IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-4, Issue-7 E-ISSN: 2347-2693

An Approach for Data Hiding Technique Based on Reversible Texture Synthesis

Akhila Sreenivas K.[1*] and Pretty Babu [2]

[1]*Department of Computer science and Engineering, Kerala University, India*
[2] *Department of Computer science and Engineering, Kerala University, India*

Available online at: www.ijcseonline.org

Received:11/Jun/2016 Revised: 22/Jun/2016 Accepted: 16/Jul/2016 Published: 31/Jul/2016

***Abstract*—** This paper proposes an improved method for steganography using reversible texture synthesis. Texture synthesis mechanism is the construction of large texture by using the input texture which is smaller in size. Combination of texture synthesis and steganography can provide better security and high embedding capacity than the existing method. Proposed method use location patch or header block to store the location of the embedded data. In the existing method the secret data was placed in source patches where as in the improved method the secret data is kept in blocks. Length of the embedding message is calculated which determines the required number of blocks needed for placing the message securely.

***Keywords*—** *Texture synthesis, Steganography, patch, Synthetic image, Header block, Data block*

I. INTRODUCTION

Steganographic concept has received a lot of attention in the field of digital media. Steganography is defined as the process of hiding a secret message inside in an ordinary message. It is an information hiding mechanism that conceals the secret messages in a secure way. Steganographic mechanism is applicable to confidential communication and secret data sharing. The word steganography comes from 2 Greek words "steganos" and "graphia". Meaning of steganos is covered and graphia means writing. So steganography is also called covered writing. Steganographic classification includes text based steganography, video steganography, image steganography and audio steganography. Image distortion is one of the main disadvantage of image steganographic algorithm. Image steganography means hiding of messages in an image. These types of algorithms uses already existing image as the cover medium. So it does not allow embedding of more secret bits because of the fixed size of cover image.

By giving proper security to our data it is impossible for an unauthorized person to access the data. Steganography is a security tool that gives protection to the data and also prevents the data from malicious attackers. Steganography can conceal a secret message with in an object. That object may be text, image, audio or video. Steganographic cover image means the medium in which the information is to be secretly placed. Stego image means a medium in which the hidden information is placed. Steganalysis is the process of detecting a secret communication. The two main important aspects of steganalysis are detecting hidden information and disabling steganography.

In this paper describes an innovative method of steganography using reversible texture synthesis to hide data in texture images. Texture synthesis mechanism means creating or forming a large texture from the input texture. Synthesized texture appears larger than the input texture but have similar visual appearance. In order to conceal secret messages we combine the texture synthesis process into steganography which hides the source texture and also it hides the embedded secret messages. This method also provides an advantage of reversibility of source texture. Proposed method provides mainly three advantages. It offers increased embedding capacity compared to the existing technique without changing the quality of image. In the existing technique the number of source patches available for data embedding depends on the number of non negative entries in the index table. Next advantage is that increased security**.** In the existing method the index table contains the information about where the data is being embedded. In this paper the embedding information is included in the synthetic image itself. Also the data embedded in the synthetic texture is hidden inside the original synthetic texture which is not easily detectable. Proposed system also offers reduced need of data structure because here there is no need of index table. Therefore processing time and work can be reduced.

Texture synthesis can create varying size of texture images. Procedural based and exemplar based classification of texture synthesis existed. In this paper we use patch based texture synthesis approach comes under exemplar approach. Patch based texture synthesis approach provides better result than the other texture synthesis approach. Patch based algorithms uses patches from the source texture. Patch means an image block of an input texture and the size of the patch can be determined by the user. A patch contains two parts .They are the central part and the kernel part. Central part called the kernel region and the surrounding part called the boundary region. In this method we consider the kernel region to embed secret messages and also that region performs texture synthesis. Proposed method offers the

message embedding and message extraction procedures and also provides an advantage of source texture recovery so the same texture can used for second time also.

This paper is organized as follows section II describes about the related works. Section III contains existing method. Section IV describes the proposed method. Finally Section V contains the conclusion.

II. RELATED WORKS

A.A Effros and T.K Leung used pixel based algorithms to generate synthesized texture image [1]. For choosing the most similar pixel from an input sample texture as the output pixel uses spatial neighborhood comparisons. If a pixel is wrongly synthesized during the processing time it will cause error to the remaining result. Chih -wei fang and Jenn-Jier James Lien also uses pixel based algorithm for synthesizing a large image [2]. H.otori and S.kuriyama developed a pixel based texture synthesis for embedding secret data [3] .Secret messages are encoded into coloured dotted patterns and they are pasted on a blank image. Capacity of embedding data depends upon on dotted patterns. This method provides a small propagation of errors during message extraction.

K. Xu et al, M.F Cohen and J. shade developed a patch based mechanism to improve the quality of image instead of pixel based approach [4-5]. During the texture synthesis process patches are placed within a small overlapped region and to make sure that the patches will accept their neighbors. C. Han, E. Risser, and R. Ramamoorty used patch based synthesis for removing blurriness in an image [6]. Lexing Ying, Aaron Hertzmann, Henning Biermann takes common type of textures such as color, displacement, and transparency are used for synthesizing texture images [7]. Texture synthesis can be done directly on the surface and provides better quality images with less distortion. L. Liang et al developed a patch based sampling for real- time textures [8]. Uses the rowing approach for the overlapped regions of nearby patches. Efros and freeman developed an image quilting approach based on patch based algorithms [9]. Dynamic programming technique is used to find the minimum error path between the overlapped regions. This mechanism provides a visually plausible message by patch stitching. Li-Yi Wei and Jianwei Han et al developed an Inverse texture synthesis method [10]. Develop a small texture as output from the input texture. Efficient quality of images is generated with small data size and allows the reconstruction of the original data through the control map concept.

Z. Ni, Y.-Q. Shi, N. Ansari, and W. Su, proposes a reversible data hiding technique for image [11]. This mechanism allows the restoring of cover image with no image distortion. This approach uses histogram shifting technique for reversible data hiding method. This technique can control the pixel modifications and also having limited embedded distortion. X. Li, B. Li, B Yang, and T. Zeng presented a general framework to histogram shifting without image distortion [12]. Kuo -Chen Wu, Chung-Ming Wang proposed steganographic reversible texture synthesis algorithm use patch-based method to embed secret messages through the texture synthesis procedure [13]. The number of source patches available for data embedding depends on the number of non negative entries in the index table. So it offers limited embedding capacity and can embed data only to the non negative entries.

III. EXISTING SYSTEM

Steganographic reversible texture synthesis algorithm will take the advantage of patch-based method to embed secret messages through the texture synthesis procedure. The basic component used in this method is patch. A patch means an image block of a source texture whose size can be specified by user .A patch contains two specific parts. First is the central part called the kernel region and second part is the path surrounding the kernel region called the boundary region. Patches are combined together to form the composition image in which we are embedding our secret messages. Existing method includes message embedding procedure, message extraction and message authentication procedure and source texture recovery.

A. Message Embedding Procedure

The message embedding procedure consists of mainly 3 steps. They are a) Index Table Generation b) Patch Composition Process c) Message Oriented Texture Synthesis Generation.

a. Index Table Generation

The first process of this method is the index table creation helps to preserve the location of the source patch placed inside the synthetic texture. The index table generated allows to access the synthetic texture and helps us to extract the source texture fully. To achieve the manner of reversibility source texture can be arranged in two ways. First is the sparse manner and second is the dense manner. If the synthetic texture has a resolution much greater than that of source texture we follow sparse manner and the synthetic texture has a resolution slightly larger than that of source texture follow the dense manner representation.

b. Patch Composition Process

Second procedure involves in this mechanism is to fix the source patches into a workbench for the creation of composition image. Initially we build a blank image that blank image will be considered as the workbench .Size of the workbench is equal to the synthetic texture. By considering the source patch IDs stored in the index table, we then paste the source patches into the workbench. If there is no overlapping of source patches are found out during the pasting process we can attach the source patches directly into the work bench.

c. Message Oriented Texture Synthesis Generation

After the creation of the index table and the composition image we can embed the secret messages through the process of texture synthesis and generate final stego synthetic texture.

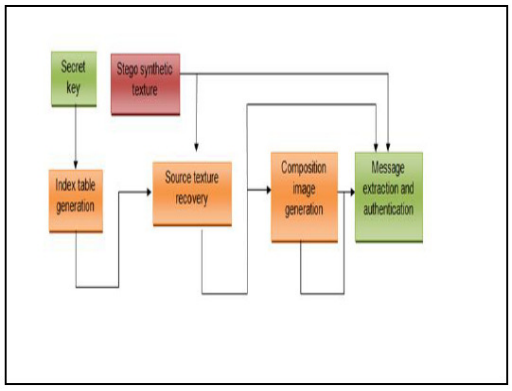

Fig 1: Message embedding procedure mechanism

B. Procedure for Message Extraction, Message Authentication and Source Texture Recovery .

Procedure for extraction of message involves the generation of index table, source texture recovery, composite image generation and authentication of secret messages. By using the secret key in the receiver side the index table as seen in the embedding procedure can be formed. After the creation of index table next process is the source texture recovery. it can be retrieved by referring the dimensions of index table. Next step is to form the composition image by pasting the source patches into a work bench with the help of index table. Final step is the message extraction and the authentication procedure.

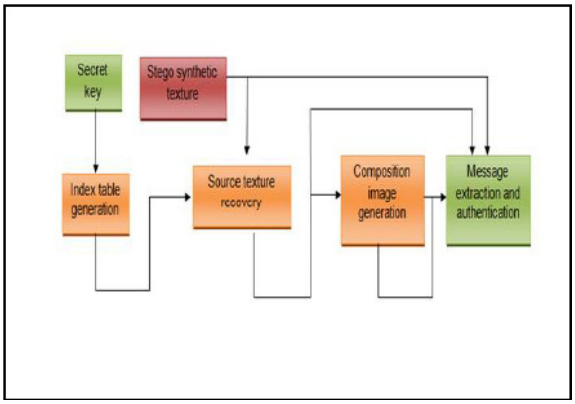

Fig 2: Message Extraction Procedure Mechanism

Disadvantage of the existing system is that the limited embedding capacity. In the existing system we cannot embed the data into the locations of index table generated source patches .So we can embed data only to the remaining patches in the synthetic image. So it offers limited embedding capacity.

IV. PROPOSED METHOD

Proposed method takes input as the source texture. This system can provide the advantage of data embedding and data extraction mechanism. We can embed the data into the data block with the help of header block. Main function of header block is to store the location of embedded data in the blocks. Embedding capacity of this method is higher than the existing technique. In the existing method the data was embedded in individual source patches where as in the enhanced data hiding method the synthetic image is divided into number of non overlapping blocks. To these blocks data is embedded by referring the header blocks.

This method offers increased security than the existing techniques. In this method the embedding information is included in the synthetic image itself and also the data embedded synthetic texture is hidden inside the original synthetic texture which is not easily detectable. Another advantage is that reduced need of data structure. Therefore the processing time and work is reduced. Enhanced data hiding method includes the following modules A) Data embedding procedure B) Data extraction and Data authentication procedure C) Source texture recovery.

A. Data Embedding Procedure

In this section describes about the data embedding mechanism. The message embedding procedure consists of mainly 3 steps. They are synthetic image and non overlapping block creation, Header block identification, Data block creation. First step is the synthetic image and non overlapping block creation. It is created in the following manner. Create synthetic image by using the

input source texture. Synthesized texture image appears larger than the input texture but have similar visual appearance. After creating the synthesized texture image next step is the division of the synthetic image. So we divide the synthetic image into non overlapping blocks. In this paper, created synthetic image is divided into 8X8 non overlapping blocks.

Next step is the header block identification. For finding the header block initially read the message to be embedded and the data hiding key. One of the numbers from the list serves as the header block in the synthetic image. The main function of header block is to store the location of the block where the data is being embedded. Final step in the message embedding procedure is data block creation. Data blocks are created by considering the pixel values of the header blocks placed in the synthetic image. Pixel values of header block are arranged in a vector form. These pixel values serve as the location of the data block. To avoid overwriting on the same data block check the presence of similar pixel value and if exist ignore it. We can embed data into these data block.

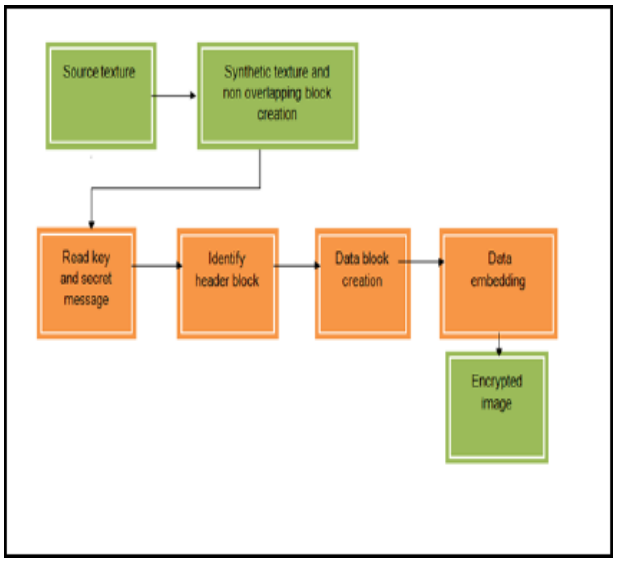

Fig 3: Message Embedding Procedure

4. using the key, generate random numbers, one of which will serve as the header block H
5. Calculate length, len of M.
6. Using len determine the number of data blocks
Dn, n=1, 2....Number of Blocks - Number of header blocks
7. Embed data into the blocks

B. Data Extraction Mechanism and Data Authentication Procedure

Data extraction involves the inverse process of data embedding mechanism. Encrypted image reaches the receiver side. First step involved in the data extraction is the use of secret data hiding key used in the embedding mechanism. Here the same secret key is used for embedding and extraction mechanism. By getting the secret key we can identify the header block. When we get the header block we can also identify the data blocks. Then we can extract the embedded data. Data authentication is carried out to check that the message or data has not been modified during their transmission. Data authentication procedure helps the receiver to verify the origin of the message.

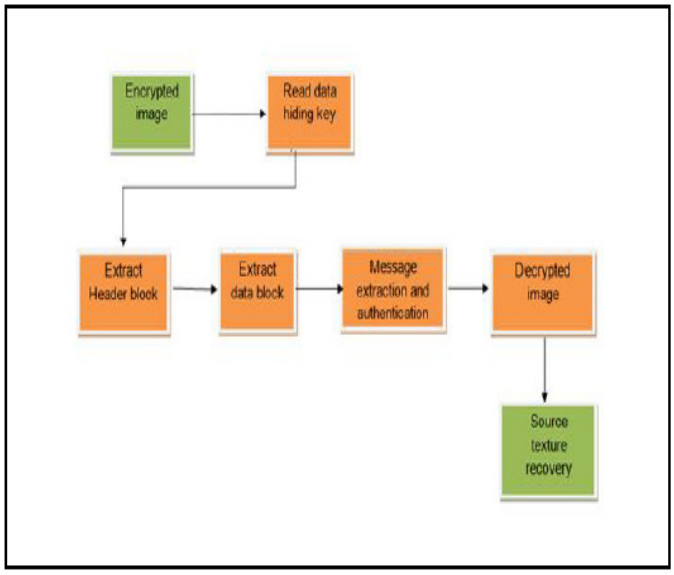

Fig 4: Message Extracting Procedure

Algorithm for Data Hiding (I, K, M)

1. Load the Source Texture Image, I
2. Create the synthetic image and divide the image into number of non overlapping blocks of size 8 x 8.
3. Input key, K and message M

Algorithm for Data Extraction (E, K, M)

1. Load the Encrypted image, E
2. Divide E into number of non overlapping blocks of size 8 x 8.
3. Using the same procedure of data hiding identifies the header and data blocks.

4. De embed data from the data blocks

C. Source Texture Recovery

After the extraction of data we get the original synthetic image. From the original synthetic image we can recover the source texture which will be exactly the same as the input source texture.

V. CONCLUSION

An approach for steganographic algorithm based on reversible capability is described in this paper. This paper describes an innovative method of steganography using reversible texture synthesis. Texture synthesis mechanism means creating or forming a large texture from the input texture. Synthesized texture appears larger than the input texture but have similar visual appearance. In order to conceal secret messages we combine the texture synthesis process into steganography which hides the source texture and also it hides the embedded secret messages. Proposed system offers increased embedding capacity and security than the existing system. Proposed method can produce visually plausible stego synthetic textures.

REFERENCES

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

[2] Fang, C-W. & Lien, J. J , Fast Image Replacement Using Multi- resolution Approach, Springer, pp. **509- 520, 2006.**

[3] H. Otori and S. Kuriyama , Data- embeddable texture synthesis, in Proc. 8th Int. Symp. Smart Graph, Kyoto, Japan, **2007**, pp. **146–157.**

[4] K. xoe Feature-aligned shape texturing, ACM Trans. Graph.,vol. **28**, no. **5, 2009,** Art. ID.

[5] 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.**

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

[7] Lexing Ying, Aaron Hertzmann, Henning Biermann, Denis Zorin, Texture and Shape Synthesis on Surfaces, Springer-Verlag London .

[8] 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.**

[9] 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.**

[10] A. S. Sonawane, M. L. Dhore , S. N. Mali Information Hiding and Recovery using Reversible Embedding , International Journal of Computer Applications Volume **24**– No.**5**, June **2011.**

[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.**

[12] 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**, Jun. **2013** .

[13] Kuo-Chen Wu and Chung-Ming Wang, Member, IEEE,S Steganography Using Reversible Texture Synthesis IEEE Transactions on Image processing Volume **24**,January **2015.**

