Synthesized Texture Reversible Data Hiding in Image Cryptography

Siddavatam Siva Jyothi[1], Dr. A. Rama Mohan Reddy[2]

[1]Academic Consultant, Dept of CSE, YV University, Kadapa

[2]Professor, Dept of CSE, Krishna Teja Institutes,Tirupathi

Abstract Recently, different techniques are available for data hiding. When to send some confidential data over insecure channel it is mandatory to embed data in some host or cover media. While sending secure data using cover media it necessary to encrypt as well as compress the cover media after compression embed confidential data. For providing this facility there various encryption/decryption techniques, compression techniques, and data embedding techniques are available. It is also important the data embedding should be reversible in nature. Here we are discussing different data embedding techniques that are reversible in nature by using encrypted image as cover media. In separable reversible data hiding in encrypted image initially the content owner encrypts the original uncompressed image, then the data hider compress the image to create sparse space to accommodate some additional data. At the receiver end, receivers extract the embedded data and recover the cover image without any loss

Keywords: Separable Reversible Data Hiding, data hiding key, encryption key, Difference expansion.

1. INTRODUCTION

Steganography is the method of hiding a message, file, image, or video within another file, message, image, or video. The word steganography combines from the two Greek words "steganos" means "protected", and "grapheins" means "writing". The advantage of steganography than cryptography is that the secret message does not attract the attention of the attackers by simple observation. The cryptography protects only the content of the message, while steganography protects the both messages and communication environment. In most of the image steganographic methods, uses the existing image as their cover medium. This leads to two drawbacks. Since the size of the cover image is fixed, embedding a large secret message will results in the distortion of the image. Thus a compromise should be made between the size of the image and the embedding capacity to improve the quality of the cover image. The distortion of the image results in second drawback, because it is feasible that a steganalytic algorithm can defeat the image steganography and thus reveal that a hidden message is conveyed in a stego image. The paper will proposes a good approach for steganography using reversible texture synthesis based on edge adaptive and tree based parity check to improve the embedding capacity. A texture synthesis process is of creating a big digital image with a similar local appearance of the original image and has an arbitrary size. And the paper is also using another two methods named edge adaptive and tree based parity check to improve the embedding capacity. The paper fabricates the texture synthesis process into steganography concealing secret messages as well as the source texture. In particular, in contrast to using an existing cover image to hide messages, our algorithm conceals the source texture image and embeds the secret messages through the process of texture synthesis. This allows us to extract the secret messages and the source texture from a stego synthetic texture. The proposed approach offers three advantages. First, since the texture synthesis can synthesize an arbitrary size of texture images. Since the Human Visual System (HVS) is less sensitive to changes in sharp regions compared to smooth regions, edge adaptive methods has been proposed to find the edge regions and hence improve the quality of the stego image as well as improve the embedding capacity and TBPC to hide the secret data into the cover image. Secondly, a steganalytic algorithm is not to defeat the steganographic approach since the texture image is composed of a source texture rather than by changing the existing image contents. Third, the reversible capability used in the project results in the recovery of the source texture so that the same texture can be used for the second round of message redirect.

Most photograph steganographic algorithms adopt an existing picture as a cover medium. The cost of embedding secret messages into this duvet photograph is the photograph distortion encountered within the stego image. This results in two drawbacks. First, for the reason that the dimensions of the cover picture is fixed, the more secret messages which are embedded permit for more image distortion. Hence, a compromise have got to be reached between the embedding capacity and the image high-quality which outcome in the limited capacity supplied in any particular duvet image. Don't forget that image steganalysis is an strategy used to notice secret messages hidden in the stego picture. A stego image includes some distortion, and regardless of how minute it's, this will intrude with the common elements of the quilt photo. This leads to the 2nd trouble for the reason that it's still possible that an snapshot steganalytic algorithm can defeat the image steganography and therefore reveal that a hidden message is being conveyed in a stego image. In this paper, we propose a novel approach for steganography making use of reversible texture synthesis. A texture synthesis approach re-samples a small texture photo drawn via an artist or captured in a image in order to synthesize a new texture snapshot with a identical regional appearance and arbitrary measurement. We weave the texture synthesis process into steganography concealing secret messages As good considering the fact that the source texture. In designated, in big difference to utilizing an existing cover photo to cover messages, our algorithm conceals the supply texture snapshot and embeds secret messages through the system of texture synthesis. This allows for us to extract the key messages and the deliver texture from a stego synthetic texture. To the high-quality of our expertise, steganography taking advantage of the reversibility has ever been awarded throughout the literature of texture synthesis. Our procedure presents three benefits. First, for the reason that the feel synthesis can synthesize an arbitrary measurement of texture pictures, the embedding capability which our scheme offers is proportional to the scale of the stego texture picture. Texture synthesis has obtained numerous concentration lately in pc imaginative and prescient and laptop graphics [8]. Essentially the most recent work has concerned about texture synthesis by illustration, where a supply texture picture is re-sampled utilizing either pixel-based or patch-situated algorithms to supply a new synthesized texture photograph with similar neighborhood look and arbitrary measurement. Pixel-founded algorithms [9], [10], [11] generate the synthesized image pixel by pixel and use spatial neighborhood comparisons to prefer essentially the most an identical pixel in a sample texture as the output pixel. Considering the fact that each and every output pixel is dependent upon the already synthesized pixels, any wrongly synthesized pixels in the course of the system influence the relaxation of the result inflicting propagation of error.

Otori and Kuriyama [12], [13] pioneered the work of combining data coding with pixel-centered texture synthesis. Secret messages to be concealed are encoded into coloured dotted patterns and they are straight painted on a clean snapshot.

A pixel-centered algorithm coats the rest of the pixels utilising the pixel-based texture synthesis approach, for that reason camouflaging the existence of dotted patterns. To extract messages the printout of the stego synthesized texture photo is photographed earlier than applying the data-detecting mechanism. The potential offered through the procedure of Otori and Kuriyama relies on the quantity of the dotted patterns. Nonetheless, their method had a small error rate of the message extraction. Patch-based algorithms [14], [15] paste patches from a source texture as an alternative of a pixel to synthesize textures. This approach of Cohen et al. And Xu et al. Improves the snapshot fine of pixel-headquartered synthetic textures since texture constructions throughout the patches are maintained. However, for the reason that patches are pasted with a small overlapped region throughout the bogus procedure, one desires to make an effort to be certain that the patches trust their neighbors. Secondly, a steganalytic algorithm just isn't extra likely to defeat this steganographic method in view that the stego texture graphic consists of a source texture instead than through using enhancing the present picture contents. Zero.33, the reversible capability inherited from our scheme provides performance to recover the supply texture. Since the recovered supply texture is precisely the equal as the usual deliver texture, it may be employed to proceed onto the 2d round of secret messages for steganography if wanted. Experimental results have demonstrated that our proposed algorithm can furnish particularly a variety of numbers of embedding capacities, produce visually believable texture portraits, and get good the supply texture. Separable reversible data hiding in encrypted image requires different encryption technique, compression technique and data hiding technique for encrypted image. Data hiding technique is to embed some secret information into some carrier signal by altering the insignificant components for copyright protection. In general cases, the data hiding operation will result in distortion in the host signal. However, such distortion is too small and is unacceptable to some applications, such as military or medical images. In this case secret message is embedded with a reversible manner so that the original image contents can be perfectly restored after extraction of the hidden data. A number of reversible data hiding techniques have been proposed, and they can be roughly classified into three types: lossless compression based methods, [3] Digital watermarking techniques [13,14] and Difference expansion (DE) methods [13].The lossless compression based LSB technique[12],performing lossless compression in order to create a spare space to accommodate additional secret data

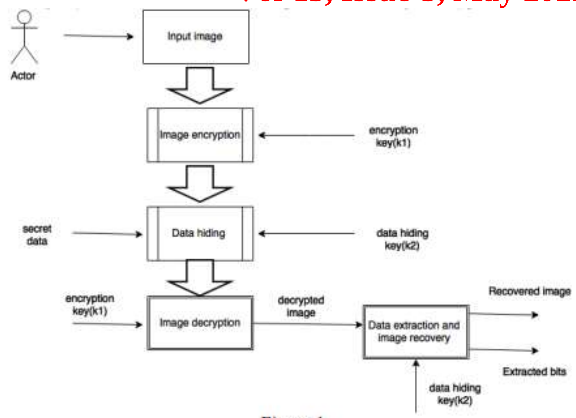


Figure 1

II. LITERATURE SURVEY

A. Separable Reversible Data Hiding in Encrypted Image: -

This technique proposes a novel scheme for separable reversible data hiding in encrypted images [1]. In the first part, a content owner (sender) encrypts the original image i.e. the uncompressed image using key known as an encryption key. Then, the data hider may compress the lower bits i.e. the least significant bits (LSB) of the encrypted image using a new key known as a data-hiding key to create a sparse space to accommodate some additional data. Now with the encrypted image containing the additional data, if a receiver has the data-hiding key, then the receiver can extract the additional data though the receiver does not have an idea about the original image content. If the receiver has encryption key, then the receiver can decrypt the image similar to the original image but receiver cannot extract the additional data. If the receiver has both the keys i.e. data-hiding key and the encryption key, then receiver can extract the additional data and recover the image i.e. the original content of the image without any error by exploiting the spatial correlation.

B. Lossless Compression Method for Encrypted Gray Image: - Wei Liu et.al [3] suggested a lossless compression method for encrypted gray image using progressive decomposition and rate-compatible punctured turbo codes. In this method resolution progressive compression algorithm, that has been shown to have much better coding efficiency and less computational complexity than existing approaches.Wei Liu and et.al observed that lossless compression of encrypted sources can be achieved through Slepian-Wolf coding. For encrypted real-world sources such as images, they are trying to improve the compression efficiency. In this paper researchers proposed a resolution progressive compression scheme which compresses an encrypted image progressively in resolution, such that the decoder can observe a low-resolution version of the image, study local statistics based on it, and use the statistics to decode the next resolution level. Researcher focused on the design and analysis of a practical lossless image codec, where the image data undergoes stream-cipher based encryption before compression. Resolution progressive compression is used for this problem that has much better coding efficiency and less computational complexity than existing approaches.

C. Lossy Compression and Iterative Reconstruction for Encrypted Image: - X. Zhang [10] presented lossy compression method in which an encrypted gray image can be efficiently compressed by discarding the excessively rough and fine information of coefficients generated from orthogonal transform.

When having the compressed data, a receiver may reconstruct the principal content of original image by retrieving the values of coefficients. A pseudorandom permutation is used to encrypt an original image, and the encrypted data are efficiently compressed by discarding the excessively rough and fine information of coefficients generated from orthogonal transform. After receiving the compressed data, with the aid of spatial correlation in natural image, a receiver can reconstruct the principal content of the original image by iteratively updating the values of coefficients. This way, the higher the compression ratio and the smoother the original image, the better the quality of the reconstructed image. The compression ratio and the quality of reconstructed image vary with different values of compression parameters. In the encryption phase , only the pixel positions are shuffled and the pixel values are not masked.

D. A Buyer–Seller Watermarking Protocol: - Nasir Memon and Ping Wah Wong [13] worked on a buyer-seller watermarking protocol that is the concept of digital watermarking. In this protocol researchers stated that the seller does not get to know the exact watermarked copy that the buyer receives. Hence the seller cannot create copies of the original content containing the buyer's watermark. However, in case the seller finds an unauthorized copy and can identify the buyer from whom this unauthorized copy has originated and furthermore also prove this fact to a third party by means of dispute resolution protocol. Hence, the buyer cannot claim that an unauthorized copy may have originated from the seller. The watermark embedding protocol is based on public key cryptography and has little overhead in terms of the total data communicated between the buyer and the seller. Nasir Memon and Ping Wong stated the concept of hiding the data in encrypted form of the data. Here seller is doing data (fingerprint/Watermark in this case.) embedding while he does not know the original data content. The data is in the encrypted form.

E. Buyer-seller watermarking protocol based on homomorphic cryptosystem and composite signal: - M.Deng and et.al [14] proposed Buyer-seller watermarking protocol based on homomorphic cryptosystem and composite signal representation in the encrypted domain. Developed the composite signal representation which allows us to decrease both the computational overhead and the large communication bandwidth which are mostly due to the use of homomorphic public-key encryption schemes. Complexity estimates show that the most computational demanding part of the protocol is the encryption of the content and the embedding of the watermark in the encrypted domain. In order to evaluate the feasibility of this part, a practical implementation of an encrypted domain watermark embedding method, based on different watermarking algorithms, has been implemented and tested on different images. The results show that the version using composite signal representation can run in less than two minutes with a performance in terms of robustness almost indistinguishable from that of the corresponding plaintext embedding algorithms. A stego image includes some distortion, and regardless of how minute it's, this will intrude with the common elements of the quilt photo. This leads to the 2nd trouble for the reason that it's still possible that an snapshot steganalytic algorithm can defeat the image steganography and therefore reveal that a hidden message is being conveyed in a stego image. In this paper, we propose a novel approach for steganography making use of reversible texture synthesis. A texture synthesis approach re-samples a small texture photo drawn via an artist or captured in a image in order to synthesize a new texture snapshot with a identical regional appearance and arbitrary measurement. We weave the texture synthesis process into steganography concealing secret messages As good considering the fact that the source texture. In designated, in big difference to utilizing an existing cover photo to cover messages, our algorithm conceals the supply texture snapshot and embeds secret messages through the system of texture synthesis. This allows for us to extract the key messages and the deliver texture from a stego synthetic texture. To the high-quality of our expertise, steganography taking advantage of the reversibility has ever been awarded throughout the literature of texture synthesis. Our procedure presents three benefits. First, for the reason that the feel synthesis can synthesize an arbitrary measurement of texture pictures, the embedding capability which our scheme offers is proportional to the scale of the stego texture picture. Secondly, a steganalytic algorithm just isn't extra likely to defeat this steganographic method in view that the stego texture graphic consists of a source texture instead than through using enhancing the present picture contents. Zero.33, the reversible capability inherited from our scheme provides performance to recover the supply texture. Since the recovered supply texture is precisely the equal as the usual deliver texture, it may be employed to proceed onto the 2d round of secret messages for steganography if wanted. Experimental results have demonstrated that our proposed algorithm can furnish particularly a variety of numbers of embedding capacities, produce visually believable texture portraits, and get good the supply texture. Separable reversible data hiding in encrypted image requires different encryption technique, compression technique and data hiding technique for encrypted image. Data hiding technique is to embed some secret information into some carrier signal by altering the insignificant components for copyright protection. In general cases, the data hiding operation will result in distortion in the host signal. However, such distortion is too small and is unacceptable to some applications, such as military or medical images. In this case secret message is embedded with a reversible manner so that the original image contents can be perfectly restored after extraction of the hidden data. A number of reversible data hiding techniques have been proposed, and they can be roughly classified into three types: lossless compression based methods, [3] Digital watermarking techniques [13,14] and Difference expansion (DE) methods [13].The lossless compression based LSB technique[12],performing lossless compression in order to create a spare space to accommodate additional secret data

F. Reversible Data Embedding Using a Difference Expansion: - Jun Tian [19] developed a simple and efficient reversible data-embedding method for digital images in that researcher explored the redundancy in the digital content to achieve reversibility. Both the payload capacity limit and the visual quality of embedded images are best. As a basic requirement, system achieved the policy of quality degradation on the image after data embedding should be low.

III. PROPOSED WORK

Texture synthesis has obtained numerous concentration lately in pc imaginative and prescient and laptop graphics [8]. Essentially the most recent work has concerned about texture synthesis by illustration, where a supply texture picture is re-sampled utilizing either pixel-based or patch-situated algorithms to supply a new synthesized texture photograph with similar neighborhood look and arbitrary measurement. Pixel-founded algorithms [9], [10], [11] generate the synthesized image pixel by pixel and use spatial neighborhood comparisons to prefer essentially the most an identical pixel in a sample texture as the output pixel. Considering the fact that each and every output pixel is dependent upon the already synthesized pixels, any wrongly synthesized pixels in the course of the system influence the relaxation of the result inflicting propagation of error.

Otori and Kuriyama [12], [13] pioneered the work of combining data coding with pixel-centered texture synthesis. Secret messages to be concealed are encoded into coloured dotted patterns and they are straight painted on a clean snapshot.

A pixel-centered algorithm coats the rest of the pixels utilising the pixel-based texture synthesis approach, for that reason camouflaging the existence of dotted patterns. To extract messages the printout of the stego synthesized texture photo is photographed earlier than applying the data-detecting mechanism. The potential offered through the procedure of Otori and Kuriyama relies on the quantity of the dotted patterns. Nonetheless, their method had a small error rate of the message extraction. Patch-based algorithms [14], [15] paste patches from a source texture as an alternative of a pixel to synthesize textures. This approach of Cohen et al. And Xu et al. Improves the snapshot fine of pixel-headquartered synthetic textures since texture constructions throughout the patches are maintained. However, for the reason that patches are pasted with a small overlapped region throughout the bogus procedure, one desires to make an effort to be certain that the patches trust their neighbors.

Liang et al. [16] presented the patch-centered sampling process and used the feathering procedure for the overlapped areas of adjacent patches. Efros and Freeman [17] reward a patch stitching procedure referred to as “snapshot quilting”. For each new patch to be synthesized and stitched, the algorithm first searches the supply texture and chooses one candidate patch that satisfies the pre-outlined error tolerance with appreciate to neighbors along the overlapped area. Next, a dynamic programming process is adopted to reveal the minimal error path by means of the overlapped region. This announces an best boundary between the chosen candidate patch and the synthesized patch, producing visually believable patch stitching. Steganography is the method of hiding a message, file, image, or video within another file, message, image, or video. The word steganography combines from the two Greek words “steganos” means “protected”, and “grapheins” means “writing”. The advantage of steganography than cryptography is that the secret message does not attract the attention of the attackers by simple observation. The cryptography protects only the content of the message, while steganography protects the both messages and communication environment. In most of the image steganographic methods, uses the existing image as their cover medium. This leads to two drawbacks. Since the size of the cover image is fixed, embedding a large secret message will results in the distortion of the image. Thus a compromise should be made between the size of the image and the embedding capacity to improve the quality of the cover image. The distortion of the image results in second drawback, because it is feasible that a steganalytic algorithm can defeat the image steganography and thus reveal that a hidden message is conveyed in a stego image. The paper will proposes a good approach for steganography using reversible texture synthesis based on edge adaptive and tree based parity check to improve the embedding capacity.

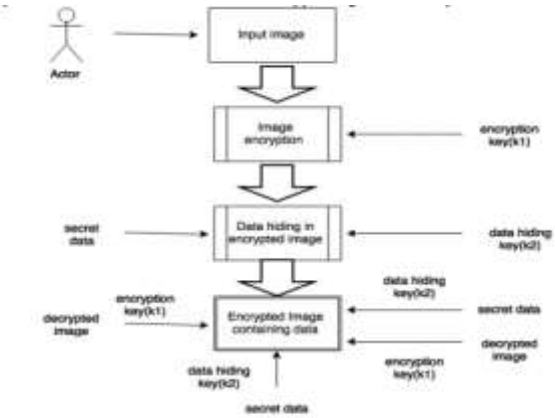


Figure 2

IV METHODOLOGY

*Message Embedding Procedure*2) *Source Texture Recovery, Message Extraction and Message Authentication Procedure3) Capacity Determination*

1. Concepts involved in Message Embedding Procedure

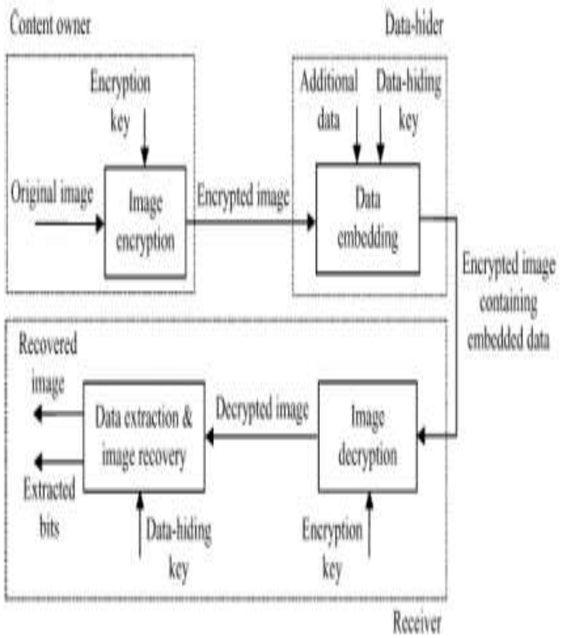
The message embedding procedure involves mainly four steps. They are A) Index Table Generation B.Patch Composition Process C.Combined TBPC and Edge Adaptive Process D.Message Oriented Texture Synthesis Generation.

A. Index Table Generation

The first process of this project is the index table generation where here will create an index table to preserve the location of the source patch set inside the synthetic texture. The index table will allow us to access the synthetic texture and extract the source texture wholly. The texture of any size according to our wish can be generated using this index table.

B.Patch Based Composition

The second step that has to be used in this project is to attach the source patches into a workbench to create a composition image. First here will set up an empty image as the workbench where the size of the workbench is proportional to the synthetic texture. By referring to the source patch IDs stored in the index table, we then attach the source patches into the workbench. During the attaching process, if no imbrications of the source patches are found, we can attach the source patches directly into the workbench.



C.Combined TBPC and Edge Adaptive Process

Embedding capacity is one of the most important requirements for steganography methods, and it is important for steganography process not to leave any noticeable traceable to the human eyes after hiding the secret data. Here will give a hybrid image steganography method that combines edge adaptive and TBPC methods together. The proposed method exploits the high contrast regions of an image as embedding locations. It is known that human eyes cannot discover modifications in the edge areas since they can do in smooth areas. Therefore, the number of hidden bits is on the basis of the variation value between the two pixels of each block. The integration of TBPC leads to a better embedding capacity. Thus, the proposed method mixes up the strengths of edge adaptive and TBPC. D.Message Oriented Texture Synthesis Generation. After the creation of the composition image we have to embed the secret message through the message- oriented texture synthesis to generate the final stego synthetic texture.

1)Concept Involved In Source Texture Recovery, Message Extraction, and Message Authentication Procedure

The message extracted for the receiver side consist of creating the index table, attaining the source texture, performing the texture synthesis, and extracting and authenticating the secret message hidden inside the stego synthetic texture.

2. *Capacity Determination.*

The next step is to look for how much data can be embedded in the stego texture image.The embedding capacity can be related to the capacity in bits that can be hidden at each patch(BPP, bit per patch), and to the number of embeddable patches in the stego synthetic texture (*EPn).* Each patch can hide at least one bit of the secret message.

$$TC = \text{BPP} \times EPn = \text{BPP} \times (TPn - SPn)$$

V. EXPERIMENTAL RESULTS AND ANALYSIS

A. Results of the Embedding Capacity

We collect our experimental results on a personal computer with an i7-2600 3.4GHz CPU and 4GB memory. We adopt four source textures for the results of our collection. Table III presents the total embedding capacity our algorithm can provide when different resolutions of the synthetic texture are produced by concealing various BPPs. It is interesting to point out that given a fixed number of BPP, the larger the resolutions of the source texture Sw×Sh (96×96 vs. 192×192), the smaller the total embedding capacity (TC) our algorithm will offer (6160 bits vs. 5890 bits for 10 BPP). This is because the larger source texture will contain more source patches SPn (9 vs. 36) that we need to paste which cannot conceal any secret bits.

This will reduce the number of embeddable patches (EPn) on the composition image (616 vs. 589), thus reducing the total embedding capacity. Nevertheless, we can employ larger BPP (11 vs. 14) in order to convey more secret messages (6776 bits vs. 8246 bits). The maximal capacity provided by our algorithm is 34398 bits. We compare our embedding capacity with the work presented by Otori and Kuriyama [13]. In their algorithm, a data-coded texture image of 480□640 pixels contains 5□5 or 10□10 dotted patterns, thus concealing the capacity in the range of 200 to 800 bits. In contrast, our scheme conveys the capacity ranging from 12285 to 34398 bits in a stego texture synthesis image of 1024□1024 pixels. Consequently, the capacity we offer varies from 4.50 to 50.39 times more than our counterparts'. Besides, our scheme extracts the secret messages correctly, while their scheme exhibits a small error rate when extracting secret messages. compares stego synthetic textures with different capacities.

We also present the pure synthetic texture which does not convey any secret message. No significant visual difference exists between the two stego synthetic textures and the pure synthetic texture. In addition, no significant visual difference can be perceived when comparing two stego synthetic textures embedded by 5 BPP vs. 10 BPP. The computing times of Fig. 8 are shown in Table IV. The range of the computing time is 6.8% to 8.7% more than that needed for pure texture synthesis. Nevertheless, our algorithm is flexible enough to provide a different embedding capacity by simply altering the patch size, satisfying the desirable capacity or texture quality demanded by users. Also, our algorithm can retrieve the source texture, making possible the second round of message embedding using the recovered source texture.

TABLE III
TOTAL EMBEDDING CAPACITY IN BITS OUR ALGORITHM CAN PROVIDE

$S_w \times S_h$	SP_n	EP_n	TC (5BPP)	TC (10 BPP)	TC (BPP_{max})
Synthetic texture size: $T_w \times T_h$ =1008×1008; Patch size: $P_w \times P_h$ =48×48; Boundary depth: P_d =8					
96×96	9	616	3080	6160	6776
128×128	16	609	3045	6090	7308
192×192	36	589	2945	5890	8246
$T_w \times T_h$ =1024x1024, $P_w \times P_h$ =24x24, P_d =4					
96×96	36	2565	12825	25650	30780
128×128	64	2537	12685	25370	32981
192×192	144	2457	12285	24570	34398

TABLE IV
COMPUTING TIME (SECOND)

Capacity	Pure	4 BPP	5 BPP	8 BPP	10 BPP	12 BPP
Rope net (192×192)	1562	1680	1557	1680	1541	1680
Metal (192×192)	1671	1816	1665	1768	1644	1816
Peanuts (96×96)	141	141	141	141	136	N/A
Ganache (128×128)	385	402	385	402	385	411

B. Image Quality Comparison

We use several mechanisms to determine the image quality of the stego synthetic texture. We define the first measurement, which is called the mean squared error of the overlapped area (MSEO) to determine the image quality of the synthetic results. MSEO reflects the similarity between the candidate patch and the synthesized area where we will specifically operate the image quilting technique during the message-oriented texture synthesis process. Consequently, the MSEO has a non-zero value even in the case of the pure patch-based texture synthesis.

If the MSEO produces a small value, it implies that the synthetic texture shows a high image quality of the overlapped areas. Obviously, the lower the MSEO value, the higher quality of the synthetic texture image. The equation of MSEO is shown in (6), where OLi stands for the overlapped area of the working location, i, p   stands for the pixel j of the candidate patch in OLi, and p   stands for the pixel j of the synthesized area in OLi.

The MSEO can be calculated during the texture synthesis procedure. When we synthesize a patch, we accumulate the squared errors of all the pixels in the overlapped area between the synthesized area and the selected candidate patch. When the texture synthesis procedure is finished, we can divide the summation of squared errors by the patch size and the number of the synthesized patches to produce the MSEO. It comes as no surprise that the MSEO is dependent on the capacity per patch. Table V compares the MSEO to the embedding capacity for the four test source textures. When we consider the same patch size, the MSEO increases as we convey more secret bits per patch. These MSEO values coincide with the image quality when visualizing the stego synthetic texture. For a larger patch size, the boundary depth increases leading to a larger MSEO.

Taking the test source texture “peanuts” as an example, the MSEO is 1034.6 when conveying 5 bits of secret message per patch, while it shows the MSEO of 1236.9 when conveying 10 bits per patch. Fig. 8(c) shows a slightly higher visual image quality than Fig. 8(d). The second scheme we adopt for measuring the image quality is the Pearson Product

Moment Correlation (PPMC) [20]. It is employed as a measure of how well two variables are related. The Pearson coefficient values that this scheme produces are between 1 and -1. A result of 1 means that there is a perfect positive correlation between the two variables, while a result between 0.5 and 1.0 represents a high correlation. Table VI shows average Pearson coefficients across three R-G-B channels measured under the base of the histogram produced for each channel. In Case-A and Case-B, we reveal the Pearson coefficients between the stego synthetic textures and the pure synthetic texture. We report in Case-C the Pearson coefficients between two stego synthetic textures with different BPPs.

The Pearson coefficients shown in Case-A are very close to 1.0. This represents the high correlation between the pure and stego synthetic textures. In Case-B, although the Pearson coefficient values slightly decrease along with the increase of the embedding capacity per patch, they retain the feature of a close relation between the pure and stego synthetic textures. The Pearson coefficients in Case-C are very close as well, which indicates that the stego synthetic textures conveying 5 BPP remain highly correlated to those conveying 10 BPP. All the Pearson coefficients shown in this table exhibit that our steganographic algorithm is effective in producing stego synthetic textures of high image quality in comparison to pure synthetic textures produced by conventional texture synthesis. Next, we employ the SSIM (Structural SIMilarity) index [21] to quantify the similarity between the pure and stego synthetic textures.

The SSIM is an image quality assessment method for measuring the change in luminance, contrast, and structure in an image. The SSIM index is in the range of [-1, 1] and when it equals to 1, the two images are identical. Table VII shows the SSIM indices for different patch sizes. The SSIM index values are close despite their concealing different embedding capacities.

The SSIM algorithm is highly sensitive to translation, scaling and rotation. This explains the low index values shown in Table VII because the patch-based method generates an entirely new synthetic texture image from a number of patches in the source texture resulting in a different level of image translation and scaling.

TABLE V
A COMPARISON OF MSEO WITH RESPECT TO EMBEDDING CAPACITY

	Patch size: $P_w \times P_h$ =24x24 Boundary depth: P_d =4			Patch size: $P_w \times P_h$ =48x48 Boundary depth: P_d =8		
	Pure	5 BPP	10 BPP	Pure	5 BPP	10 BPP
Rope net	651.2	899.4	1327.6	936.3	1224.2	1565.3
Metal	595.6	795.2	1096.4	837.4	1132.3	1391.1
Peanuts	547.8	776.9	1053.1	918.2	1034.6	1236.9
Ganache	184.5	219.3	318.3	365.3	483.1	556.2

TABLE VI
COMPARISON OF PEARSON CORRELATION COEFFICIENTS

	Case-A	Case-B	Case-C
	Pure vs. 5 BPP	Pure vs. 10 BPP	5 BPP vs. 10 BPP
Rope net	0.9991	0.9991	0.9987
Metal	0.9990	0.9984	0.9995
Peanuts	0.9980	0.9974	0.9993
Ganache	0.9991	0.9955	0.9980

Patch size: $P_w \times P_h$=48×48, boundary depth: P_d =8.

TABLE VII
COMPARISON OF SSIM INDEX

	Patch size: $P_w \times P_h$ =24x24 Boundary depth: P_d =4 $T_w \times T_h$ =504×504		Patch size: $P_w \times P_h$ =48x48 Boundary depth: P_d =8 $T_w \times T_h$ =488×488	
Pure vs.	5 BPP	10 BPP	5 BPP	10 BPP
Rope net	0.31	0.27	0.34	0.28
Metal	0.28	0.26	0.28	0.26
Peanuts	0.08	0.08	0.08	0.07
Ganache	0.18	0.16	0.17	0.15

CONCLUSION

This project proposes a reversible steganographic algorithm using texture synthesis based on edge adaptive and tree based parity check. Given an original source texture, first we have to produce a large stego synthetic texture hiding the secret messages. By using a conventional patch-based method the textures are synthesized. The project will also provides reversibility to retrieve the original source texture from the stego synthetic textures, making possible a second round of texture synthesis if needed. This paper also introduce another image steganography method that combines the edge adaptive and TBPC algorithms to heighten the payload and imperceptibility of the stego image, and thus minimizing the possible distortion during the embedding process to minimize the probability of discovering the secret message data from unauthorized users and also resulting in high embedding capacity. Given an customary source texture, our scheme can produce a enormous stego artificial texture concealing secret messages. To the quality of our capabilities, we're the primary that can exquisitely weave the steganography into a traditional patch-headquartered texture synthesis. Our approach is novel and presents reversibility to retrieve the usual source texture from the stego artificial textures, making feasible a 2d circular of texture synthesis if wanted. With the 2 approaches now we have offered, our algorithm can produce visually plausible stego synthetic textures although the key messages including bit “zero” or “1” have an uneven appearance of probabilities. The provided algorithm is comfy and powerful against an RS steganalysis attack. We consider our proposed scheme offers colossal benefits and supplies an possibility to extend steganographic functions. One feasible future be trained is to expand our scheme to help different varieties of texture synthesis techniques to improve the photo high-quality of the bogus textures. One other possible learn would be to mix other steganography procedures to expand the embedding capacities.

REFERENCE

[1] Y. Guo, G. Zhao, Z. Zhou, and M. Pietikäinen, “Video texture synthesis with multi-frame LBP-OP and diffeomorphic growth model,” *IEEE Trans. Image Process.*, vol. 22, no. 10, pp. 3879–3891, Oct. 2013.

[2] L.-Y. Wei and M. Levoy, “Fast texture synthesis using tree- structured vector quantization,” in *Proc. 27th Annu. Conf. Comput. Graph. Interact. Techn*, 2000, pp. 479–488.

[3] A. A. Efros and T. K. Leung, “Texture synthesis by non-parametric sampling,” in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, Sep. 1999, pp. 1033–1038.

[4] C. Han, E. Risser, R. Ramamoorthi, and E. Grinspun, “Multiscale texture synthesis,” *ACM Trans. Graph.*, vol. 27, no. 3, 2008, Art. ID 51.

[5] H. Otori and S. Kuriyama, “Data-embeddable texture synthesis,” in *Proc.8th Int. Symp. Smart Graph*, Kyoto, Japan, 2007, pp. 146–157. [6] H. Otori and S. Kuriyama, “Texture synthesis for mobile data communications,”*IEEE Comput. Graph. Appl.*, vol. 29, no. 6, pp. 74–81, Nov. /Dec. 2009.

[7] M. F. Cohen, J. Shade, S. Hiller, and O. Deussen, “Wang tiles for image and texture generation,” *ACM Trans. Graph.*, vol. 22, no. 3, pp. 287–294, 2003.

[8] K. Xu *et al.*, “Feature-aligned shape texturing,” *ACM Trans. Graph.*, vol. 28, no. 5, 2009, Art. ID 108.

[9] L. Liang, C. Liu, Y.-Q. Xu, B. Guo, and H.-Y. Shum, “Real-time texture synthesis by patch-based sampling,” *ACM Trans. Graph.*, vol. 20, no. 3, pp. 127–150, 2001.

[10] A. A. Efros and W. T. Freeman, “Image quilting for texture synthesis and transfer,” in *Proc. 28th Annu. Conf. Comput. Graph. Interact. Techn*, 2001, pp. 341–346.

[11] 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.

[12] Li, R.Y., Au, O.C., Lai, K.K., Yuk, C.K., Lam, and S.Y.: *"Data hiding with tree based parity check".* In: 2007 IEEE International Conference on Multimedia and Expo,pp. 635–638. IEEE (2007)

[13] Hou, C.L., Lu, C., Tsai, S.C., Tzeng, W.G.: *"An optimal data hiding scheme with tree-based parity check.*" IEEE Transactions on Image Processing 20(3), 880–886 (2011)

[14] Liu, G., Liu, W., Dai, Y., Lian, S.: *Adaptive steganography based on block complexity and matrix embedding .Multimedia Systems*20 (2), 227–238(2014)

[15] X. Li, B. Li, B. Yang, and T. Zeng, *"General framework to histogram shifting-based reversible datahiding, IEEE Trans.Image Process"* vol.22,no.6,pp.2181– 2191,Jun.2013.

[16] M. Cancellaro, F. Battisti, M. Carli, G. Boato, F. G. B. Natale, and A. Neri, "A commutative digital image watermarking and encryption method in the tree structured Haar transform domain," Signal Processing:\\Image Commun., vol. 26, no. 1, pp. 1–12, 2011.

[17] D. Kundur and K. Karthik, "Video fingerprinting and encryption principles for digital rights management," Proceedings IEEE, vol. 92, no. 6, pp. 918–932, Jun. 2004.

[18] X. Zhang, "Reversible data hiding in encrypted image," IEEE Signal Process. Lett., vol. 18, no. 4, pp. 255–258, Apr. 2011.

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

[20] 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.

[21] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Lossless generalized- LSB data embedding," IEEE Trans. Image Process., vol. 14, no.2, pp. 253–266, Feb. 2005.