INTERNATIONAL JOURNAL OF SCIENTIFIC PROGRESS AND RESEARCH (IJSPR) ISSN: 2349-4689

Steganography Using Reversible Texture Synthesis

[1]Nischitha N, [2]Prof. K.M Sowmya Shree

[1]PG Scholar (MCA), Dept. of MCA, P.E.S.College Engineering, Mandya Karnataka, India

[2]Assistant Professor, Dept. of MCA, P.E.S.College Engineering, Mandya, Karnataka, India

Abstract:We propose a novel approach for Steganography using a reversible texture synthesis. A texture synthesis process Resample a smaller texture image, which synthesizes a new texture image with a similar local appearance and an arbitrary size. We weave the texture synthesis process into Steganography to conceal secret messages. In contrast to using an existing cover image to hide messages, our algorithm conceals the source texture image and embeds secret messages through the process of texture synthesis. This allows us to extract the secret messages and source texture from a stego synthetic texture. Our approach offers three distinct advantages. First, our scheme offers the embedding capacity that is proportional to the size of the stego texture image. Second, a steganalytic algorithm is not likely to defeat our Steganography approach. Third, the reversible capability inherited from our scheme provides functionality, which allows recovery of the source texture. Experimental results have verified that our proposed algorithm can provide various numbers of embedding capacities, produce a visually plausible texture images, and recover the source texture.

Keywords : Data embedding, example-based approach, Reversible, Steganography, texture synthesis.

I. INTRODUCTION

In the last decade many advances have been made in the area of digital media, and much concern has arisen regarding Steganography for digital media. Steganography is a singular method of information hiding techniques. It embeds messages into a host medium in order to conceal secret messages so as not to arouse suspicion by an eavesdropper .A typical Steganography application includes covert communications between two parties whose existence is unknown to a possible attacker and whose success depends on detecting the existence of this communication . In general, the host medium used in Steganography includes meaningful digital media such as digital image, text, audio, video, 3D model ,etc. A large number of image Steganography algorithms have been investigated with the increasing popularity and use of digital images . Most image Steganography algorithms adopt an existing image as a cover medium.

The expense of embedding secret messages into this cover image is the image distortion encountered in the stego image. This leads to two drawbacks. First, since the size of the cover image is fixed, the more secret messages which are embedded allow for more image distortion. Consequently, a compromise must be reached between the embedding capacity and the image quality which results in the limited capacity provided in any specific cover image.

Recall that image steganalytic is an approach used to detect secret messages hidden in the stego image. A stego image contains some distortion, and regardless of how minute it is, this will interfere with the natural features of the cover image. This leads to the second drawback because it is still possible that an image steganalytic algorithm can defeat the image Steganography and thus reveal that a hidden message is being conveyed in a stego image.

In this paper, we propose a novel approach for Steganography using reversible texture synthesis. A texture synthesis process re-samples a small texture image drawn by an artist or captured in a photograph in order to synthesize a new texture image with a similar local appearance and arbitrary size.

We weave 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 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. To the best of our knowledge, steganography taking advantage of the reversibility has ever been presented within the literature of texture synthesis.

Our approach offers three advantages. First, since the texture synthesis can synthesize an arbitrary size of texture images, the embedding capacity which our scheme offers is proportional to the size of the stego texture image. Secondly, a steganalytic algorithm is not likely to defeat this steganographic approach since the stego texture image is composed of a source texture rather than by modifying the existing image contents. Third, the reversible capability inherited from our scheme provides functionality to recover the source texture. Since the recovered source texture is exactly the same as the original source texture, it can be employed to proceed onto the second round of secret messages for steganography if needed. Experimental results have verified that our proposed algorithm can provide various numbers of embedding capacities, produce visually plausible texture images, and recover the source texture. Theoretical analysis indicates that there is an insignificant probability of breaking down our Steganography approach, and the scheme can resist an RS steganalysis attack .

II .SYSTEM MODEL

Texture synthesis has received a lot of attention recently in computer vision and computer graphics . The most recent work has focused on texture synthesis by example, in which a

source texture image is re-sampled using either pixel-based or patch-based algorithms to produce a new synthesized texture image with similar local appearance and arbitrary size.

Pixel-based algorithms generate the synthesized image pixel by pixel and use spatial neighborhood comparisons to choose the most similar pixel in a sample texture as the output pixel. Since each output pixel is determined by the already synthesized pixels, any wrongly synthesized pixels during the process influence the rest of the result causing propagation errors.

Earlier the secret messages to be used were encoded into colored dotted patterns and then they were embedded behind an image that was a blank image. To implement this pixel-based algorithm was used with the help of pixel-based texture synthesis method that works with the existence of dotted patterns. However, using pixel based algorithm had a small error rate of the message extraction. Hence patch based algorithm has been applied to remove this disadvantage. For image hiding in steganography texture synthesis “patch based algorithm” is being used instead of pixel based algorithm. A patch denotes an image size of a source texture where its size can be specified by the user. It can be represented by its width (Pw) and height (Ph). It basically consist of two parts i.e central part and an outer part where the central part also known as the kernel region with size of Kw × Kh, and the part surrounding the central region is referred to as the boundary region with the depth (Pd). Kernel block consist of source texture with the size of Sw × Sh. Here the source texture can be further divided into different number of nonoverlapped kernel blocks, each having size of Kw × Kh, where KB represent the collection of all kernel blocks thus generated, and ||KB|| denotes the number of elements in the given set. Indexing is implemented for each source patch kbi, i.e., KB = {kbi| i =0 to ||KB||-1}. Here Index Table Generation Process is used in which an index table maintains the record of location of the entire source patch. This leads to easily recognize the synthetic texture and hence retrieve the source texture completely and easily. This is one of the major advantages of using indexing in our algorithm.

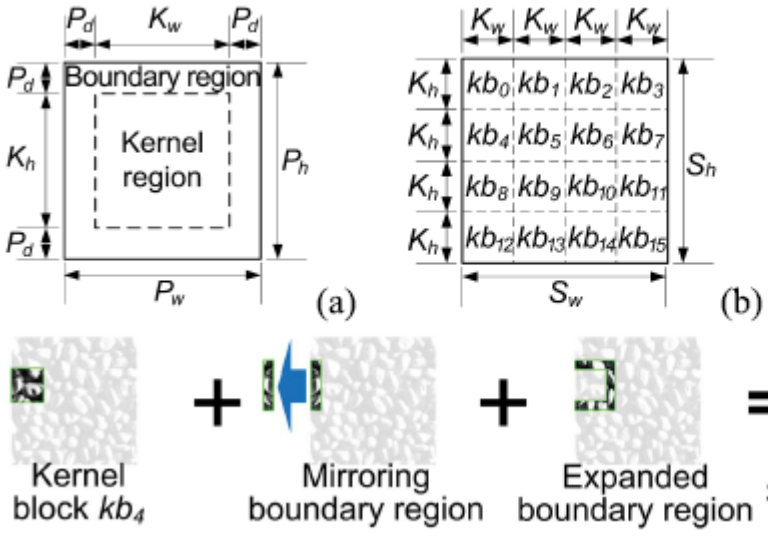

Fig1.- Patch, kernel blocks, and source patch. (a) The diagram of a patch. The central part of a patch is the kernel region; the other part around the kernel region is the boundary region. (b) An illustration of non-overlapped kernel blocks subdivided from the source texture. (c) The diagram of source patches derived by the expanding process using kernel blocks. (d) The boundary mirroring and expanding for a source patch.

III. PREVIOUS WORK

Most image steganographic algorithms adopt an existing image as a cover medium. The expense of embedding secret messages into this cover image is the image distortion encountered in the stego image.The most recent work has focused on texture synthesis by example, in which a source texture image is re-sampled using either pixel-based or patch-based algorithms to produce a new synthesized texture image with similar local appearance and arbitrary size.Otori and Kuriyama pioneered the work of combining data coding with pixel-based texture synthesis. Secret messages to be concealed are encoded into colored dotted patterns and they are directly painted on a blank image.

IV. PROPOSED METHODOLOGY

In this paper, we propose a novel approach for steganography using reversible texture synthesis. A texture synthesis process re-samples a small texture image drawn by an artist or captured in a photograph in order to synthesize a new texture image with a similar local appearance and arbitrary size.

We weave 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 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 three fundamental differences between our proposed message-oriented texture synthesis and the conventional patch based texture synthesis are described in following: The first difference is the shape of the overlapped area. During the conventional synthesis process, an L-shape overlapped area is normally used to determine the similarity of every candidate patch. In contrast, the shape of the overlapped area in our algorithm varies because we have pasted source patches into the workbench. Consequently, our algorithm needs to provide more flexibility in order to cope with a number of variable shapes formed by the overlapped area.

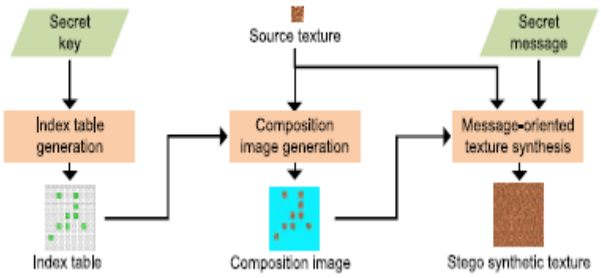

Fig2 The flowchart of the three-process message embedding procedure.

V.SIMULATION/EXPERIMENTAL RESULTS

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.

TABLE 1. Total embedding capacity in bits our algorithm can provide

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

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

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

VI. CONCLUSION

This paper proposes a reversible steganographic algorithm using texture synthesis. Given an original source texture, our scheme can produce a large stego synthetic texture concealing secret messages. To the best of our knowledge, we are the first that can exquisitely weave the steganography into a conventional patch-based texture synthesis. Our 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. With the two techniques we have introduced, our algorithm can produce visually plausible stego synthetic textures even if the secret messages consisting of bit “0” or “1” have an uneven appearance of probabilities. The presented algorithm is secure and robust against an RS steganalysis attack. We believe our proposed scheme offers substantial benefits and provides an opportunity to extend steganographic applications. One possible future study is to expand our scheme to support other kinds of texture synthesis approaches to improve the image quality of the synthetic textures. Another possible study would be to combine other steganography approaches to increase the embedding capacities.

. FUTURE SCOPES

One possible future study is to expand our scheme to support other kinds of texture synthesis approaches to improve the image quality of the synthetic textures. Another possible study would be to combine other Steganography approaches to increase the embedding capacities..

REFERENCES

[1] N. F. Johnson and S. Jajodia, “Exploring steganography: Seeing the unseen,” Computer, vol. 31, no. 2, pp. 26–34, 1998.

[2] N. Provos and P. Honeyman, “Hide and seek: An introduction to steganography,” IEEE Security Privacy, vol. 1, no. 3, pp. 32–44, May/Jun. 2003.

[3] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, “Information hiding survey,” Proc. IEEE, vol. 87, no. 7, pp. 1062–1078, Jul. 1999.

[4] Y.-M. Cheng and C.-M. Wang, “A high-capacity steganographic approach for 3D polygonal meshes,” Vis. Comput., vol. 22, nos. 9–11, pp. 845–855, 2006.

[5] S.-C. Liu and W.-H. Tsai, “Line-based cubism-like image—A new type of art image and its application to lossless data hiding,” IEEE Trans. Inf. Forensics Security, vol. 7, no. 5, pp. 1448–1458, Oct. 2012.

[6] I.-C. Dragoi and D. Coltuc, “Local-prediction-based difference expansion reversible watermarking,” IEEE Trans. Image Process., vol. 23, no. 4, pp. 1779–1790, Apr. 2014.

[7] J. Fridrich, M. Goljan, and R. Du, “Detecting LSB steganography in color, and gray-scale images,” IEEE MultiMedia, vol. 8, no. 4, pp. 22–28, Oct./Dec. 2001.

[8] 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, Oct. 2013.

[9] 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.

[10] 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.