT is applied for mid band of DCT coefficients. After DWT process there is four sub bands. The binary watermark singular values are embedded to each sub band by modifying singular values of host. The coefficient is optimized with genetic algorithms and PSO. In [19] both the host and watermark images are in RGB format. Each channel of watermark image is embedded in corresponding channel of host image. Multi objective genetic algorithm is used as optimization method. In [20] DWT is applied to the host image than one middle frequency sub band is selected to perform DCT and SVD sequentially. SVD coefficients are modified with scale factor which is optimized by PSO. In [21] gray scale watermark is inserted in RGB image in three different ways. Firstly, same watermark is embedded to each channel, secondly each watermark is embedded into one channel, thirdly one watermark is separated into three parts and each part is embedded in one channel. Watermark separating gives the best result. Secondly, separated watermark embedding is used with PSO and better results are obtained because of different scale factors for each channel. In [22] gray level watermark is embedded into a gray level host. Host image is decomposed with Haar wavelet, High-Low (HL) and Low-High (LH) components' singular values are modified with the principal components of watermark. Scale factor is determined with PSO. In [23] gray level watermark is embedded into gray level host by modifying singular values. PSO algorithm is used to find scale factor. In [24] gray level host image is divided into blocks. Each blocks singular value is modified with watermark singular values. The scale factor is chosen with firefly optimization.

In present study three biometric images palm print [28], iris [29] and ear [30] are used as watermark. They embedded into three channels of RGB host image using SVD method. Scale factor is selected with both iterative and analytical methods. Different fitness functions are selected to compare the performance.

II. PRELIMINARIES

In this study multi-biometric images are embedded into an RGB host image. The watermarking method is based on to decompose both host image and watermark images into singular values and mixing these singular values with a scale factor. As will be detailed in experimental results, the choosing of scale factor is important in means of robustness and invisibility. In order to choose the scale factor, newer optimization methods are used in this study. This section gives preliminary information about the methods used in experiments.

A. Singular Value Decomposition:

A matrix $A\epsilon\mathbb{R}^{mxn}$ can be decomposed into two orthogonal matrices $U\epsilon\mathbb{R}^{mxm}$, $V\epsilon\mathbb{R}^{nxn}$ and one diagonal matrix $S\epsilon\mathbb{R}^{mxn}$.

$$A = USV^T \tag{1}$$

$$A = \begin{bmatrix} u_{11} & \cdots & u_{1m} \\ \vdots & \cdots & \vdots \\ \vdots & \cdots & \vdots \\ u_{m1} & \cdots & u_{mm} \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & 0 & \cdots & \sigma_{m,n} \end{bmatrix} \begin{bmatrix} v_{11} & \cdots & v_{1n} \\ \vdots & \ddots & \vdots \\ v_{n1} & \cdots & v_{nn} \end{bmatrix} \tag{2}$$

U and V matrices are eigenvectors of AA^T and A^TA matrices sequentially. $SS^T$and S^TS diagonal matrices have diagonal elements which are eigenvalues of AA^T and A^TA matrices sequentially. The diagonal matrix involves singular values.

B. Spiral Optimization:

Spiral optimization is a new iterative method for multipoint searching and it is first proposed by Tamura and Yasuda in [25], [26]. It is a metaheuristics method based on analogy of spiral phonemea in nature. 2D and 3D spirals are shown in Fig. 3 and Fig. 4. The spiral model starts with initial states $X_i(0) \in \mathbb{R}^n$ and converges to the spiral center where spiral radius gradually decreases.

Algorithm in N dimension:

Step 0: Preparation

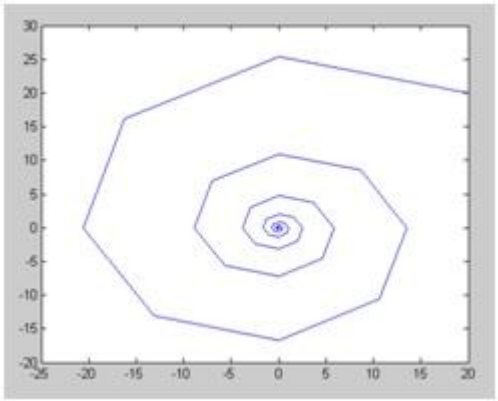

Fig. 3. Trajectory of 2D spiral with r=0.9 and $\theta = \frac{\pi}{4}$

In preparation stage the number of search points (m≥2), rotation angle for each spiral ($0 < \theta < 2\pi$) , radius for

International Journal of Electrical Electronics & Computer Science Engineering
Volume 5, Issue 1 (February, 2018) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

each spiral $(0 < r < 1)$ and iteration number $(kmax)$ are set.

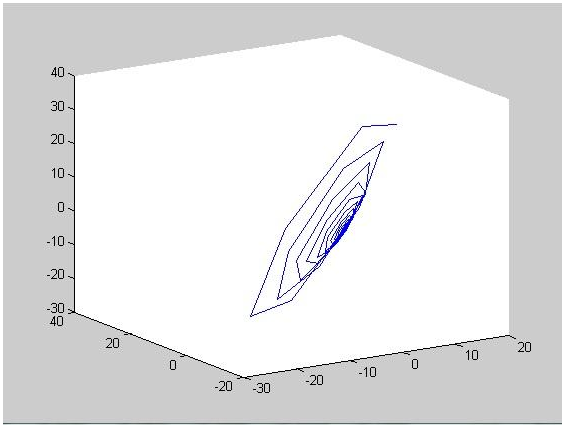

Fig. 4. Trajectory of 2D spiral with r=0.9 and $\theta = \frac{\pi}{4}$

Step 1: Initialization

For $k = 0$, the initial values of N dimensional m search points $X_i(0)$ are determined as in Equation (3).

$$Xi(0) \in \mathbb{R}^n,\ i=1,2,3,\ldots..m \quad (3)$$

Fitness function is calculated using initial values of search points. The search point value which minimizes the fitness function is labelled as center point X^* as following Equations (4) and (5).

$$X^* = X_{ig}(0), \quad (4)$$

$$ig = argmin\, f(X_i(0)),\ \ i=1,2,3,\ldots..m \quad (5)$$

Step 2: Updating search points

All search points are updated using rotation matrix $S_n(r,\theta)$ and center point X_*using Equations (6), (7) and (8).

$$X_i(k+1) = S_n(r,\theta)X_i(k) - (S_n(r,\theta) - I_n)X^*\ \ i = 1,2,3,\ldots..m \quad (6)$$

$$S_n(r,\theta) = r.R(\theta) \quad (7)$$

$$R(\theta) = \prod_{i=1}^{n-1}(\prod_{j-1}^{i} R_{n-1,n+1-j}(\theta)) \quad (8)$$

Rotation matrix $R_{i,j}(\theta)$ has nxn dimension for n dimensional optimization as in Equation (9).

Step 3: Update center point X_* .

The center point is updated using Equations (10) & (11)

$$X^*(k+1) = X_{ig}(k+1), \quad (10)$$

$$i_g = argmin\, f(X_i(k+1)),\ \ i=1,2,3,\ldots..m \quad (11)$$

Step 4: Check termination criteria.

If $k = k_{max}$ then stop. Otherwise, set $k = k + 1$ and return Step 2.

$$R_{i,j}(\theta) = \begin{array}{c} \\ i \\ \\ \\ \\ j \\ \\ \end{array} \begin{array}{c} \begin{array}{cccccc} i & & & & j & \end{array} \\ \begin{vmatrix} \ddots & & & & & \\ & \boldsymbol{cos(\theta)} & & & \boldsymbol{sin(\theta)} & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & \boldsymbol{-sin(\theta)} & & & \boldsymbol{cos(\theta)} & \\ & & & & & \ddots \\ & & & & & & 1 \end{vmatrix} \end{array} \quad (9)$$

C. Particle Swarm Optimization:

Particle swarm optimization was developed by Kennedy and Eberhart in 1995, inspired by the food search behaviour of animal species such as bird and fish [27]. The algorithm starts with definition of random points for food search. Search points are updated according to global best point and the best value of each point own self till that time. The algorithm is as follows.

Algorithm in N Dimension:

Step 0: Preparation

In this step the number of initial points ($m \geq 2$), coefficients c_1 ,c_2 and iteration number (k_{max}) are set.

Step 1: Initialization

For $k = 0$ the initial values of m points are specified in N dimension as following Equation (12).

$$X_i(0) \in \mathbb{R}^n,\ i=1,2,3,\ldots..m \quad (12)$$

The fitness function is calculated for initial point values. The point which gives minimum result is called as global best point g_{best} as in next Equations (13) and (14).

$$g_{best} = X_{ig}(0), \quad (13)$$

$$i_g = argmin\, f(X_i(0)),\ \ i=1,2,3,\ldots..m \quad (14)$$

In this step, the position of each points are set as best point p_{best} for themselves as can be seen from Equation (15).

$$p_{best} = X_i(0),\ i=1,2,3,\ldots..m \quad (15)$$

After defining g_{best} and p_{best} all search points are updated using global best point g_{best} and best point value ve p_{best}.

Step 2: Update search points

After initialization of search points, all points have new positions. These positions are used to recalculate fitness function. The point minimizes the fitness function becomes new global best point g_{best} .

International Journal of Electrical Electronics & Computer Science Engineering
Volume 5, Issue 1 (February, 2018) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

$g_{best}=X_{ig}(k)$, (16)

$i_g=argmin\, f(X_i(k)),\ \ i=1,2,3,\ldots..m$ (17)

For each points, the fitness function value calculated for the predetermined p_{best} value is compared with the fitness function value calculated for the new point positon X_i(k) in step k. The point gives a smaller value for the fitness function becomes the new $pbest$.

To update the search points below Equation (18) is applied.

$X_i(k+1) = X_i(k) + c1*rand*(p_{best}-X_i(k)) + c2*rand*(g_{best}-X_i(k))$ (18)

Step 4: Check termination criteria.

If $k = k_{max}$ then stop. Oherwise, set $k = k + 1$ and return Step 2.

D. Polynomial Regression:

Regression analysis is used to model the expected value of a dependent variable y in terms of the value of an independent variable x. In simple linear regression model Equation (19) is used, where ε is an unobserved random error.

$y=a_0 + a_1x + \varepsilon_1,$ (19)

The expected value of y as an n[th] degree polynomial, is modelled as following.

$y = a_0 + a_1x + a_2x^2 + a_3x^3 +\ldots\ldots\ldots.. + a_nx^n + \varepsilon$ (20)

This model can be expressed in matrix form in terms of design matrix **X,** a response vector $\vec{y}$, a parameter vector $\vec{a}$ and a vector $\vec{\varepsilon}$ of random errors as in Equation (21). Row number i shows the x and y values for ith sample.

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 & {x_1}^2 & \cdots & {x_1}^m \\ 1 & x_2 & {x_2}^2 & \cdots & {x_2}^m \\ 1 & x_3 & {x_3}^2 & \cdots & {x_3}^m \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & x_n & {x_n}^2 & \cdots & {x_n}^m \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix} + \begin{bmatrix} \varepsilon_0 \\ \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_m \end{bmatrix} \quad (21)$$

Pure matrix notation is:

$\vec{\boldsymbol{y}} = \boldsymbol{X}\vec{\boldsymbol{a}} + \vec{\boldsymbol{\varepsilon}}$ (22)

The vector of estimated polynomial regression coefficients using ordinary least squares estimation is

$\hat{\vec{\boldsymbol{a}}} = (\boldsymbol{X^T X})^{-1}\boldsymbol{X^T}\vec{\boldsymbol{y}}$ (23)

In this study response is fitness function, design matrix is scale factor.

III. WATERMARK EMBEDDING

Uni-biometric or multi-biometric systems can be used for identification of a person. Multi biometric systems have both advantages and disadvantages over uni-biometric systems. The main advantages are the higher accuracy and security. In uni-biometric systems False Acceptance Rate (FAR) and False Rejection Rate (FRR) errors can be high due to large population coverage. Additionally, in case of stolen of this biometric data, malicious people could reach all relevant information to that data. The disadvantage is the complexity.

To increase the security of biometric images in data center, the biometric images can be stored as watermark in a host image.

In this paper, three biometric data are embedded in RGB host Lenna 512x512 image. Biometric data chosen as palm print, iris and ear images with size of 128x128. Palm print image is embedded in R channel, iris image is embedded in G channel and ear image is embedded in B channel. The embedding procedure is same for all channels but the scale factor of each channel is different. Scale factors of each channel are defined after optimization process.

The embedding procedure for R channel is summarized as below step by step.

1. Decompose RGB host image into three channels.
2. Perform SVD to R channel of host image H_R and get U_R SR VR matrices using equation (24).
   $H_R = U_RS_RV_R^T$ (24)
3. Perform SVD to first biometric data B_1 and get U_1 S_1 V_1 matrices as follows.
   $B_1=U_1S_1V_1^T$ (25)
4. Obtain summation of S1 matrix of first biometric data B_1 and S_R matrix of R channel of host image H_R to get SRY.
   $S_{RY} = (1-\alpha_R)S_R + \alpha_RS_1$ (26)
5. Compose U_R , V_R and new matrix S_{RY} to get R channel of watermarked image.
   $W_R = U_RS_{RY}V_R^T$ (27)

Repeat this algorithm for B and G channels.

IV. WATERMARK EXTRACTING

The extracting procedure for R channel is summarized as below step by step.

1. Decompose RGB watermarked image into three channels.
2. Perform SVD to R component of watermarked image W_R and get UR SRy V_R matrices.
   $W_R = U_RS_{RY}V_R^T$ (28)
3. Extract S_1 matrix of first biometric image B_1 from R channel of watermarked image W_R .
   $S_{1,\,extracted} = [S_{RY} - (1-\alpha_R)S_R]/\alpha_R$ (29)
4. Use extracted matrix $S_{1,extracted}$ to get first biometric watermark image.
   $B_{1,\,extracted} = U_RS_{1,\,extracted}V_R^T$ (30)

Repeat this algorithm for B and G channels to get second and third biometric images.

International Journal of Electrical Electronics & Computer Science Engineering
Volume 5, Issue 1 (February, 2018) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

V. EXPERIMENTAL DETAILS

In this study Lenna 512x512 RGB image is used as host. 128x128 grayscale three biometric images are used as watermarks. Watermark images are combined with host images to generate 512x512 size. These generated new images are embedded in host. They are embedded in different locations as shown below. The reason for that is to prevent to loss of three biometric images at the same time in case of an attack applied to specific region of watermarked image. Biometric images are chosen as palm print, iris and ear of a person. Sample images are taken from CASIA palm print database, CASIA iris database and AMI ear database respectively.

Fig. 5. Watermark for R Component

Fig. 6. Watermark for G Component

Fig. 7. Watermark for B Component

In this study 9 types of attacks are applied to watermarked image. These attacks are;

1) Salt & Pepper noise (d=0,002)
2) Gauss noise (d=0,01)
3) Rotate (-1 angle)
4) Crop (20 pixels each side)
5) Resize (512-256-512)
6) Resize (512-1024-512)
7) Jpeg compression (Q=%75)
8) Jpeg compression (Q=%90)
9) Speckle noise (d=0,004)

Before optimization, same scale factor is practiced for all channels Below graphics show the performance of the watermarking by changing the scale factor.

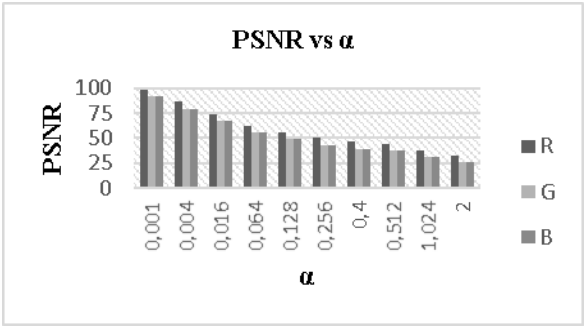

Fig. 8. PSNR Between Host Image and Watermarked Image

Figure 8 shows the PSNR value calculated between host image and watermarked image for each channel. As it seen, the increase in scale factor value results a decrease in PSNR value.

Figure 9 shows that the CC between watermark images and the retrieved ones after salt and pepper noise. The increase in scale factor up to a point result causes an increase in CC. But after a value the CC is same.

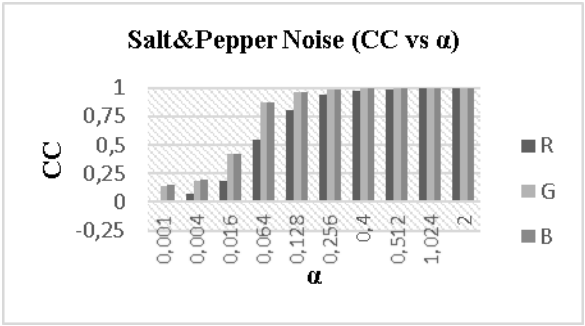

Fig. 9. CC between Watermark Images and the Retrieved Ones after Salt and Pepper Noise

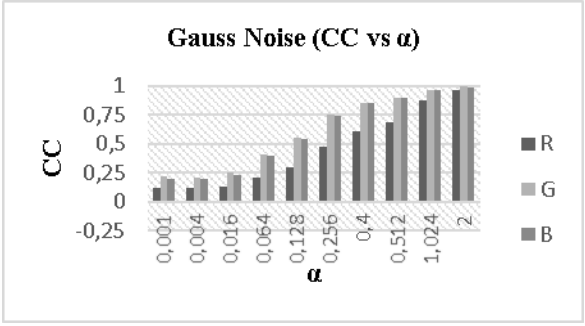

Fig. 10. CC between Watermark Images and the Retrieved Ones after Gaussian Noise

In Figure 10, it is seen that CC between watermark images and the retrieved ones increases when scale factor increases.

International Journal of Electrical Electronics & Computer Science Engineering
Volume 5, Issue 1 (February, 2018) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

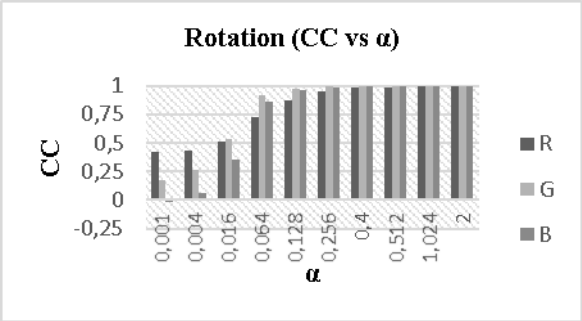

Fig. 11. CC between Watermark Images and the Retrieved Ones after Rotation Attack

Figure 11 illustrates the relationship between scale factor and rotation attack.

In Figure 12, It is seen that SVD watermarking is very robust to rotation attacks. The correlation coefficient is always nearly 1.

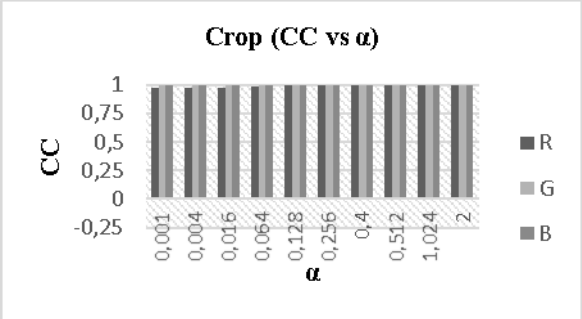

Fig. 12. CC between Watermark Images and the Retrieved Ones after Crop Attack

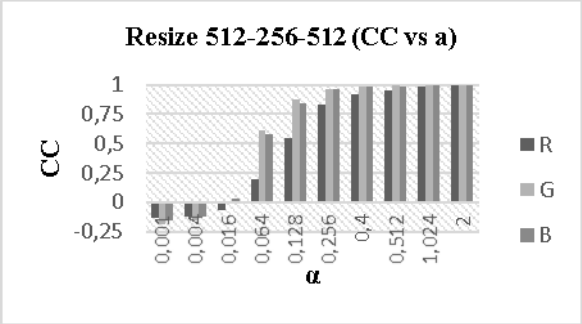

Fig. 13. CC between Watermark Images and the Retrieved Ones after Resize 512-256-512 Attack

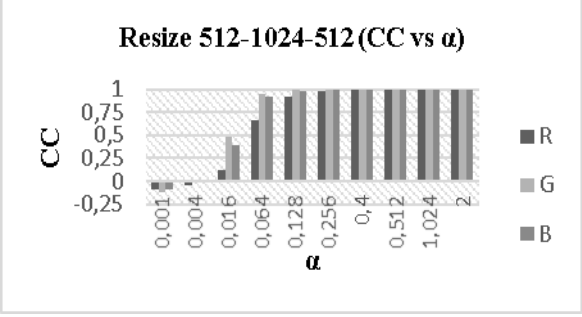

Fig. 14. CC between Watermark Images and the Retrieved Ones after Resize 512-1024-512 Attack

Figure 13 and Figure 14 shows the CC values that are retrieved after resize attacks. The performance is better for resize 512-1024-512 attack.

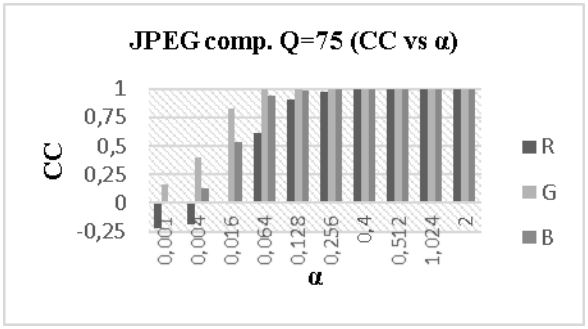

Fig. 15. CC Between Watermark Images and the Retrieved Ones after JPEG Compression Q=75 Attack

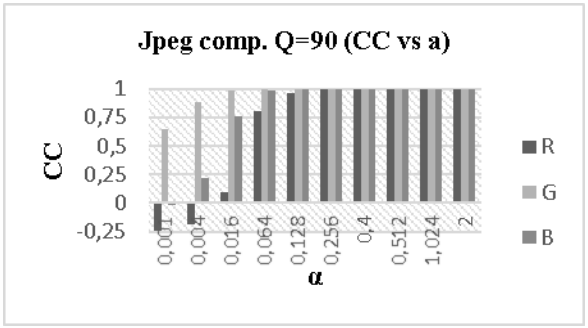

Fig. 16. CC Between Watermark Images and the Retrieved Ones after JPEG Compression Q=90 Attack

From Figure 15 and Figure 16 it can be concluded that the performance is similar for JPEG compression Q=75 and Q=90 attacks.

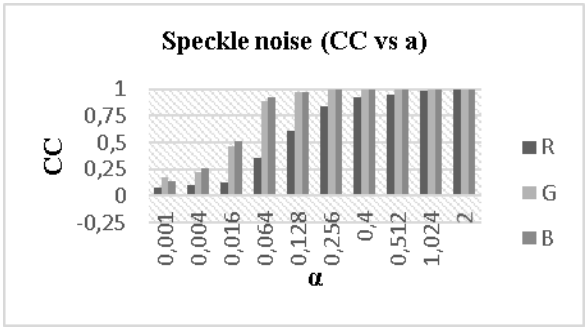

Fig. 17. CC between Watermark Images and the Retrieved Ones after Speckle Noise

As it can be seen from the figures, choosing scale factor effects the performance of watermarking. To find optimum scale factors for each RGB channels different optimization methods are used. The methods can be classified into two groups. First one is 3D optimization at the same time for each channel. For 3D optimization; Spiral Optimization and Particle Swarm Optimizations are used.

The second one is using an analytical tool to optimize scale factor for each channel independently. For this,

International Journal of Electrical Electronics & Computer Science Engineering
Volume 5, Issue 1 (February, 2018) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

the polynomial regression method is used and the optimization is done for 3 different fitness functions.

A. Finding Scale Factor with 3D Spiral Optimization:

During watermarking, first, same scale factor is used for each channel. It is seen that robustness is directly proportional with scale factor value, while invisibility is inversely proportional with scale factor value. This means that optimization of scale factor is important to achieve high performance for both robustness and invisibility. Therefore, 3D spiral optimization is used to get optimum scale factors α_R, α_G, $\alpha B$for each channel.

Spiral optimization parameters are initialized as following.

- The number of search points $m = 20$
- Rotation angles $\pi/4 < \theta i < 2\pi$, $i = 1,2,3, \ldots . m$
- Spiral radius $0{,}9 < r_i < 1$, i=1,2,3,…..mIteration number. $k_{max} = 20$

To update search points X_i , center point $X*$ is settled according to the fitness function firstly. Fitness function must be selected to maximize robustness and invisibility. The first term in the fitness function shows the robustness, it is the inverse of the correlation coefficient mean between original watermark images and retrieved ones after N applied attacks. The second terms show the invisibility. $CC_{RGB,host}$ is the correlation coefficient between host image and watermarked image after changing both images RGB to gray scale. The fitness function in Equation (31) should be minimized.

$$Fitness_{RGB} = \frac{N}{\sum_{i=1}^{N} CC_{RGB,i,wat}} - CC_{RGB,host} \quad (31)$$

$CC_{RGB,wat}$ in Equation (32) shows the correlation coefficient between original watermark and the extracted watermark after an applied attack. It is the average of correlation coefficients of R, G and B channels.

$$CC_{RGB,i,wat} = \frac{CC_{R,i,wat} + CC_{G,i,wat} + CC_{B,i,wat}}{3} \quad (32)$$

For images I_1 and I_2 the correlation coefficient is calculated as following Equation (33).

$$CC = \frac{\sum_i \sum_j (I_1(i,j) - I_{1,mean})(I_2(i,j) - I_{2,mean})}{\sqrt{\sum_i \sum_j (I_1(i,j) - I_{1,mean})^2 (\sum_i \sum_j (I_2(i,j) - I_{2,mean})^2)}} \quad (33)$$

B. Finding Scale Factor with 3D Particle Swarm Optimization:

The second 3D optimization method used in this paper is Particle Swarm optimization. Similar to 3D spiral optimization firstly initial parameters in PSO are initialized as following.

- The number of search points $m = 20$
- Coefficients $c1{=}c2{=}20$
- Iteration number. $k_{max} = 20$

After setting these parameters optimization process is started using same fitness functions with 3D spiral optimization according to equations (31), (32), (33). The algorithm is started with 20 points and iterated 20 times. It is run for 5 times.

C. Finding Scale Factor with Polynomial Regression:

After 3D optimization with iterative methods, polynomial regression is used to model fitness function for each channel separately. Totally three models are obtained and the scale factors α_R, αG, α_B are found from these models. Using polynomial regression, three different applications are handled and the results are compared within themselves and also 3D iterative optimization methods.

In first application, the fitness function is similar to that used in 3D optimization methods. In this application fitness functions are separately defined for R, G and B channels as in Equations (34) (35) and (36).

$$F_R = \frac{N}{\sum_{i=1}^{N} CC_{R,i,wat}} CC_{R,host} \quad (34)$$

$$F_G = \frac{N}{\sum_{i=1}^{N} CC_{G,i,wat}} CC_{G,host} \quad (35)$$

$$F_B = \frac{N}{\sum_{i=1}^{N} CC_{B,i,wat}} CC_{B,host} \quad (36)$$

$CC_{R,i,wat}$ shows the correlation coefficient between original watermark and the extracted watermark after an applied attack for R channel. $CC_{G,i,wat}$, $CC_{B,i,wat}$ have same definitions for other channels. $CCR,host$ is the correlation coefficient between R host image and watermarked R image. $CC_{G,host}$ and $CC_{B,host}$ have same definitions for G channel and B channel. With the help of fitness functions and polynomial regression Equations (37), (38) and (39) are developed.

$$-0.002901\alpha^2 + 0.019218\alpha - 0.02913 = 0 \quad (37)$$

$$-0.019146\alpha 2 + 0.0691\alpha - 0.0569 = 0 \quad (38)$$

$$-0.03918\alpha^2 + 0.1484\alpha - 0.1098 = 0 \quad (39)$$

There are no real roots in defined range of functions, so the derivations of these functions are used to find scale factors α_R, α_G, α_B gives minimum result.

In second application the fitness functions are specified as in Equations (40) (41) and (42) sequentially for R channel, G channel and B channel.

$$F_R = FSNR_{R,imp.} - \frac{1}{N}\sum_{i=1}^{N} PSNR_{R,rob.} \quad (40)$$

$$F_G = FSNR_{G,imp.} - \frac{1}{N}\sum_{i=1}^{N} PSNR_{G,rob.} \quad (41)$$

$$F_B = FSNR_{B,imp.} - \frac{1}{N}\sum_{i=1}^{N} PSNR_{B,rob.} \quad (42)$$

International Journal of Electrical Electronics & Computer Science Engineering
Volume 5, Issue 1 (February, 2018) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

The PSNR (in dB) is defined as in Equation (43). For 8 bit images it is 255. Mean Square Error (MSE) between two images I and K is formulated in Equation (44). m and n shows the width and height of images, i and j shows the pixel position.

$$PSNR = 10\log_{10}\left(\frac{MAX_I^2}{MSE}\right) = 20\log_{10}\left(\frac{MAX_I}{MSE}\right) \quad (43)$$

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2 \quad (44)$$

The first term in the fitness functions shows the impercibility, it is the PSNR between host image and watermarked image. The second terms show the robustness, it is the average PSNR between original watermark and retrieved one after N applied attacks. Fitness values should be zero to find optimum scale factors. Benefit from fitness functions (40) (41) (42) and polynomial regression, below mathematical models are derived. Equation (45) is a model for R channel, equation (46) is a model for G channel and equation (47) is model for B channel.

$$-51.45\alpha^3 + 113.6\alpha^2 - 107.9\alpha + 33.37 = 0 \quad (45)$$

$$-51.50\alpha^3 + 112.8\alpha^2 - 107.1\alpha + 30.11 = 0 \quad (46)$$

$$-137\alpha^3 + 219.2\alpha^2 - 154.6\alpha + 30.60 = 0 \quad (47)$$

The roots found in the defined range of above functions are scale factors α_R, α_G, α_B.

In third application; fitness functions are proposed as below Equations (48) (49) and (50).

$$F_R = CC_{R,imp.} - \frac{1}{N}\sum_{i=1}^{N} CC_{R,rob.} \quad (48)$$

$$F_G = CC_{G,imp.} - \frac{1}{N}\sum_{i=1}^{N} CC_{G,rob.} \quad (49)$$

$$F_B = CC_{B,imp.} - \frac{1}{N}\sum_{i=1}^{N} CC_{B,rob.} \quad (50)$$

The first term in the fitness functions shows the invisibility, it is the CC between host image and watermarked image. The second terms show the robustness, it is the average CC between original watermark and retrieved one after N applied attacks.

The functions that makes the fitness value 0 are found as following Equations (51) (52) and (53).

$$-0.000696\alpha^3 + 0.006972\alpha^2 - 0.02766\alpha + 0.03615 = 0 \quad (51)$$

$$-0.002519\alpha^3 + 0.01469\alpha^2 - 0.03443\alpha + 0.02490 = 0 \quad (52)$$

$$-0.02139\alpha^3 + 0.07108\alpha^2 - 0.1015\alpha + 0.04883 = 0 \quad (53)$$

The roots found in the defined range of above functions are scale factors α_R, α_G, α_B.

VI. Results And Discussions

Firstly, different scale factors are used for SVD watermarking and seen that performance results are highly dependent on scale factors. According to fitness function we may need 3D optimization or 1D optimization. For 3D optimization we have to use iterative methods. Because if polynomial regression is used there will be three unknown parameters in a model. This model with a single function cannot be solved analytically. In addition to that, there are lots of combination of three scale factors so it is not easy to find. As 3D optimization methods Spiral optimization and Particle swarm optimization is executed five times. The results show that both optimization methods are convenient to optimize 3D watermarking based on SVD. But the average fitness function value is less for spiral optimization. In most studies particle swarm optimization is used in literature. This study shows spiral optimization is better for this application. If 1D optimization is done, polynomial regression can be used instead of iterative methods. For an unknown it is easier to find solution to a model and it gives accurate results as well.

From the Table (1) it can be noticed that to maximize invisibility of watermarking, Model 2 is most proper selection in both terms of PSNR and CC. To maximize robustness 3D spiral optimization is the suitable method. If fitness function values are compared the minimum values are achieved for Model 3.

Table I. Results for Different Optimization Methods

	3D Spiral Optimization (Average)	3D Particle Swarm Optimization(Average)	Model 1	Model 2	Model 3
PSNR - grayscale host	28,4673	28,5044	45.3602	55,6133	45,4092
CC - grayscale host	0,9977	0,9974	0.9979	0,9997	0,9980
PSNR – between host & watermarked R	29,33816	32,2262	30.8470	43,5445	30,8033
PSNR – between host & watermarked G	29,90552	27,4925	29.3177	38,3039	29,6218
PSNR – between host & watermarked B	26,78602	27,9785	31.2226	41,6322	32,4133
CC– between host & watermarked R	0,9942	0,9965	0.9958	0,9998	0,9958
CC – between host & watermarked G	0,9967	0,9954	0.9969	0,9995	0,9971
CC – between host & watermarked B	0,9783	0,9829	0.9921	0,9992	0,9939
PSNR – between original & retrieved watermark after attacks for channel R	53,2645	52,6211	52,47657	43,2719	53,0585

PSNR – between original & retrieved watermark after attacks for channel G	49,1751	50,3271	46,23263	38,291	45,98513
PSNR – between original & retrieved watermark after attacks for channel B	45,2931	46,7269	51,73257	40,9596	50,5457
CC – between original & retrieved watermark after attacks for channel R	0,9971	0,9945	0,9956	0,9443	0,9957
CC – between original & retrieved watermark after attacks for channel G	0,9979	0,9979	0,9972	0,98384	0,99703
CC – between original & retrieved watermark after attacks for channel B	0,9973	0,9977	0,99567	0,96994	0,99443
CC average - between original & retrieved watermarks after attacks for R, G, B average	0,9974	0,9967	0,99616	0,9660	0,99572
Fitness value for 3D optimizations	0,004907	0,005108	-	-	-
Fitness value – R for 1D optimization	-	-	0,00862	0,2726	0,0001
Fitness value – G for 1D optimization	-	-	0,00591	0,0129	0,0001
Fitness value – B for 1D optimization	-	-	0,01225	0,6726	-0,00053

VII. REFERENCES

[1] A. Akter, Nur-E-Tajnina and M. A. Ullah, "Digital image watermarking based on DWT-DCT: Evaluate for a new embedding algorithm," 2014 International Conference on Informatics, Electronics & Vision (ICIEV), Dhaka, 2014, pp. 1-6.

[2] Lee, C., Lee, H., "Geometric attack resistant watermarking in wavelet transform domain," in Optics Express vol. 13, no. 4, pp. 1307-1321 2005. Xiaowei Xu, S. Dexter and A. M. Eskicioglu, "A Hybrid Scheme for encryption and watermarking," Proceedings of the SPIE Security, Steganography, and Watermarking of Multimedia Contents VI Conf., San Jose, CA (c).

[3] K. Konstantinides, B. Natarajan, and G.S. Yovanof, "Noise Estimation and Filtering Using Block-Based Singular Value Decomposition," IEEE Trans. Image Processing, in press, 6 1997, pp.479- 483.

[4] V.I. Gorodetski, L.J. Popyack, V. Samoilov, and V.A. Skormin, "VD-Based Approach to Transparent Embedding Data into Digital Images," Proceedings International Workshop on Mathematical Methods, models and Architecture for Computer Network Security, Lecture Notes in Computer Science, 2052,

[5] D. V. S. Chandra, "Digital Image Watermarking Using Singular Value Decomposition," Proceeding of 45th IEEE Midwest Symposium on Circuits and Systems, Tulsa, OK, 2002, pp. 264-267

[6] R. Liu and T. Tan, "A SVD-Based Watermarking Scheme for Protecting Rightful Ownership," IEEE Transaction on Multimedia, April 2002, pp. 121-128.

[7] S.C. Byun, S.K. Lee, A. Tewfik, B.H. Ahn," A SVD Based Fragile Watermarking Scheme for Image Authentication," Digital Watermarking: First International Workshop, IWDW 2002, Seoul, Korea, Nov. 2002.

[8] K.L. Chung, C. Shen, L. Chang, "A novel SVD and VQ based image hiding scheme," Pattern Recognition Letters, 2002 pp.1051-1058.

[9] F. Liu, K. Han, C. Wang, "A novel blind watermark algorithm based On SVD and DCT," Proceedings of IEEE International Conference on Intelligent Computing and Intelligent Systems, (2009) 283-286

[10] X. Ma and X. Shen, "A novel blind grayscale watermark algorithm based on SVD," Proceedings of ICALIP, 2008 pp.1063-1068.

[11] M. Makhloghi, F.A. Tab, H. Danyali, "A new robust blind DWT-SVD based digital image watermarking," Proceedings of ICEE, 2011, pp.1-5.

[12] C.S. Shieh, H.C. Huang, F.H. Wang, J.S. Pan, Genetic watermarking based on transform domain techniques, Pattern Recognition, 2004, pp.555–565.

[13] Z. Wei, J. Dai, J. Li, Genetic watermarking based on DCT domain techniques, in Proc. of IEEE CCECE/CCGEI, 2006, pp.2365–2368.

[14] V. Aslantas, L. A. Dogan, and S. Ozturk, "DWT-SVD based image watermarking using particle swarm optimizer, " in Proc. IEEE Int. Conf. Multimedia Expo, Hannover, Germany, 2008, pp. 241-244.

[15] M. Monemizadeh and S. Seyedin, "Optimal dwt-svd domain image watermarking using multi-objective evolutionary algorithms," in WRI World Congress on Computer Science and Information Engineering(CSIE), vol. 3, 2009, pp. 259-263.

[16] H. Modaghegh, H. Khosravi, and M. Akbarzadeh, "A new adjustable blind Watermarking based on GA and SVD, " IEEE, 6th International Conference on Innovations in Information Technology, pp. 6-10, 2009.

[17] F. Golshan and K. Mohammadi, "A hybrid intelligent SVD-based digital image watermarking," Proc. Systems Engineering (ICSEng), pp.137-141, 2011.

[18] Golea NE-H, Melkemi KE, Melkemi M (2011) A novel multi-objective genetic algorithm optimization for blind RGB color image watermarking. In: Seventh international conference on signal-image technology and internet-based systems (SITIS), pp 306–313.

[19] V. Sivavenkateswara Rao ., Shekhawat Rajendra S. And Srivastava V. K. "A DWT-DCT-SVD Based Digital Image Watermarking Scheme Using Particle Swarm Optimization", 2012 IEEE Students' Conference on Electrical, Electronics and Computer Science

[20] I. A. Ansari, M. Pant and F. Neri, "Analysis of gray scale watermark in RGB host using SVD and PSO," 2014 IEEE Symposium on Computational Intelligence for Multimedia, Signal and Vision Processing (CIMSIVP), Orlando, FL, 2014, pp.1-7.

[21] V. S. V. Rao, R. S. Shekhawat and V. K. Srivastava, "A reliable digital image watermarking scheme based on SVD and particle swarm optimization," 2012 Students Conference on Engineering and Systems, Allahabad, Uttar Pradesh, 2012, pp. 1-6.

[22] S. Laha, J. Chowdhury, A. Khan and S. K. Sarkar, "A watermarking scheme based on Singular Value Decomposition and Particle Swarm Optimization," 2013 3rd IEEE International Advance Computing Conference (IACC), Ghaziabad, 2013, pp. 888-892.

[23] K. V. Durga, G. Mamatha and C. H. Bindu, "SVD based image watermarking with firefly algorithm," 2015 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, 2015, pp.1-7.

[24] K. Tamura and K. Yasuda, "Spiral optimization -A new multipoint search method," 2011 IEEE International Conference on Systems, Man, and Cybernetics, Anchorage, AK, 2011, pp. 1759-1764.

[25] K. Tamura and K. Yasuda, "Spiral Multipoint Search for Global Optimization," 2011 10th International Conference on Machine Learning and Applications and Workshops, Honolulu, HI, 2011, pp.470-475.

[26] J. Kennedy, and R. Eberhart. Proceedings of the IEEE International Conference on Neural Networks, page 1942--1948. (1995)

[27] CASIA Palm print Database, http://biometrics.idealtest.org/

[28] CASIA Iris Image Database, http://biometrics.idealtest.org/

[29] AMI Ear Database, http://www.ctim.es/research_works/ami_ear_database/