International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 04 Issue: 08 | Aug -2017 www.irjet.net p-ISSN: 2395-0072

Secret-Fragment Visible Mosaic Images by Genetic Algorithm

Chandan Ghonmode[1], Anoop kumar khambra[2]

[1]PG Student [digital communication], Dept. of ECE, Patel College of science and technology, Bhopal, India
[2]Assistant professor, Dept. of ECE, Patel College of science and technology, Bhopal, India

---

Abstract – *A new secure image transmission method is proposed in which it transforms the secret image into secret fragment-visible mosaic image. Mosaic image is created by dividing the secret image and target image into fragments of equal size and fitting these secret tile blocks into target blocks. For tile image hiding, a mapping sequence is generated using Genetic Algorithm (GA).*

Color transformations are performed to make the mosaic image similar to the target image. After color transformation rotation is performed. Rotating each tile block into an optimal rotation angle with minimum root mean square error value with respect to its corresponding target blocks. For the recovery of the secret image from the mosaic image embed relevant information into the created mosaic image. Overflows/underflows in the transformed color values can also be handled by using this method. By using the same key and the mapping sequence, the secret image can be recovered.

Key Words: Image hiding, Mosaic image, Genetic Algorithm, Key Based Random Permutation, Color Transformation.

I.INTRODUCTION

Images from different sources are utilized and transmitted through the internet for variety of applications. These applications include confidential enterprise archives, military image databases, document storage systems, online personal photograph albums and medical imaging systems. These images contain confidential or private information. Therefore, such information should be protected from leakages while transmitting through internet.

mosaic image is used to hide the secret image based on Genetic algorithm (GA) . Genetic algorithm (GA) is used to generate a mapping sequence for tile image hiding. Genetic algorithm (GA) provides better clarity in the retrieved secret image and reduces the computational complexity. The quality of original cover image remains same after embedding the secret image. So better security and robustness is assured.

The mosaic image is created by dividing the secret image and target image into equal number of fragments and fitting these secret tile blocks into corresponding target blocks according to the mapping sequence generated by Genetic Algorithm (GA).

Fig1: Generated Image

II.PROPOSED ALGORTHM

The proposed method includes two main phases, mosaic image creation and secret image recovery. In the first phase, mosaic image is created by fitting the secret tile locks to the corresponding target blocks. Mosaic image contains fragments of secret image with color corrections based on the similarity of the target image.

The mosaic image creation includes four stages:

- fitting the tile images of the secret image into the target blocks of a target image,
- transformation of the color characteristics of each tile blocks in the secret image to that of the corresponding target block in the target image,
- based on the minimum RMSE value, rotate each tile images into a direction with respect to corresponding target block, and
- embedding the relevant secret information into the mosaic image for the further recovery of the secret image.

In the second phase, the secret image is recovered from the secret image by extracting the secret information embedded in the mosaic image.
This phase includes two stages:

- extracting the embedded information from the mosaic image for secret image
  recovery and
- recovering the secret image using the extracted information.

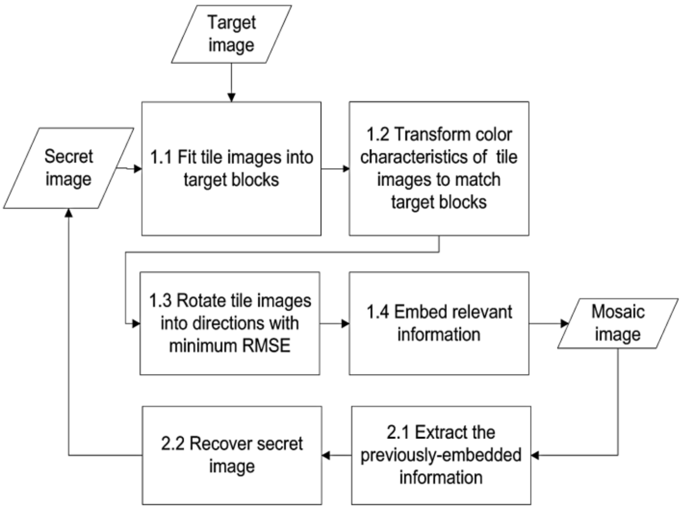


Fig 2: Flowchart of the proposed method

In this method, Genetic Algorithm (GA) is used for achieving additional security, robustness and also to improve the clarity of the image. Here, mosaic image is created by hiding the tiles of secret image into the arbitrarily selected target by using GA. In this method, GA

is used to generate a mapping sequence for fitting the tiles of the secret image into the target blocks.

Algorithm 1: Mosaic image creation

Input: a secret image S, a target image T, and a secret key K.

Output: a secret-fragment-visible mosaic image F.

Step 1: Divide the secret image S and target image T into n tiles.

Step 2: Compute the means and standard deviations of each tile image and each target block for the three color channels.

Step 3: Map the images using a mapping sequence L generated using Genetic Algorithm.

Step 4: Create a mosaic image F by fitting the tile images into the corresponding target blocks according to L.

Step 7: For each pixel in each tile image of mosaic image with color value is transform into new color values.

Step 8: Compute the RMSE values of each color transformed tile image with respect to its corresponding target block after rotating the tile image into each of the directions, 0o, 90o, 180o, and 270o and select the optimal direction with minimum RMSE value.

Algorithm 2: Creating the mapping sequence

Input: Target image, Secret image.

Output: Mapping sequence.

Step 1: Determine the population size and maximum generation size.

Step 2: [Start] Generate initial population of n chromosomes. A pixel in each block is the chromosome.

Step 3: [Fitness] Evaluate the PSNR value of each block and set as the fitness function f(x).

Step 4: [New Population] Create a new population by repeating the following steps until the new population is complete.

Step 5: [Selection] Select two parent chromosomes from a population according to their fitness.

Step 6: [Crossover] With a crossover probability, crossover the parents to form a new offspring. If no crossover is performed offspring's is the exact copy of the parents.

Step 7: [Mutation]With a mutation probability, mutate new offspring at each locus.

Step 8: [Accepting] Place new offspring in the new Population.

Step 9: [Replace] Use new generated population for a further run of the algorithm.

Step 10: [Test] If the end condition is satisfied, stop, and return the best solution in current population.

Step11: [Loop] Go to step 2

Algorithm 3: Secret image recovery

Input: a mosaic image F with n tile images and the secret key K.

Output: the secret image S.

Step 1: Extract the bit stream I from the mosaic image using the reverse RCM to obtain the number of iterations for embedding Mt' and the total number of pixel pairs used in the last iteration.

Step 2: Extract the bit stream Mt' using the values of Ni and N pair.

Step 3: Decrypt the bit stream Mt' into Mt by K.

Step 4: Decompose Mt into n bit streams.

Step 5: Decode Mi for each tile image to obtain the index of the target blocks, the optimal rotation angle, the means and standard deviation quotients and the overflow/underflow residual values.

Step 6: Recover one by one in a raster-scan order the tile images by the following steps:

1) Rotate the block in the reverse optimal angle,

2) recover the original pixel values using the extracted mean and standard deviation quotients,

3) compute CS and CL,

4) find out the pixels with values 255 or 0 which indicate overflow/underflow

5) add respectively the values CS and CL to the corresponding residual values of the found pixels and

6) take the results as the final pixel values, resulting in a final tile image.

Step 7: Compose all the final tile images to form the desired secret image S as output.

Simulation is done in Matlab 2013a.

Target image & Secret image.

Mosaic image created with tile image size 8×8

Recovered secret image

The Mean Square Error (MSE) and the Peak Signal to Noise Ratio (PSNR) are the two error metrics used to compare image compression quality. This ratio is often used as a quality measurement between the original and a mosaic image. If one of the signals is an original signal of acceptable (or perhaps pristine) quality, and the other is a distorted version of it whose quality is being evaluated, then the MSE may also be regarded as a measure of signal quality. MSE is a signal fidelity measure. The goal of a signal fidelity measure is to compare two signals by providing a quantitative score that describes the level of error/distortion between them. Usually, it is assumed that one of the signals is a pristine original, while the other is distorted or contaminated by errors.

Suppose that x = { xi |i = 1, 2, · · ·, N} and y = { yi |i = 1, 2, · · · , N} are two finite-length, discrete signals (e.g., visual images), where N is the number of signal samples (pixels, if the signals are images) and xi and yi are the values of the i th samples in x and y, respectively. The MSE between the signals x and y is

MSE (x,y) = 1/N ∑ (xi-yi)2

In the MSE, we will often refer to the error signal ei,= xi − yi, which is the difference between the original and distorted signals. If one of the signals is an original signal of acceptable (or perhaps pristine) quality, and the other is a distorted version of it whose quality is being evaluated, then the MSE may also be regarded as a measure of signal quality

MSE is often converted into a peak-to-peak signal-to-noise ratio (PSNR) measure.

PSNR= 10 log10 L/MSE

Where L is the dynamic range of allowable image pixel intensities. For example, for images that have allocations of 8 bits/pixel of gray-scale, L = 28− 1 = 255. The PSNR is useful if images having different dynamic ranges are being compared, but otherwise contains no new information relative to the MSE.

PSNR values without using genetic algorithm	**PSNR values using genetic algorithm**
19.2918	33.5722
18.2166	30.4578
16.5648	18.4678
18.4678	34.4163
17.3559	31.7823
19.4828	34.6861

Table 1: Comparison of PSNR values

The table values show a better result for the proposed method. High PSNR value indicates that the recovered secret image is nearly similar to the original secret image. Thus with the help of genetic algorithm an optimal tile-block fitting can be obtained which results in better clarity of the recovered secret image. The PSNR and the RMSE values of the mosaic image is checked and it is above 30 and RMSE is approximately equal to one indicates that the created mosaic image is similar to the target image.

The performance of this method is improved by using color transformations. Color transformation makes the mosaic image color values similar to the target image. Performance of the mosaic image creation is analyzed using three parameters: Peak-Signal Noise Ratio (PSNR), Root Mean Square Error (RMSE) and numbers of required bits embedded for recovering secret images.

The pixel to pixel relation between the parts is found to get the most similar parts of the secret image and the cover image. It is calculated using the R.M.S.E values of the parts of the images. The average values of the red, green and blue component of each part in the secret image and the cover image is calculated. Their differences give the R.M.S.E values of the

The pixel to pixel relation between the parts is found to get the most similar parts of the secret image and the cover image. It is calculated using the R.M.S.E values of the parts of the images. The average values of the red, green and blue component of each part in the secret image and the cover image is calculated. Their differences give the R.M.S.E values of the

corresponding parts. Using these methods, the RMSE values of all the parts of the images are calculated and all these values are compared with each other to find out the minimum value. These minimum valued part of the cover image is selected to place the corresponding part of the secret image.

The pixel to pixel relation between the parts is found to get the most similar parts of the secret image and the cover image. It is calculated using the R.M.S.E values of the parts of the images. The average values of the red, green and blue component of each part in the secret image and the cover image is calculated. Their differences give the R.M.S.E values of the corresponding parts. Using these methods, the RMSE values of all the parts of the images are calculated and all these values are compared with each other to find out the minimum value. These minimum valued part of the cover image is selected to place the corresponding part of the secret image.

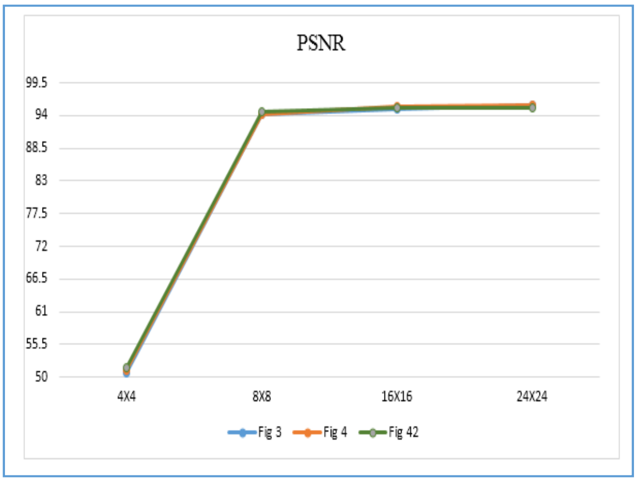
III. RESULT & DISCUSSION

PSNR is the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. PSNR is an approximation to human perception of reconstruction quality. A higher PSNR generally indicates that the reconstruction is of higher quality. The PSNR value of the mosaic image with respect to the target image and the decrypted secret image with respect to the original secret image are checked and it is above 30 which indicates that the mosaic image is look similar to the target image and the decrypted secret image is similar to the original secret image.

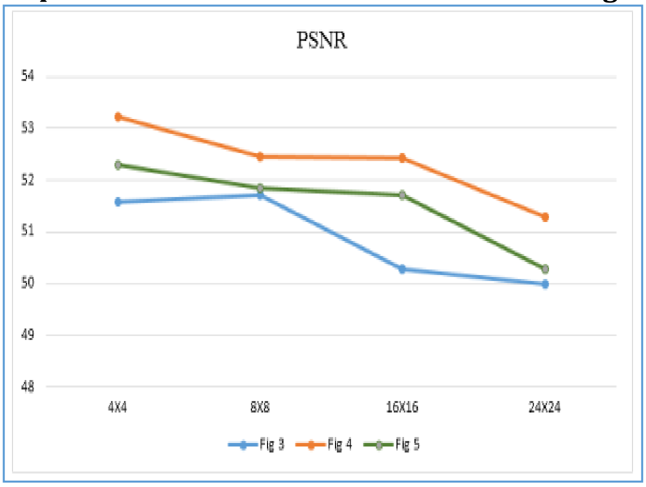
Shows the experimental result of mosaic image creation with tile image size 4×4.

Shows the plots of trends of various parameters versus different tile image sizes. PSNR values of created mosaic images with respect to target images, PSNR values of recovered secret images with respect to original ones, RMSE values of created mosaic images with respect to target images, RMSE values of recovered secret images with respect to original ones and the numbers of required bits embedded for recovering secret images are plotted.

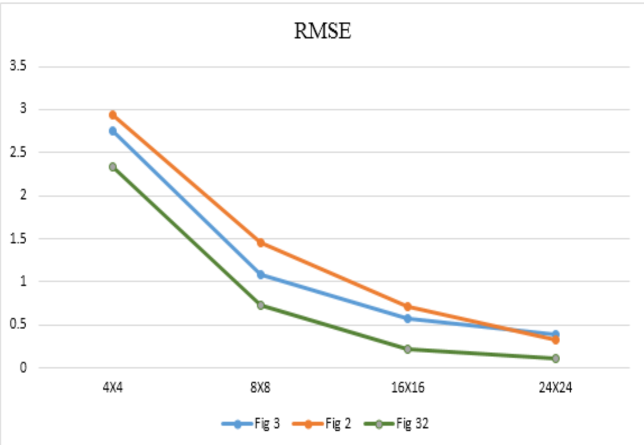
a) **PSNR values of created mosaic images with respect to target Images:**



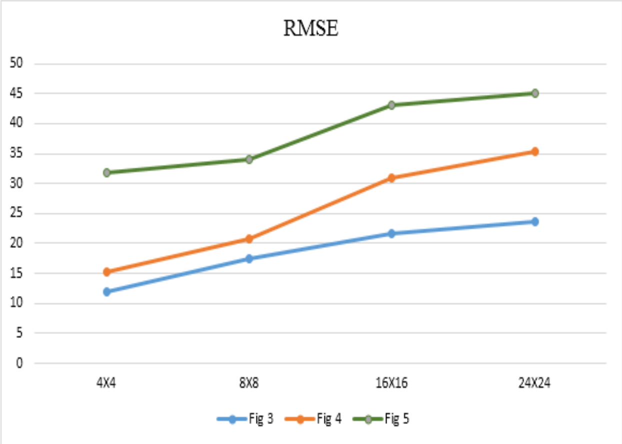
b) PSNR values of recovered secret images with respect tooriginal



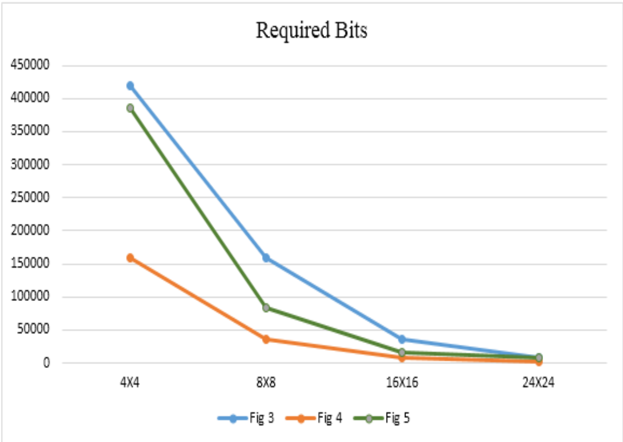
(c) RMSE values of created mosaic images with respect to target images:



(d)RMSE values of recovered secret images with respect to original image



(e) Numbers of required bits embedded for recovering secret images:



IV. TABULAR REPRESENTATION

The Root Mean Square Error (RMSE) is a used to compute the error between recover image and original image. The different sizes used for analysis are 4X4,8X8, 16X16 and 24X24. A comparison with various tile image sizes was done and checks the PSNR, RMSE values.

The PSNR and the RMSE values of the mosaic image is checked and it is above 30 and RMSE is approximately equal to 1 indicates that the created mosaic image is similar to the target image.

PSNR values without using genetic algorithm	PSNR values using genetic algorithm	RMSE values without using genetic algorithm	RMSE values using genetic algorithm	Different tile image sizes
53.2	52	12	0.3	4x4
52.5	94	17	0.7	8x8
52.4	95	21	0.3	16x16
51.2	97	24	0.2	24x24

V. CONCLUSION

The design of the NSCT is reduced to the design of a Images from different sources are utilized and transmitted through the internet for variety of applications. These applications include confidential enterprise archives, military image databases, document storage systems, online personal photograph albums and medical imaging systems. These images contain confidential or private information. Therefore, such information should be protected from leakages while transmitting through internet. Now a days, different methods have been proposed for secure image transmission.

REFERENCES

[1] J. Fridrich, "Symmetric ciphers based on two-dimensional chaotic maps," Int. J. Bifurcat. Chaos, vol. 8, no. 6, pp. 1259–1284, 1998.

[2] G. Chen,Y. Mao, and C. K. Chui, "A symmetric image encryption scheme based on 3D chaotic cat maps," Chaos Solit. Fract., vol. 21, no. 3, pp. 749–761, 2004.

[3] L. H. Zhang, X. F. Liao, and X. B. Wang, "An image encryption approach based on chaotic maps," Chaos Solit. Fract., vol. 24, no. 3, pp. 759–765, 2005.

[4] H. S. Kwok and W. K. S. Tang, "A fast image encryption system based on chaotic maps with finite precision representation," Chaos Solit. Fract., vol. 32, no. 4, pp. 1518–1529, 2007.

[5] S. Behnia, A. Akhshani, H. Mahmodi, and A. Akhavan, "A novel algorithm for image encryption based on mixture of chaotic maps," Chaos Solit. Fract., vol. 35, no. 2, pp. 408–419, 2008.

[6] D. Xiao, X. Liao, and P. Wei, "Analysis and improvement of a chaosbased image encryption algorithm," Chaos Solit. Fract., vol. 40, no. 5, pp. 2191–2199, 2009.

[7] V. Patidar, N. K. Pareek, G. Purohit, and K. K. Sud, "A robust and secure chaotic standard map based pseudorandom permutationsubstitution scheme for

image encryption," Opt. Commun., vol. 284, no. 19, pp. 4331–4339, 2011.

[8] C. K. Chan and L. M. Cheng, "Hiding data in images by simple LSB substitution," Pattern Recognit.., vol. 37, pp. 469–474, Mar. 2004.

[9] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 3, pp. 354–362, Mar. 2006.

[10] J. Tian, "Reversible data embedding using a difference expansion," IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 8, pp. 890–896, Aug. 2003.

[11] Y. Hu, H.-K. Lee, K. Chen, and J. Li, "Difference expansion based reversible data hiding using two embedding directions," IEEE Trans. Multimedia, vol. 10, no. 8, pp. 1500–1512, Dec. 2008.

[12] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y.-Q. Shi, "Reversible watermarking algorithm using sorting and prediction," IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 7, pp. 989–999, Jul. 2009.

[13] X. Li, B. Yang, and T. Zeng, "Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection," IEEE Trans. Image Process., vol. 20, no. 12, pp. 3524–3533, Dec. 2011.

[14] W. Zhang, X. Hu, X. Li, and N. Yu, "Recursive histogram modification: Establishing equivalency between reversible data hiding and lossless data compression," IEEE Trans. Image Process., vol. 22, no. 7, pp. 2775–2785, Jul. 2013.