Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 5.258

IJCSMC, Vol. 5, Issue. 3, March 2016, pg.388 – 394

REVERSIBLE PATCH BASED TEXTURE SYNTHESIS FROM A STEGOSYNTHETIC SOURCE IMAGE

S.Sheerin Uzaima, M.O.Ramkumar, M.E

Department of CSE, IFET College of Engineering, Villupuram

sherinshoukathali@gmail.com, ramkumar.mo86@gmail.com

ABSTRACT: We propose a novel approach for steganography using a reversible texture synthesis from a stegosynthetic source image. This involves resampling of the original source image in order to hide the existence of secret message. We weave the texture synthesis process into steganography to conceal secret messages without leaving the chance for hacking. This allows us to extract the secret messages and source texture from a stego synthetic texture. This provides substantial benefits and provides an opportunity to extend various steganographic applications. Generally, Steganography is the science of hiding the existence of messages within images. In our paper, a patch based sampling is used to hide the existence of secret messages. From a stego synthetic source image, splitted into patches, secret messages are hidden in any of the patches, typically hard for unauthorized users to conceal the hidden message. The secret key is revealed only to the dedicated group of registered users, using a reversible steganographic algorithm. The source texture can be recovered using our proposed algorithm.

KEYWORDS: Reversible steganography, message-oriented, texture-synthesis, stegosynthetic texture, index table

I. INTRODUCTION

Steganography is the practice of concealing a file, message, image or a video within another file, message, image or a video. It is a science of hiding informations, necessary for secure communication over an insecure network or channel ,namely privacy, confidentiality, key exchange, authentication and non repudiation.The goal of cryptography is to make data unreadable by a third party. There are many ways for hiding information such as using a covert medium, hiding texts within web pages, using ciphers etc..,One of the most widely used method is Digital Watermarking. In our paper we use retrieving the secret information from a source image. 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, 3Dmodel etc.

A texture synthesis process re-samples a smaller texture image which synthesizes a new texture image with a similar local appearance.

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. It involves hiding the information within infinite number of patches. A typical steganographic 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. 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. No significant visual difference exists between the two stego synthetic textures and the pure synthetic texture.

II. RELATED WORKS

Texture analysis: how to capture the essence of texture? Need to model the whole spectrum: from repeated to stochastic texture. This problem is at intersection of vision, graphics, statistics, and image compression .Some Previous Work–multi-scale filter response histogram matching[11] [Heeger and Bergen,'95]–sampling from conditional distribution over multiple scales [DeBonet,'97]–filter histograms with Gibbs sampling [Zhu et al,'98]–matching 1st and 2nd order properties of wavelet coefficients [Simoncelli and Portilla,'98]–N-gram language model [Shannon,'48]–clustering pixel neighborhood densities [Popat and Picard,'93].Our Approach.[1]Our goals:–preserve local structure–model wide range of real textures–ability to do constrained synthesis. Our method:-Texture is "grown" one pixel at a time–conditional pdf of pixel given its neighbors synthesized thus far is computed directly from the sample image A Novel method of Steganography to achieve Reversible Data Hiding (RDH)[10] is proposed using Histogram Modification (HM). In paper the HM technique is revisited and a general framework to construct HM-based RDH is presented by simply designing the shifting and embedding functions on the cover image. The Secret Image is embedded inside the cover image using several steps of specific shifting of pixels with an order. The secret image or logo is retrieved without any loss in data on the cover and as well as in the secrete image. The Experimental results show the better Peak Signal to Noise Ratio (PSNR) with the existing methods. This paper investigates the effectiveness of prediction-error expansion reversible watermarking on textured images[6].Five well performing reversible watermarking schemes are considered, namely the schemes based on the rhombus average, the adaptive rhombus predictor, the full context predictor as a weighted average between the rhombus and the four diagonal neighbors, the global least-squares predictor and its recently proposed local counterpart. The textured images are analyzed and the optimal prediction scheme for each texture type is determined. The local least-squares prediction based scheme provides the best overall results.

III. PROPOSED WORK

We propose a novel approach for steganography using reversible texture synthesis A texture synthesis process re-samples a small texture image and provides an image of arbitrary size and shape, which holds the hidden message. It involves generation of an index table with a desirable number of rows and columns. The index table is split into given number of rows and columns and the secret message is hidden in any of the column or row according to the sender's choice. Then the image is merged into its original form with a merging file name. The image will only be sent to the list of available users, that is the users with valid username and registered users. At the receiver end, the receiver, for whom the message is intended will receive it in their inbox. As the splitting procedure is only revealed to the registered user, they split the merged image using secret key and read the message. First, we will define some basic terminology to be used in our algorithm. The basic unit used for our steganographic texture synthesis is referred to as a "patch." We illustrate our proposed method in this section. First, we will define some basic terminology to be used in our algorithm.

IV. ALGORITHM

Reversible-steganographic algorithm

First, our scheme offers the embedding capacity that is proportional to the size of the stegotexture image. Second, a steganalytic algorithm is not likely to defeat our steganographic 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.

USER INTERFACE DESIGN:

This is the first module of our project. The important role for the user is to move login window to data owner window. This module has created for the security purpose. In this login page we have to enter login user id and password. It will check username and password is match or not (valid user id and valid password). If we enter any invalid username or password we can't enter into login window to user window it will shows error message. So we are preventing from unauthorized user entering into the login window to user window. It will provide a good security for our project. So server contain user id and password server also check the authentication of the user. It well improves the security and preventing from unauthorized data owner enters into the network. In our project we are using SWING for creating design. Here we validate the login user and server authentication.

INDEX TABLE GENERATION

The first process is the index table generation where we produce an index table to record the location of the source patch set SP in the synthetic texture. The index table allows us to access the synthetic texture and retrieve the source texture completely. We first determine the dimensions of the index table (Tpw × Tph). Given the parameters Tw and Th, which are the width and the height of the synthetic texture we intend to synthesize, the number of entries in this index table can be determined where TPn denotes the number of patches in the stego synthetic texture. For simplicity, we chose appropriate parameters for Tw, Th, Pw, Ph, and Pd, so that the number of entries is an integer. As an example, if Tw × Th =488 × 488, Pw × Ph = 48 × 48, and Pd = 8, then we can generate an index table (12 × 12) containing 144 entries.

COMPOSITE IMAGE GENERATION

The second process of our algorithm is to paste the source patches into a workbench to produce a composition image. First, we establish a blank image as our workbench where the size of the workbench is equal to the synthetic texture. By referring to the source patchIDs stored in the index table, we then paste the source patches into the workbench. During the pasting process, if no overlapping of the source patches is encountered, we paste the source patches directly into the workbench. However, if pasting locations cause the source patches to overlap each other, we employ the image quilting technique to reduce the visual artifact on the overlapped area.

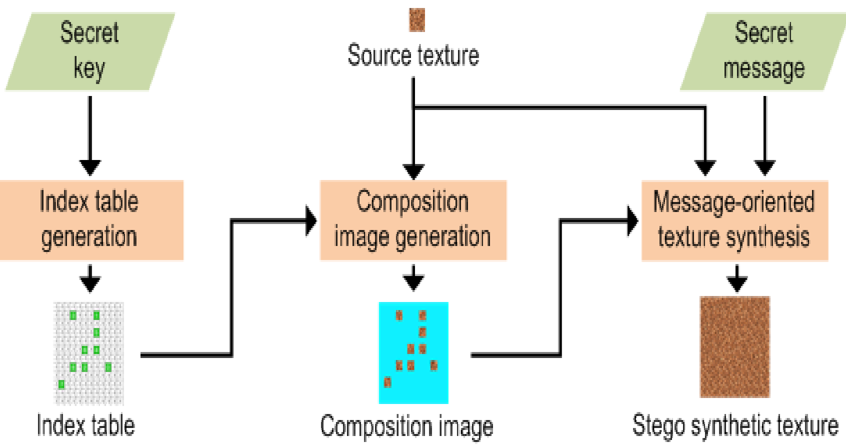
MESSAGEORIENTED TEXTURE SYNTHESIS

We have now generated an index table and a composition image, and have pasted source patches directly into the workbench. We will embed our secret message via the message-oriented texture synthesis to produce the final stego synthetic texture. The three fundamental differences between our proposed message-oriented texture synthesis and the conventional patch based texture synthesis are described. The first difference is the shape of the overlapped area.

RECOVERY AND AUTHENTICATION PROCEDURE

The message extracting for the receiver side involves generating the index table, retrieving the source texture, performing the texture synthesis, and extracting and authenticating the secret message concealed in the stego synthetic texture. The extracting procedure contains four steps. Given the secret key held in the receiver side, the same index table as the embedding procedure can be generated. The next step is the source texture recovery. Each kernel region with the size of $Kw \times Kh$ and its corresponding order with respect to the size of $Sw \times Sh$ source texture can be retrieved by referring to the index table with the dimensions $Tpw \times Tph$. We can then arrange kernel blocks based on their order, thus retrieving there covered source texture which will be exactly the same as the source texture.

V. SYSTEM ARCHITECTURE



VI. EXPERIMENTAL RESULTS

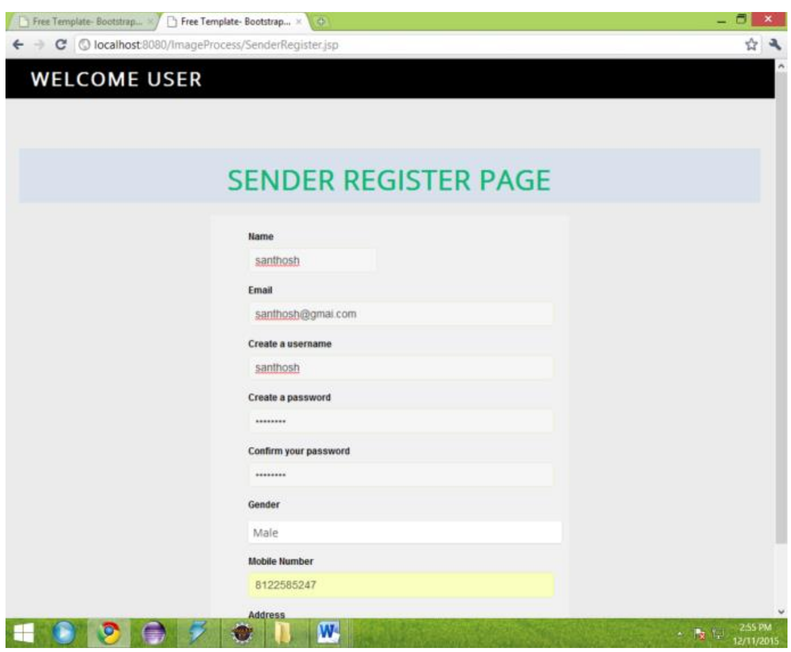


FIG1. User interface module

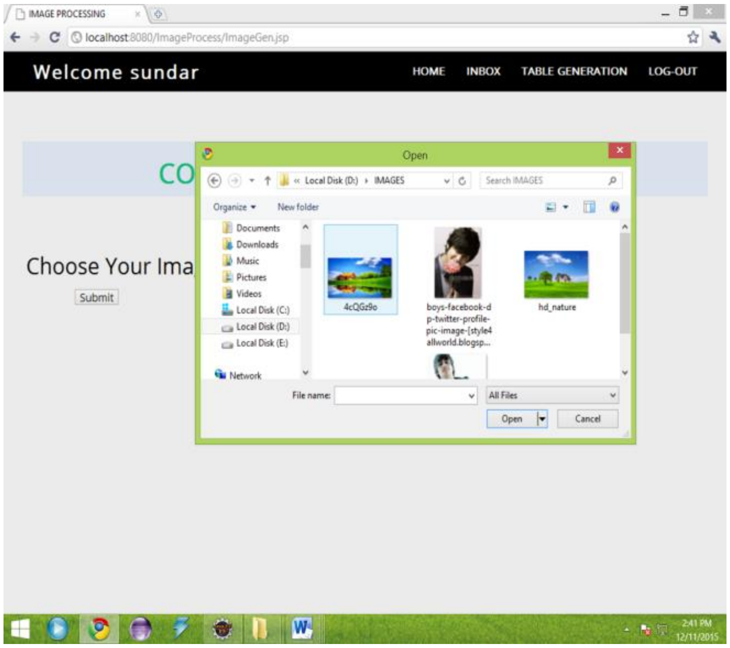


FIG2. Composite Image Generation

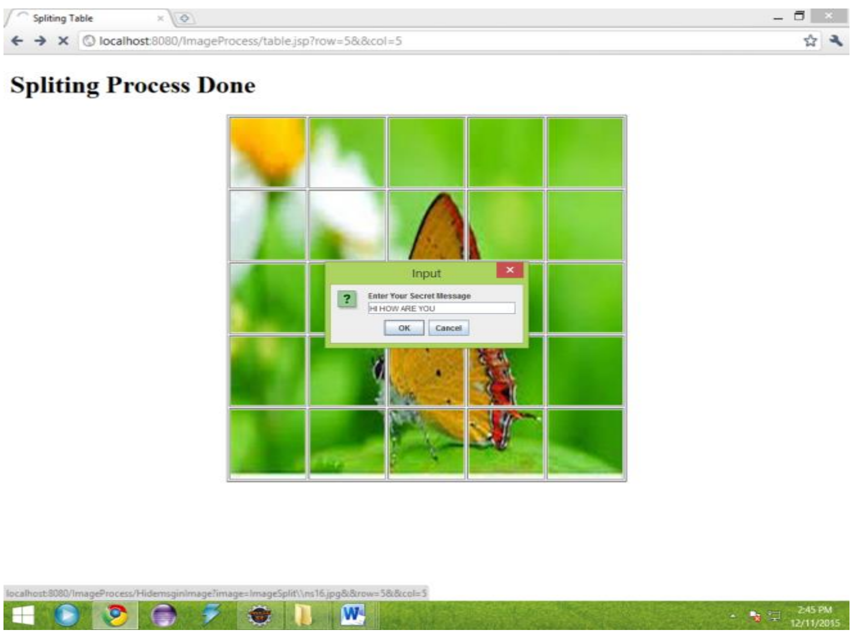


FIG3. Image splitting

FIG4.Image Merging

VII. CONCLUSION

This paper proposes a reversible steganographic algorithm which involves hiding of textures within a stego source image. Our method is efficient among various methods of image hiding, as the composite image is generated and splitted, within which the secret information is hidden and the image is merged after hiding the information. The authentication and recovery procedure is also simple and cannot be affected by an RS attack. Our algorithm provides reversibility to retrieve the original source texture.

VIII. FUTURE ENHANCEMENT

Future study is to expand our scheme to support other kinds of texture synthesis approaches to improve the image quality of the synthetic textures and to apply this algorithm to images of any size. Another possible study would be to combine other steganography approaches with the embedding technologies.

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. Honey man, "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 steganographicapproach 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," IEEETrans. 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-parametricsampling," in Proc. 7th IEEE Int. Conf. Comput. Vis., Sep. 1999,pp. 1033–1038.
[11] C. Han, E. Risser, R. Ramamoorthi, and E. Grinspun, "Multiscaletexturesynthesis," ACM Trans. Graph., vol. 27, no. 3, 2008, Art. ID 51.
[12] H. Otori and S. Kuriyama, "Data-embeddable texture synthesis," in Proc.8th Int. Symp. Smart Graph. Kyoto, Japan, 2007, pp. 146–157.
[13]Suhasni,M.O.Ramkumar,Image Re-Ranking for websearch al,International Journal of Computer Science and Mobile Computing, Vol.4 Issue.3, March-2015

