Middle-East Journal of Scientific Research 24 (S1): 212-219, 2016
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2016.24.S1.42

Secure Image Transmission by Lossless Secret Fragment Mosaic Image Creation in Videos

[1]J.S.M. Melvina, [1]Mandava Nikhita Santhoshi and [2]S. Malathi

[1]Third Year, Department of CSE, Panimalar Engineering College, Chennai, India
[2]Professor, Department of CSE, Panimalar Engineering College, Chennai, India

Abstract: A multitude of digital images preserve important data and so providing authentication to these images is often a challenging task. Although a number of image encryption techniques have been proposed, there are no encryption techniques capable of providing a noiseless mosaic image that does not require a large database. Thus, we propose a novel technique that transforms a secret image into a meaningful mosaic image of the same size that looks exactly like the preselected target image. The transformation process is handled by a secret key and only with that key a person can recover the secret image in alossless manner from the mosaic image. Hence,the secret fragment mosaic image creation technique is aimed to design a novel method that can, not only transform a secret image into a secret fragment-visible mosaic image of the same size, but also has the visual appearance of any freely selected target image, without the need of an actual database nor any compression techniques. Therefore, thisaids in preserving the crucial image data in a secure and safe manner.

Key words: Video processing · Database · Mosaic image · Secret image · Target image

INTRODUCTION

In today's world of technological convergence, securing the information is the ultimate concern. Providing only network security is not enough with the drastic increase in cybercrime. Security offered to images of company projects, includes securing images such as blueprints, secret images of concern to the army or of company's interest, using image steganography and stitching. This is beneficial to the company. Moreover, since the secret image is broken down into parts and then sent into the receiver, it becomes difficult for the trespassers to get access to all the parts of the images at once. Thus, when we enhance the security to a higher level, it makes it difficult for the intruder to detect and decode the document. A variety of methods have been proposed for securing image transmission of which two common approaches are image encryption and data hiding. Image encryption is a technique that exploits the natural property of an image such as, high redundancy and strong spatial correlation, in an attempt to get an encrypted image. The encrypted image is often a noise image where nobody can obtain the secret image from it unless the person has the correct key. This encrypted image is a meaningless file, which cannot provide any additional information before decryption and often arouses the attacker's attention during transmission, due to its randomness in form.An alternative to avoid the problem of distortion or randomness is by adopting data hiding technique that hides a secret message into a cover image in such a way, that no one can realize the existence of the secret data. A major concern in the data hiding method in images is the difficulty in embedding a large amount of message data into a single image. However, if one wants to hide it into a cover image, a secret image of the same size, then, the secret image must be highly compressed in advance.

For instance, in a data hiding method that has an embedding rate of 0.5 bits per pixel, a secret image with 8 bits per pixel must be compressed at a rate of at least 93.75%, in advance, in order to hide the secret image into a cover image. All the existing data hiding methods mostly utilize the techniques of LSB substitution, histogram shifting, difference expansion, prediction-error expansion, recursive histogram modification and discrete cosine/wavelet transformations.

Corresponding Author: J.S.M. Melvina, Third Year, Department of CSE, Panimalar Engineering College, Chennai, India.

The proposed method transforms a secret image into a meaningful mosaic image of same size and one that looks exactly like the preselected target image. The mosaic image is an assemblage of the fragments of a secret image in disguise of another image called the target image that is preselected from a database. The transformation process is handled by a secret key and only with that key a person can recover the secret image in a lossless manner from the mosaic image. After a target image is selected, the given secret image is divided into rectangular fragments, called the tile images. These tile images are fit into similar blocks in the target image, called target blocks, according to a similarity criteria based on color variations. The color characteristics of each tile image corresponds to that of the corresponding target block in the target image, resulting in a mosaic image that looks like the target image. Relevant methodologies have been proposed to conduct nearly lossless recovery of the original secret image from the resultant mosaic image. Hence, the main objective is to design a novel method that can transform a secret image into a secret fragment-visible mosaic image of same size, which also has the visual appearance of any freely selected target image, without the actual need of a database or image compression.

Related Work: J. Fridrich proposed a symmetric cipher technique based on two-dimensional chaotic maps [1] in the year 1998. A chaotic map is an evolution function, which exhibits some sort of chaotic behavior. An extension to this technique was proposed by G. Chen *et al.,* which extended it to three-dimensional chaotic maps [2] which were more efficient. H.S. Kwok *et al.,* proposed an image encryption system which also used chaotic maps using a finite precision representation [4]. A novel image encryption algorithm based on a mixture of chaotic maps [5] was proposed by S. Behnia *et al.,* in the year 2008. The mixture application of chaotic maps shows advantages of large key space and high level security.

V. Patidar, N.K. Pareek, G. Purohit and K.K. Sud, in the year 2009, proposed a technique for image encryption using a standard chaotic map based on pseudorandom permutation substitution scheme [7]. A simple LSB substitution scheme was used to provide image encryption [8]. This technique was proposed by C. K. Chan *et al.,* in the year 2004. J. Tian, in the year 2003, proposed an image encryption technique which was implemented by reversible data embedding using a difference expansion [10]. An efficient data hiding technique by using difference expansion of two embedding directions [11] was proposed by Y. Hu *et al.,* in the year 2008. V. Sachnev *et al.,* introduced a reversible watermarking algorithm using sorting and prediction [12] in the year 2009. A refinement to the reversible watermarking technique was proposed by X. Li *et al.* this was based on adaptive predictive error expansion and pixel expansion [13].

A reversible histogram modification by establishing equivalency between reversible data hiding and lossless data compression [14] was proposed by X. Li *et al.,* in the year 2013. C.C. Chang *et al.,* proposed a reversible data hiding technique which required the use of DCT based compression images [16]. A reversible watermarking technique was proposed by S. Lee *et al.,* which was based on integer to integer wavelet transformation [17]. The control of distortions introduced by the watermarking was further studied. An efficient watermarking technique based on wavelet coefficient quantization [18] was proposed by W.H. Lin *et al.,* in the year 2008. X. Hu *et al.,* proposed a technique to estimate the signal distribution [19] in the year 2013. This estimation aided the data hiding of images.

W.B. Pennebaker *et al*., have presented a technique using chaotic schemes for data hiding [20] in 1993. These techniques also provide security functions for visual check, which might be applicable in some applications. Another technique for information hiding was presented by I-Jen Lai and Wen-Hsiang Tsai [21]. A secret key is adopted that randomly selects some blocks of mosaic images to embed the tile image. This makes harder for an attacker to retrieve information without knowing the secret key.

E. Reinhard [22] *et al*., in the year 2001, proposed a method which adopts color transferring techniques to provide encryption to images. D.L. Ruderman, T.W. Cronin and C.C. Chiao [23] proposed a method for visual coding by taking the statistics of the cone responses to natural images. D. Coltuc *et al.* [24] proposed a rapid watermarking technique which uses reversible contrast mapping in the year 2007. An optimal LSB substitution for image hiding was proposed by R.Z. Wang *et al*., [25] in the year 2001.

Proposed Work: The proposed method includes two main phases namely the mosaic image creation phase and the secret image recovery phase.

Figure 1 depicts a flow chart that describes the working of the two phases. In the first phase, which is the mosaic image creation phase, a mosaic image is produced that consists of the fragments of an input secret image

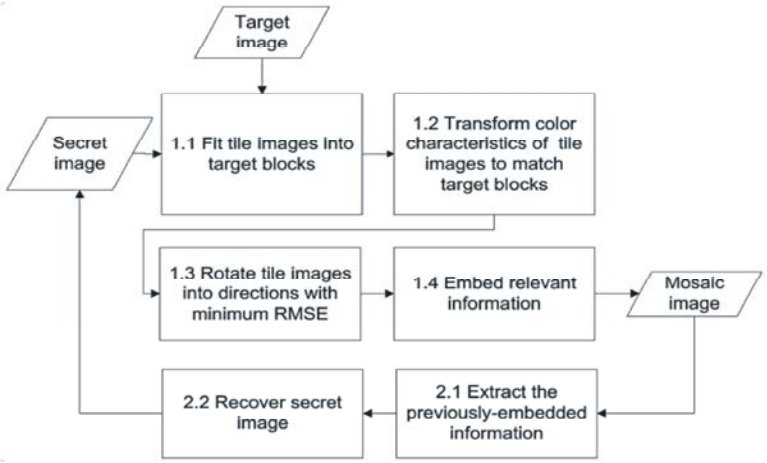

Fig. 1: Flow diagram of the proposed method

with color corrections according to a criteria based on color variations. It includes four stages. The first stage places the tile images of the secret image into the target blocks of a preselected target image. After this, the color characteristics of each of the tile images in the secret image are altered to become the corresponding target block in the target image. Then, we rotate each tile image in a direction that has minimum RMSE value with respect to the corresponding target block. Finally we embed the relevant information into the mosaic image being created, for recovering the secret image. The second phase involves extracting the embedded informationin order to recover the nearly lossless secret image from the generated mosaic image. This phase includes two important stages, Inthe first stage, we extract the embedded information from the mosaic image for recovering the secret image and in the second stage we retrieve the secret image using the extracted information. Thus, by using this technique, only the receiver who has the key can decode the secret image. Even though an eavesdropper who does not have the key might try all possible permutations of the tile images in the mosaic image to get the secret image back, the number of all possible permutations here is $n!$ And so the probability to correctly guess the permutation is $p=1/n!$, this is very small in value.

Creating Mosaic Image: Figure 2 clearly describes the proposed system of our secure image transmission system. Initially the user logs himself as the administrator

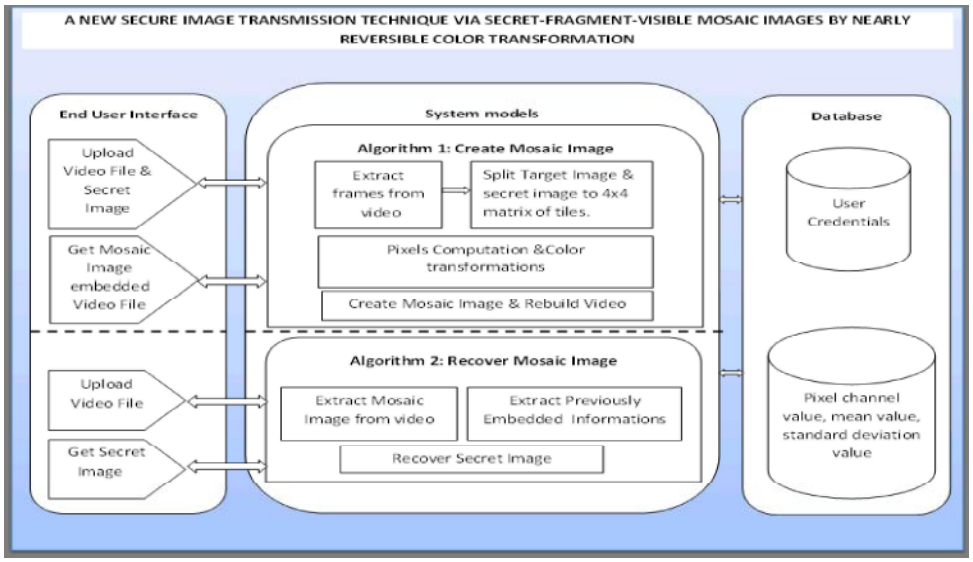

Fig. 2: Framework of the proposed system

and selects a video and uploads it. The video is then decompressed into frames in 800*800 pixels format. Once all the frames are obtained from the video (10f per second), they are saved in a database for future reference while selecting the target image. Also the user is required to upload any one secret image depending upon their preferences. Then the color transformation and the computations are performed with the help of the following algorithm.

Algorithm for Mosaic image creation (A, B, C)

Input: a secret image A, a target image B and a Secret key C.
//Fitting the tile images into the target blocks.
Step 1:
if (size of B != size of A){
convert size of B==size of A;
divide A into n tiles{T1, T2,.. ., Tn};
divide B into n blocks {B1, B2,.. ., Bn};
}
Step 2:for(i=1;i<=n;i++){
for(j=1;j<=n;j++) {
Compute mean, SD and avg SD of Ti and Bj;}
}
Step 3:Sort Stile ={T1, T2,.. .,Tn};
Sort Btarget ={B1, B2,.. ., Bn};
Step 4: Create M by fitting blocks according to L.
// performing color conversions
Step 5:for each ci where c=r, g,or b{
transform ci into a new;
if (ci !< 255 ||ci 0){
ci == 255 or 0;
compute residual value Ri for pi ;
Count++;}
}
// rotating the tile images.
Step 6:Rotate Ti to θ with smallest RMSE value;
Compute RMSE value of each Ti in M
// Embedding
Step 7: Construct a Huffman table HT
Step 8: For each Ti in M, construct a bit stream Mi
Step 9: Concatenate Mi of all Ti in F into Mt;
Step 10: encrypt Mt usingsecret key K

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

Here, the user selects a secret image and target image then splits it into 4x4 matrices of Tile images. The entire tiles' pixels and features are computed for further color transformations. If the size of the target image B is different from that of the secret image A, the size of B is subjected to alteration in order to be identical to that of A; The secret image A is divided into n tile images {T1, T2,.. . , Tn} and the target image B into n target blocks {B1, B2,.. . , Bn} provided each Ti or Bi should be of size NT. The means and standard deviations for each tile images Ti and each target block Bj is computed for the three color channels. Then the average standard deviation is computed for Ti and Bj for n iterations of i and j. The tile images are then subjected to sorting and stored as Stile ={T1,T2,...,Tn}; similarly the target blocks are sorted and stored as Btarget ={B1,B2,…,Bn}; based on the computed average standard deviations of the block. The sorted tiles and the blocks are mapped in a one to one manner and reordered based on the indices of the tiles, resulting in a mapping pattern(sequence) L of the form: T1 → Bj1, T2 → Bj2,..., Tn→ Bjn.

A mosaic image M is thus created by fitting the tile images into the corresponding target blocks according to the sequence L. After this the color conversions between the tile images and the target blocks are achieved by creating a counting table TB of 256 records, each with an index corresponding to a residual data value. An initial value of zero is assigned to each entry(note that each residual value will be in the range of 0 to 255). For each mapping Ti →Bji in sequence L, the means μc and μ _ c of Ti and Bji are represented, respectively, by eight bits; their standard deviation quotient qc appearing in (3) is represented by seven bits, where c = r, g, or b.

For every pixel pi in every tile Ti of mosaic image M with color value ci where c = r, g, or b, ci is transformed into a new value ci by (3); if ci is not smaller than 255 or if it is not larger than 0, then ci is changed to be 255 or 0, respectively; A residual value Ri for pixel pi is computed; the count in the table TB is incremented by 1 for those indexes which are identical to Ri. For rotating the tile images, the RMSE values of each color transformed tile image Ti is computed in M with respect to its corresponding target block Bji after rotating Ti into each of the directions θ =0, 90, 180 and 270; Ti is then rotated into the optimal direction θ with the smallest RMSE value.

In order to complete the creation process of the mosaic image along with the rebuilding of the video, a Huffman table HT is constructed using the content of the counting table TB to encode all the residual values computed previously. For each tile image Ti in mosaic image M, a bit stream Mi is created for recovering Tn, including the bit-segments which encode the data items of the index of the corresponding target block Bji, the optimal rotation angle θ° of Ti, the means of Ti and Bji

and the related standard deviation quotients of all three color channels and the bit sequence for overflows / underflows with residuals in Ti encoded by the Huffman table HT constructed.

The bit streams Mi of all Ti in M is concatenated in a raster-scan order to form a total bit stream Mt. The secret key K is used to encrypt Mt into another bit stream Mt; Mt is then embedded into M by the reversible contrast mapping scheme proposed.

A bit stream I is then created including the number of conducted iterations Ni for embedding Mt, the number of pixel pairs Npair used in the last iteration and the Huffman table HT constructed for the residuals. This bit stream I is then embedded into the mosaic image M. Thus after computing all the operations the mosaic image is inserted into all the target frames which are later merged into a video in *.avi format for uncompressing images.

Retrieving the Secret Image: The second phase of the system is the retrieval of the secret image from the video. So far the image has been encrypted and has been included along with the video.

Algorithm for Secret image recovery (M,K)

Input: a mosaic image M with n tile images
{T1, T2,...,Tn} and the secret key K
// Extracting the secret image recovery information.
Step 1: Extract from M the bit stream I
Step 2:Decrypt the bit stream Mt into Mt byK.
Step 3: Decompose Mt into n bit streamsM1 through Mn for the n to-be-constructed tile images T1 through Tn in A
Step 4: Decode Mi for each tile image Ti to obtain the encrypted data items:
// Recovering the secret image.
Step 5: Recover one by one in a raster-scan order the tile images Ti, i= 1 through n,
Step 6: Compose all the final tile images to form the desired secret image A as output.

Output: the secret image of A

The user on the receiving end uploads the video which contains the mosaic image, extracts all the frames from the video files and recovers all secret image tiles from mosaic image. Initially, the user extracts from M, the bit stream I by a reverse version of the scheme proposed in [24] and decodes them to obtain the number of iterations Ni for embedding Mt, the total number of used pixel pairs N and the Huffman table HT for encoding the values of the residual overflows or underflows. The bit stream Mt is then extracted using the values Ni and Npair by the same scheme used in the previous step. The bit stream Mt is decrypted into Mt by K. Mt is then decomposed into n bit streams M1 through Mn for the n to-be-constructed tile images T1 through Tn in A, respectively. Mi is then decoded for each tile image Ti in order to obtain the index ji of the block Bji in M corresponding to Ti, the optimal rotation angle $\theta°$ of Ti, the means of Ti and Bji and the related standard deviation quotients of all color channels and the overflow/underflow residual values in Ti decoded by the Huffman table HT.

The recovery stage involves recovering one by one, in a raster-scan order, the tile images Ti, i = 1 through n, of the desired secret image A by rotating in the reverse direction the block indexed by ji, namely Bji, in M through the optimal angle $\theta°$ and fitting the resulting block content into Ti to form an initial tile image Ti. We use the extracted means and related standard deviation quotients to recover the original pixel values in Ti. The extracted means and standard deviation quotients is used to compute the two parameters cS and cL. Ti is then scanned to find out pixels with values 255 or 0 which indicate that overflows or underflows, respectively, have occurred there. The values cS or cL is added respectively to the corresponding residual values of the found pixels and finally the results are taken as the final pixel values, resulting in a final tile image Ti. The final tile images are then composed to form the desired secret image A as output.

RESULTS AND DISCUSSIONS

In order to increase the security of the proposed method, the embedded information for later recovery is encrypted with a secret key as seen in Algorithm 1. Only the receiver who has the key can decode the secret image. However, an eavesdropperwho does not have the key may still try all possible permutations of the tile images in the mosaic image to get the secret image back. Fortunately, the number of all possible permutations here is $n!$ and so the probability for the attacker to correctly guess the permutation is $p=1/n!$ this is very small in value.

For example, for the typical case in which we divide a secret image of size 1024×768 into tile images with block size 8×8, the value n is (1024×768)/(8×8) = 12,288. So the probability to guess the permutation correctly without the key is $1/n! = 1/(12{,}288!)$. So breaking the system by this way of guessing is computationally infeasible. In fact, we can view the addressed problem here as a square jigsaw puzzle problem, which is to reconstruct a complete image from a set of unordered square puzzle parts.

Recently, many methods have been proposed to try to solve this problem automatically by utilizing measures of feature-based similarity, dissimilarity-based compatibility, prediction-based compatibility and so on. But these state-of-art methods can only solve partially problems with limited numbers of puzzle parts automatically. Also, the jigsaw puzzle problem has been proved to be NP-complete, which means that we cannot solve the problem in polynomial time. In fact, the time complexity is as mentioned in, which is too big a number as well for our case here with n = 12,288. However, when n is much smaller, say smaller than 1000, some compatibility metrics may be utilized to solve the square

So, a large value of n should be used to increase the security of the proposed method. In addition, the addressed puzzle problem of the proposed method is more complicated than the conventional square jigsaw puzzle problem because the color characteristics of the puzzle parts have been changed, that is, adjacent puzzle parts have different color appearances, meaning that a greedy search using color similarities between originally adjacent fragments for image reconstruction as done in conventional manual reconstruction techniques is infeasible, either.

Furthermore, even if one happens to guess the permutation correctly, one still will not be aware of the correct parameters for recovering the original color appearance of the secret image because such parameter information for color recovery is encrypted as a bit stream using a secret key. Even so, it still should be assumed, in the extreme case, that the attacker will observe the content of the mosaic image with a correct permutation and try to figure useful information out of it.

For example, an attacker might analyze the spatial continuity of the mosaic image in order to estimate a rough version of the secret image. To increase the security of the proposed method against this type of attack, one possible way to is to use the key to randomize important information of a secret image, such as the positions of the pixels in the secret image, before transforming the secret image into a mosaic image by the proposed method. Consequently, only authorized users with the key will be able to access the correct secret image.

Also, the input for the system, video file should be of the following two- namely. mkv file or. avi file. This is because the Dot Net Framework used, supports the files of above formats only. Similarly, the image that has to be transmitted securely, otherwise called as secret image should be of format. jpg or. png. The output of the proposed system will also be of the above mentioned formats. Precisely, the system can handle only video files of. mkv and. avi formats and image files of. jpg and. png formats.

Table 1 Different Encryption Techniques

	Comparison Study		
S.No.	Technique used	Number of attacks	Time taken
1.	Data Hiding	4.3	1.4
2.	Image Encryption	4.1	1.7
3.	SFVM	2.8	2.6
4.	SFVM in videos	1.7	4.2

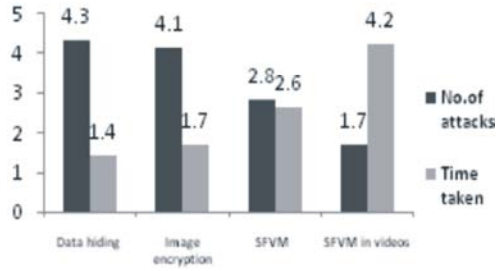

Fig. 3: Secure transmission over a medium

Table 1 depicts the comparison study that has been undertaken on the various encryption techniques and their efficiencies. We observe that the data hiding technique has experienced the most number of attacks with minimum time when compared with the remaining techniques. SFVM in videos seem to establish itself as the best technique among the four.

Figure 3 depicts the inversely proportional relationship between the number of attacks made and the time taken to break the system of various secure transmission techniques. These techniques range from the conventional data hiding and image encryption methods to the new techniques of Secret fragment visible mosaic image creation during transmission and the proposed Secret fragment visible mosaic image creation in videos for enhanced security.

CONCLUSION

In this, we proposed the Secure Image Transmission by Lossless Secret Fragment Mosaic Image Creation in Videos, a new method for maximizing the security of images from attacks and leaks while transmitting over a medium. It can not only create meaningful mosaic images but also can transform a secret image into a mosaic of same data size to be used as a camouflage of the secret image. With proper pixel color transformations and skillful

schemes for handling overflow, underflow conditions in the conversion values of the pixel colors, secret-fragment visible mosaic images with very close similarities to arbitrarily-considered target images can be created with no need of a large image database. Moreover there is lossless recovery of the original secret images from the created mosaic mages. Experimental results have shown great feasibility of the proposed method. The Secret-fragment visible mosaic imaging in video technology may establish image security suggestions in mobile application development. We plan to develop more cost-based models for cost-based image security. We want to study the possibility of transmission of videos of various other formats with same technique we postulated and proposed. Future studies may be one for applying the proposed method to other color models excluding RGB.

REFERENCES

1. Fridrich, J., 1998. "Symmetric ciphers based on two-dimensional chaotic maps," Int. J. Bifurcat. Chaos, 8(6): 1259-1284.
2. Chen, G., Y. Mao and C.K. Chui, 2004. "A symmetric image encryption scheme based on 3D chaotic cat maps," Chaos Solit. Fract., 21(3): 749-761.
3. Zhang, L.H., X.F. Liao and X.B. Wang, 2005. "An image encryption approach based on chaotic maps," Chaos Solit. Fract, 24(3): 759-765.
4. Kwok, H.S. and W.K.S. Tang, 2007. "A fast image encryption system based on chaotic maps with finite precision representation," Chaos Solit. Fract, 32(4): 1518-1529.
5. Behnia, S., A. Akhshani, H. Mahmodi and A. Akhavan, 2008. "A novel algorithm for image encryption based on mixture of chaotic maps," Chaos Solit. Fract, 35(2): 408-419.
6. Xiao, D., X. Liao and P. Wei, 2009. "Analysis and improvement of a chaos based image encryption algorithm," Chaos Solit. Fract, 40(5): 2191-2199.
7. Patidar, V., N.K. Pareek, G. Purohit and K.K. Sud, 2011. "A robust and secure chaotic standard map based pseudorandom permutation substitution scheme for image encryption," Opt. Commun., 284(19): 4331-4339.
8. Chan, C.K. and L.M. Cheng, 2004. "Hiding data in images by simple LSB substitution," Pattern Recognit., 37: 469-474, Mar. 2004.
9. Ni, Z., Y.Q. Shi, N. Ansari and W. Su, 2006. "Reversible data hiding," IEEE Trans. Circuits Syst. Video Technol., 16(3): 354-362, Mar. 2006.
10. Tian, J., 2003. "Reversible data embedding using a difference expansion," IEEE Trans. Circuits Syst. Video Technol., 13(8): 890-896, Aug. 2003.
11. Hu, Y., H.K. Lee, K. Chen and J. Li, 2008. "Difference expansion based reversible data hiding using two embedding directions," IEEE Trans. Multimedia, 10(8): 1500-1512, Dec. 2008.
12. Sachnev, V., H.J. Kim, J. Nam, S. Suresh and Y.Q. Shi, 2009. "Reversible watermarking algorithm using sorting and prediction," IEEE Trans.Circuits Syst. Video Technol., 19(7): 989-999, Jul. 2009.
13. Li, X., B. Yang and T. Zeng, 2011. "Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection," IEEE Trans. Image Process., 20(12): 3524-3533, Dec. 2011.
14. Zhang, W., X. Hu, X. Li and N. Yu, 2013. "Recursive histogram modification: Establishing equivalency between reversible data hiding and lossless data compression," IEEE Trans. Image Process., 22(7): 2775-2785, Jul. 2013.
15. Fridrich, J., M. Goljan and R. Du, 2001. "Invertible authentication," Proc. SPIE, 3971: 197-208.
16. Chang, C.C., C.C. Lin, C.S. Tseng and W.L. Tai, 2007. "Reversible hiding in DCT-based compressed images," Inf. Sci., 177(13): 2768-2786.
17. Lee, S., C.D. Yoo and T. Kalker, 2007. "Reversible image watermarking based on integer-to-integer wavelet transform," IEEE Trans. Inf. Forens. Security., 2(3): 321-330, Sep. 2007.
18. Lin, W.H., S.J. Horng, T.W. Kao, P. Fan, C.L. Lee and Y. Pan, 2008. "An efficient watermarking method based on significant difference of wavelet coefficient quantization," IEEE Trans. Multimedia, 10(5): 746-757, Aug. 2008.
19. Hu, X., W. Zhang, X. Hu, N. Yu, X. Zhao and F. Li, 2013. "Fast estimation of optimal marked-signal distribution for reversible data hiding," IEEETrans. Inf. Forens. Secur., 8(5): 187-193, May 2013.
20. Pennebaker, W.B. and J.L. Mitchell, 1993. JPEG: Still Image Data Compression Standard. New York, NY, USA: Van Nostrand Reinhold, 1993, pp: 34-38.
21. Lai, I.J. and W.H. Tsai, 2011. "Secret-fragment-visible mosaic image-A newcomputer art and its application to information hiding," IEEE Trans. Inf. Forens. Secur., 6(3): 936-945, Sep. 2011.
22. Reinhard, E., M. Ashikhmin, B. Gooch and P. Shirley, 2001. "Color transfer between images," IEEE Comput.Graph. Appl., 21(5): 34-41, Sep.-Oct. 2001.

23. Ruderman, D.L., T.W. Cronin and C.C. Chiao, 1998. "Statistics of cone responses to natural images: Implications for visual coding," J. Opt. Soc. Amer., 15(8): 2036-2045.
24. Coltuc, D. and J.M. Chassery, 2007. "Very fast watermarking by reversible contrast mapping," IEEE Signal Process. Lett., 14(4): 255-258, Apr. 2007.
25. Wang, R.Z., C.F. Lin and J.C. Lin, 2001. "Image hiding by optimal LSBsubstitution and genetic algorithm," Pattern Recog., 34(3): 671–683.