

# An Efficient Steganographic Method Using Reversible Texture Synthesis

Munshidha K K<sup>1</sup>, Anju Augustine<sup>2</sup>

<sup>1,2</sup>Department of Computer Science and Engineering, KMP College of Engineering, Asamannoor P.O Poomala, Odakkali, Kerala, India

**Abstract:** A new steganographic method using reversible texture synthesis is proposed in this paper. A texture synthesis process re-samples a smaller texture image which synthesizes a new texture image with a similar local appearance and arbitrary size. Along with reversible texture synthesis process, reserving room method is used to embed additional data, which is the highlight of this paper. Traditional RDH algorithm is used for reserving room method. Data hider can reversibly embed additional data in stegosynthetic texture image. The proposed method can achieve real reversibility, that is, texture image recovery and data extraction cause no error.

**Keywords:** Steganography, RDH, Data embedding, Reversible data hiding, Texture synthesis

## 1. Introduction

In the most recent decade numerous advances have been made in the range of computerized media, and much concern has emerged with respect to steganography for computerized media. Steganography is a solitary system for data concealing strategies. It implants messages into a host medium keeping in mind the end goal to cover mystery messages so as not to excite suspicion by a meddler. A normal steganographic application incorporates secretive correspondences between two gatherings whose presence is obscure to a conceivable assailant and whose achievement relies on upon identifying the presence of this correspondence. When all is said in done, the host medium utilized as a part of steganography incorporates significant advanced media, for example, computerized picture, content, sound, video, 3D model, and so forth. Countless steganographic calculations have been researched with the expanding notoriety and utilization of advanced pictures.

Most image steganographic calculations receive a current picture as a spread medium. The cost of installing mystery messages into this spread picture is the picture bending experienced in the stego picture. This prompts two disadvantages. To begin with, since the span of the spread picture is settled, the more mystery messages which are inserted take into account more picture twisting. Hence, a bargain must be came to between the inserting limit and the picture quality which brings about the constrained limit gave in any particular spread picture. Review that picture steganalysis is a methodology used to distinguish mystery messages covered up in the stego image. A stego image contains some twisting, and paying little respect to how minute it is, this will meddle with the regular elements of the spread picture. This prompts the second disadvantage on the grounds that it is still conceivable that a picture steganalytic calculation can crush the picture steganography and in this way uncover a concealed message is being passed on in a stego image.

In this task, a novel methodology for steganography utilizing reversible texture synthesis was proposed. A texture synthesis process re-tests a little composition picture drawn by a craftsman or caught in a photo to blend another surface

picture with a comparative neighborhood appearance and discretionary size. We weave the composition blend process into steganography covering secret messages and in addition the source surface[1]. Specifically, rather than utilizing a current spread picture to shroud messages, our calculation hides the source composition picture and implants secret messages through the procedure of surface union. This permits us to separate the mystery messages and the source composition from a stego engineered surface. To the best of our insight, steganography exploiting the reversibility has ever been introduced inside of the writing of surface amalgamation. Our methodology offers three points of interest. To begin with, since the surface combination can incorporate a discretionary size of composition pictures, the inserting limit which our plan offers is corresponding to the measure of the stego composition picture. Besides, a steganalytic calculation is not prone to overcome this steganographic approach following the stego composition picture is made out of a source surface as opposed to by adjusting the current picture substance. Third, the reversible ability acquired from our plan gives usefulness to recoup the source surface. Since the recouped source surface is precisely the same as the first source composition, it can be utilized to continue onto the second round of mystery messages for steganography if necessary. Trial results have confirmed that our proposed calculation can give different quantities of implanting limits, deliver outwardly conceivable composition pictures, and recoup the source surface. Hypothetical separating our steganographic methodology, and the plan can oppose a RS steganalysis assault.

## 2. Related Work

Texture synthesis has gotten a great deal of consideration as of late in computer vision and PC graphics. The latest work has concentrated on texture synthesis by sample, in which a source texture image is re-examined utilizing either pixel-based or patch-based algorithms to deliver another synthesized texture image with comparative neighborhood appearance and subjective size. Pixel-based algorithms produce the orchestrated picture pixel by pixel and use spatial neighborhood correlations to pick the most comparable pixel in an example composition as the yield

pixel. Since every yield pixel is dictated by the as of now integrated pixels, any wrongly blended pixels amid the procedure impact whatever remains of the result bringing about proliferation of blunders.

Otori and Kuriyama [6], [7] spearheaded the work of consolidating information coding with pixel-based texture synthesis. Secret messages to be hidden are encoded into hued spotted examples and they are straightforwardly painted on a blank image. A pixel-based algorithm coats whatever is left of the pixels utilizing the pixel-based texture synthesis strategy, in this manner disguising the presence of spotted examples.

Patch-based algorithms [8], [9] glue patches from a source composition rather than a pixel to synthesise textures. This methodology of Cohen et al. also, Xu et al. enhances the picture nature of pixel-based engineered surfaces in light of the fact that composition structures inside the patches are kept up. Then again, since patches are stuck with a little covered district amid the manufactured procedure, one necessities to try to guarantee that the patches concur with their neighbors.

Liang et al. [10] presented the patch-based sampling methodology and utilized the feathering methodology for the covered ranges of nearby patches. Efros and Freeman [11] present a patch sewing methodology called "image quilting". For each new patch to be blended and sewed, the calculation first inquiries the source composition and picks one applicant fix that fulfills the pre-characterized blunder resistance regarding neighbors along the covered district. Next, a dynamic programming strategy is adopted to uncover the minimum error path way through the covered region. This announces an ideal limit between the picked applicant patch and the blended patch, delivering outwardly conceivable patch sewing.

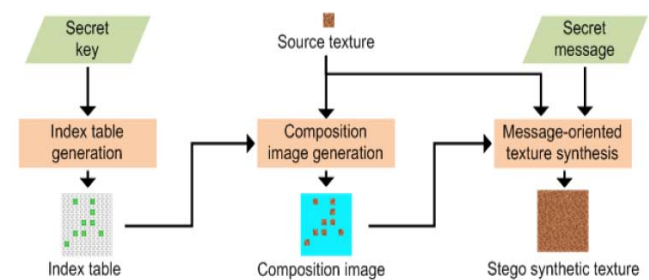
Ni et al. [12] proposed a image reversible information hiding away algorithm which can recuperate the cover image with no contortion from the stego picture after the concealed information have been extracted. Histogram shifting is a favored system among existing methodologies of reversible picture information hiding on the grounds that it can control the alteration to pixels, along these lines restricting the inserting distortion, and it just requires a little size area guide, subsequently lessening the overhead experienced. The

current best in class for reversible image information covering up is the general system displayed by Li et al. [13].

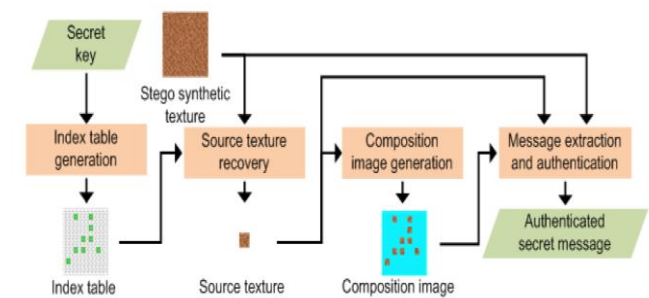
To the best of our insight, we were not able to uncover any writing that related patch-based composition amalgamation with steganography. In this paper, we show our work which takes favorable position of the patch-based techniques to insert a secret message amid the incorporating methodology. This permits the source texture to be recouped in a message extricating system, giving the usefulness of reversibility. We point of interest our strategy in the following segment.

### 3. Existing System

Texture synthesis and message extraction are as proposed in [1].

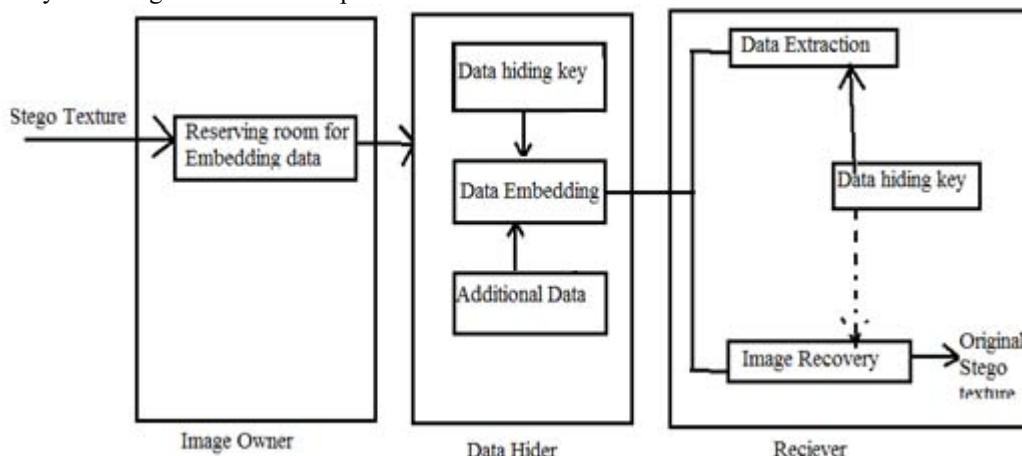


**Figure 1:** The flowchart of the three-process message embedding procedure.



**Figure 2:** The flowchart of the four-step message extracting procedure.

### 4. Proposed System



**Figure 3:** Additional Data embedding by Reserving room method

#### 4.1 Reserving Room for Embedding

Actually, to reserve room for embedding, the first stage can be divided into two steps: image partition and self reversible embedding. At the beginning, image partition step divides original image into two parts **A** and **B**; then, the LSBs **A** are reversibly embedded into **B** with a standard RDH algorithm so that LSBs of **A** can be used for accommodating messages; at last rearranged image to generate its final version.

##### 4.1.1 Image Partition

For reserving room, the operator here is a standard RDH technique, so the goal of image partition is to construct a smoother area **B**, on which standard RDH algorithms such as can achieve better performance. To do that, without loss of generality, assume the stego texture **C** is an 8 bits gray-scale image with its size  $M \times N$  and pixels  $C_{i,j} \in [0,255], 1 \leq i \leq M, 1 \leq j \leq N$ . First, the content owner extracts from the original image, along the rows, several overlapping blocks whose number is determined by the size of to-be-embedded messages, denoted by  $l$ . In detail, every block consists of  $m$  rows, where  $m = \lceil l / N \rceil$ , and the number of blocks can be computed through  $n = M - m + 1$ . An important point here is that each block is overlapped by pervious and/or sub sequential blocks along the rows. For each block, define a function to measure its first-order smoothness.

$$f = \sum_{u=2}^m \sum_{v=2}^{N-1} |C_{u,v} - \frac{C_{u-1,v} + C_{u+1,v} + C_{u,v-1} + C_{u,v+1}}{4}| \quad (1)$$

Higher  $f$  relates to blocks which contain relatively more complex textures. The content owner, therefore, selects the particular block with the highest  $f$  to be **A**, and puts it to the front of the image concatenated by the rest part **B** with fewer textured areas, as shown in Fig. 4.

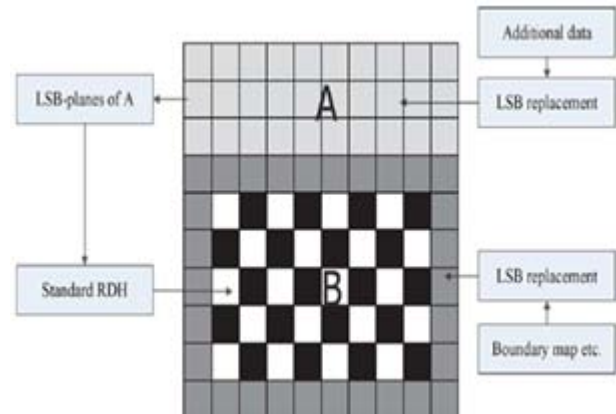
##### 4.1.2 Self Reversible Embedding

The goal of self-reversible embedding is to embed the LSB-planes of **A** into **B** by employing traditional RDH algorithms. For illustration, we simplify the method in to demonstrate the process of self-embedding. Note that this step does not rely on any specific RDH algorithm. Pixels in the rest of image **B** are first categorized into two sets: white pixels with its indices  $i$  and  $j$  satisfying  $(i + j) \bmod 2 = 0$  and black pixels whose indices meet  $(i + j) \bmod 2 = 1$ , as shown in

Fig. 4. Then, each white pixel,  $B^{i,j}$ , is estimated by the interpolation value obtained with the four black pixels surrounding it as follows.

$$B'_{i,j} = w_1 B_{i-1,j} + w_2 B_{i+1,j} + w_3 B_{i,j-1} + w_4 B_{i,j+1} \quad (2)$$

Where the weight  $w_i, 1 \leq i \leq 4$ , is determined by the same method as proposed [16]. The estimating error is calculated via  $e_{i,j} = B_{i,j} - B'_{i,j}$  and then some data can be embedded into the estimating error sequence with histogram shift, which will be described later. After that, we further calculate the estimating errors of black pixels with the help of surrounding white pixels that may have been modified.



**Figure 4:** Illustration of image partition and embedding process

Then another estimating error sequence is generated which can accommodate messages as well. Furthermore, we can also implement multilayer embedding scheme by considering the modified **B** as “original” one when needed. In summary, to exploit all pixels of **B**, two estimating error sequences are constructed for embedding messages in every single-layer embedding process. By bidirectional histogram shift, some messages can be embedded on each error sequence. That is, first divide the histogram of estimating errors into two parts, i.e., the left part and the right part, and search for the highest point in each part, denoted by  $LM$  and  $RM$ , respectively. For typical images,  $LM = -1$  and  $RM = 0$ . Furthermore, search for the zero point in each part denoted by  $LN$  and  $RN$ . To embed messages into positions with an estimating error that is equal to  $RM$ , shift all error values between  $RM + 1$  and  $RN - 1$  with one step toward right, and then, we can represent the bit 0 with  $RM$  and the bit 1 with  $RM + 1$ . The embedding process in the left part is similar except that the shifting direction is left, and the shift is realized by subtracting 1 from the corresponding pixel values [15].

## 5. Conclusion

A reversible steganographic algorithm using texture synthesis is employed in this paper. Given an original source texture, this scheme can produce a large stego synthetic texture concealing secret messages. This can exquisitely weave the steganography into a conventional patch-based texture synthesis. The method is novel and provides reversibility to retrieve the original source texture from the stego synthetic textures, making possible a second round of texture synthesis if needed. Visually plausible stego synthetic textures are produced. The presented algorithm is secure and robust against an RS steganalysis attack. The proposed method can take advantage of all traditional RDH techniques for texture images and achieve excellent performance without loss of perfect secrecy. Real reversibility is achieved.

## References

- [1] W. Kuo-Chen and W. Chung-Ming, "Steganography Using Reversible Texture Synthesis," *IEEE Transactions on Image Processing*, pp. 1-10, 2014.
- [2] Y. Guo, G. Zhao, Z. Zhou, and M. Pietikäinen, "Video texture synthesis with multi-frame LBP-TOP and diffeomorphic growth model," *IEEE Trans. Image Process.*, vol. 22, no. 10, pp. 3879-3891, 2013.
- [3] L.-Y. Wei and M. Levoy, "Fast texture synthesis using tree-structured vector quantization," in *Proc. of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, 2000, pp. 479-488.
- [4] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in *Proc. of the Seventh IEEE International Conference on Computer Vision*, 1999, pp. 1033-1038.
- [5] C. Han, E. Risser, R. Ramamoorthi, and E. Grinspun, "Multiscale texture synthesis," *ACM Trans. Graph.*, vol. 27, no. 3, pp. 1-8, 2008.
- [6] H. Otori and S. Kuriyama, "Data-embeddable texture synthesis," in *Proc. of the 8th International Symposium on Smart Graphics*, Kyoto, Japan, 2007, pp. 146-157.
- [7] H. Otori and S. Kuriyama, "Texture synthesis for mobile data communications," *IEEE Comput. Graph. Appl.*, vol. 29, no. 6, pp. 74-81, 2009.
- [8] 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.
- [9] K. Xu, D. Cohen-Or, T. Ju, L. Liu, H. Zhang, S. Zhou, and Y. Xiong, "Feature-aligned shape texturing," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 1-7, 2009.
- [10] 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.
- [11] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proc. of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, 2001, pp. 341-346.
- [12] 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, 2006.
- [13] X. Li, B. Li, B. Yang, and T. Zeng, "General framework to histogram-shifting-based reversible data hiding," *IEEE Trans. Image Process.*, vol. 22, no. 6, pp. 2181-2191, 2013.
- [14] J. L. Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *The American Statistician*, vol. 42, no. 1, pp. 59-66, 1988.
- [15] M. Kede, Z. Weiming, Z. Xianfeng, Y. Nenghai, and L. Fenghua, "Reversible Data Hiding in Encrypted Images by Reserving Room Before Encryption," *IEEE Trans. Information forensics and security.*, vol. 8, no. 3, pp. 553-562, 2013.
- [16] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 890-896, Aug. 2003

## Author Profile



**Munshidha K.K** received the B.Tech Degree in Computer Science from Cochin University of Science and Technology, Kerala, India, in 2010. She is currently pursuing M.Tech Degree in Computer Science and Engineering with Specialization in Cyber Security from Mahatma Gandhi University, Kerala, India. Her research interests include information security.



**Anju Augustine** received the B.Tech Degree in Information Technology, and M.Tech Degree in Computer Science and Engineering with Specialization in Information Systems from Mahatma Gandhi University, Kerala, India, in 2010 and 2014 respectively. She is currently working as Assistant Professor at K.M.P College of Engineering, Kerala, India..