JOURNAL OF CRITICAL REVIEWS

ISSN- 2394-5125 VOL 06, ISSUE 05, 2019

DIGITAL DATA HIDING USING TURTLE SEGMENTATION

G. Kingsley[1], M. Mailsamy[2], G. Ayappan[3]
Department of Electronics and Communication,
CMS College of Engineering and Technology,
Coimbatore, India.

ABSTRACT: A new data hiding scheme based on turtle shells to further improve the embedding capacity with high visual quality of stego image. The proposed scheme can be divided into embedding and extracting procedures.To further improve security, the encryption key K,is used.The encryption key K is a secret key that should be shared between the sender and the receiver in advance.

INTRODUCTION

Image processing involves changing the nature of an image in order to either improve its pictorial information for human interpretation, render it more suitable for autonomous machine perception. This shall be concerned with digital image processing, which involves using a computer to change the nature of a digital image. It is necessary to realize that these two aspects represent twoseparate but equally important aspects of image processing. A procedure which satisfies a condition makes an image " look better "may be the very worst procedure for satisfying condition . Humans like their images to be sharp, clear and detailed; machines prefer their images to be simple and uncluttered.

Digital Image Processing

Digital image processing helps us enhance images to make them visually pleasing accentuate regions or features of an image to better represent the content distortion during processing. Since images are defined over two dimension. Digital image processing isthe use of computer algorithms to perform image processing on digital images. As a subcategory or field of digital signal processing, digital image processing has many advantages over analog image processing. It allows a much wider range of algorithms to be applied to the input data and can avoid problems such as the build-up of noise and signal distortions during processing. Digital image processing may be modeled in the formof multidimensional systems.

DATA HIDING

DATA HIDING is referred to as a process to hide data (representing some information) into cover media. That is, the datahiding process links two sets of data, a set of the embedded data andanother set of the cover media data. The relationship between thesetwo sets of data characterizes different applications. For instance, incovert communications, the hidden data may often be irrelevant to thecover media. In authentication, however, the embedded

ISSN- 2394-5125 VOL 06, ISSUE 05, 2019

data areclosely related to the cover media. In these two types of applications,invisibility of hidden data is an important requirement. In most casesof data hiding, the cover media will experience some distortion due to data hiding and cannot be inverted back to the original media.

That is, some permanent distortion has occurred to the covermedia even after the hidden data have been extracted out. In some applications, such as medical diagnosis and law enforcement, it is critical to reverse the marked media back to the original cover media after the hidden data are retrieved for some legal considerations. In other applications, such as remote sensing and high-energy particle physical experimental investigation, it is also desired that the original cover media can be recovered because of the required high-precision nature. The marking techniques satisfying this requirement are referred to as *reversible, lossless, distortion-free*, or *invertible* data hiding techniques. Reversible data hiding facilitates immense possibility of applications to link two sets of data in such a way that the cover media can be losslessly recovered after the hidden data havebeen extracted out, thus providing an additional avenue of handling two different sets of data.

Data hiding in an image involves embedding a large amount of secret information into a cover image with minimal perceptible degradation of image quality. However, the hiding capacity for secret data and the distortion of the cover image are a tradeoff since more hidden data always results in more degradation on the visual quality of the cover image. Moreover, when data hiding is implemented onthe compressed domain of image, the hiding capacity and the visual quality of cover images can be further restricted. During the lastdecade, vector quantization (VQ) has emerged as an efficient method in image compression. One specific feature of VQ is that high compression ratios are possible with relatively small block sizes.With the rapid development of Internet technology, people can transmit and share digital content with each other conveniently.

In order to guarantee communication efficiency and save network bandwidth, compression techniques can be implemented on digital content to reduce redundancy, and the quality of the decompressed versions should also be preserved. Nowadays, most digital content, especially digital images and videos, are converted into the compressed forms for transmission. Another important issue in an open network environment is how to transmit secret or private data securely. Even though traditional cryptographic methods can encrypt the plaintext into the cipher text , the meaningless random data of the cipher text may also arouse the suspicion from the attacker. To solve this problem, information hiding techniques have been widely developed in academic and industry, which can embed secret data intothe cover data imperceptibly. Due to the prevalence of digital images on the Internet, how to compress images and hide secret data into the compressed images efficiently in-depth study.

ISSN- 2394-5125 VOL 06, ISSUE 05, 2019

DIGITAL TURTLE SEGMENTATION

Step 1. Initialize a population of particles with random positions xi(t) and velocities vi(t) and compute the corresponding fitness values f(xi(t)). Each initial particle is denoted as the local best particle pbest, and the particle that has the best fitness value is denoted as the global best particle gbest.

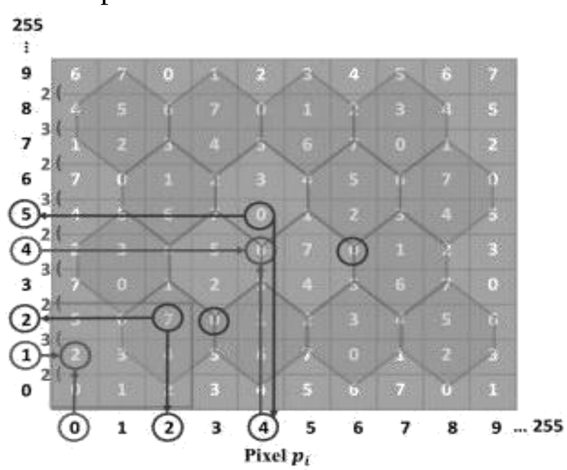
Step 2. Update the position of a particle xi(t + 1) according to its updated velocity vi(t + 1), which are determined by using Equations (1) and (2), respectively.

Step 3. Compute the fitness value of the updated particle f(xi(t + 1)).

Step 4. Compare this new fitness value f(xi(t+ 1)) with f(pbest). If f(xi(t + 1)) < f(pbest), pbest is replaced by xi(t + 1). Otherwise, pbest remains unchanged.

Step 5. Compare the current fitness value f(xi(t + 1)) with f(gbest). If f(xi(t + 1)) < f(gbest), gbest is replaced by xi(t + 1). Otherwise, gbest remains unchanged.

Step 6. If the terminal condition is met, then the best particle gbest and its fitness value are output. Otherwise, go back to Step 2.



Digital Data Hiding life-cycle phases

In the turtle-shell matrix, there are eight integers from 0 to 7 in a turtle shell or a 3 × 3 sub-block due to the architectural property of the matrix. In order to minimize the distortion of the cover image, it isimportant to determine the best table to replace the original eight integers. An example of substitution tables is shown in Figure 3. There is a total of 40,320 ways to sort these eight integers in total. So, it would take a lot of time to search all of the tables to find the best one. Thus, in this paper, we used PSO to search for the best table of the matrix.

ISSN- 2394-5125 VOL 06, ISSUE 05, 2019

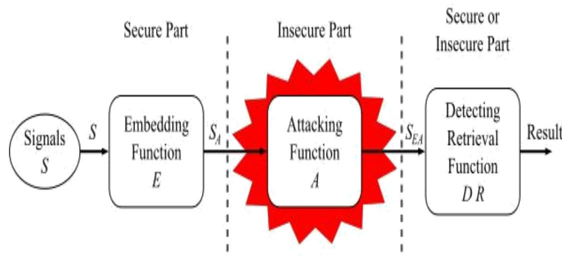


Image compression may be lossy or lossless. Lossless compression is preferred for archival purposes and often for medical imaging, technical drawings, clip art, or comics. Lossy compression methods, especially when used at low bit rates, introduce compression artifacts. Lossy methods are especially suitable for natural imagessuch as photographs in applications where minor (sometimes imperceptible) loss of fidelity is acceptable to achieve a substantial reduction in bit rate. The lossy compression that produces imperceptible differences may be called visually lossless.

Data embedding procedure

In the proposed scheme, the indices in the first row and first column of the index table, are called seed indices and will be kept unchanged and don't carry any secret data. The data embedding procedure embed data in residual indices as follows:

Input: A VQ index table, the secret bit stream, a codebook(n= codebook size, b= Flog2n⌉), and parameter c(d= b- c, r= 2d- 1) Output: The embedded index table

For each index II do:

Step 1: Compute SMD for all codewords in the codebook and thensort the codebook based on SMD.

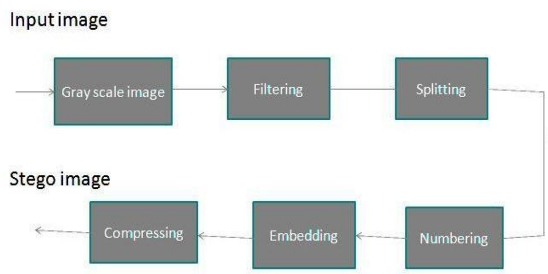
Step 2: Find pi which is the position of Ii in the sorted codebook

. **Step 3**: Embedded index Ii' is obtained based on pi according to thefollowing two cases:

Case 1 (pi< r): Retrieve next c secret bits and compute its decimalvalue s. Then compute pI 'which is the position of the embedded index as follows: pi' =pi× 2c+sk. Then index I i' which is the index located in the position pi' of the sorted codebookis sent to the output as embedded index.

Case 2 (pi≥r): Retrieve next c secret bits and compute its decimal value sk. Then compute pi' which is the position of an index asindicator as follows: pi' = r× 2c+ sk and then send Ii'||Ii to the output, where Ii' is the index located in the position pi' of the sorted codebook and Ii current index and || is the concatenation operation.

ISSN- 2394-5125 VOL 06, ISSUE 05, 2019



The details of the extracting algorithm at the receiver are given as follows. Input: the smooth codebook Y0 = {y00, y01, . . . ,y0 p−1},the complex codebook Y1 = {y10, y11, . . . , y1q−1}, and the receivedbit stream (including side information and VQ indices). Relative parameters: the side-match state code book size r.

Step 1. Let the currently processed (decoded) image block bexi, extract the first bit from the received bit stream.

Step 2. If the extracted bit is '0', then the master codebook for xi is codebook Y0 and examine the next bit in the received bit stream. Otherwise, the master codebook for xi is codebook Y1 and go to step 4.

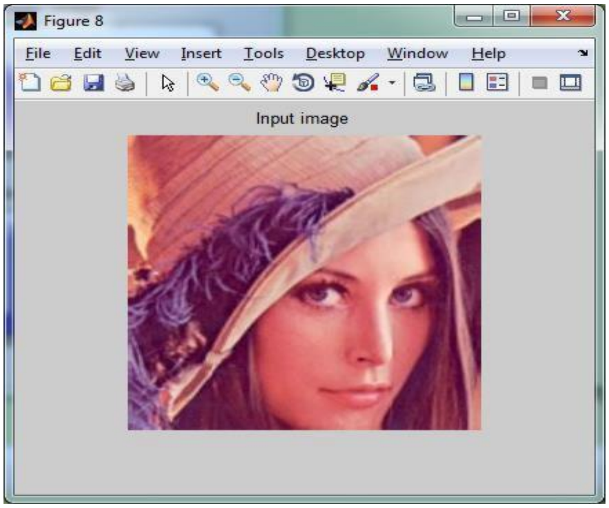
Step 3. If the next bit is '0', remove this next bit and extract thesucceeding log2 p bits from the received bit stream. Use the log2 p bits and codebook Y0 to decode block xi by VQ. Otherwise, remove this next bit and extract the succeeding log2 r bits from the received bit stream. Generate the side matchstate codebook S i from codebook Y0, and decode xi, using the log2 r bits based on SMVQ. Collect the log2 r bits as part of thesecret bit stream.

Step 4. Extract the succeeding log2 q bits from the received bit stream. Use the log2 q bits and codebook Y1 to decode block xiby VQ directly.

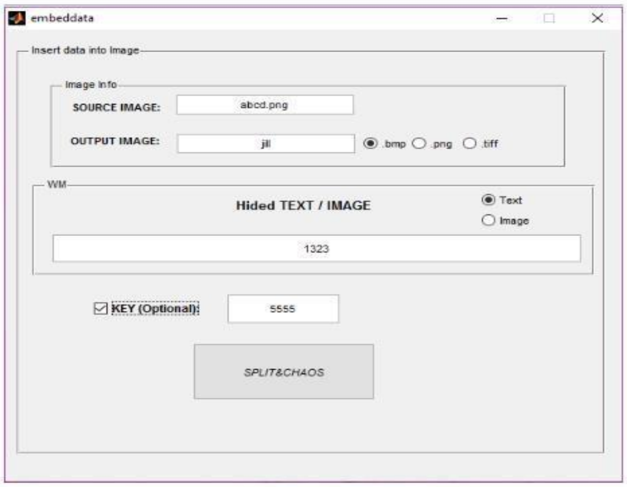
Step 5. If there exists image blocks to be processed, go to step1.

Step 6. Merge the collected bits to obtain the whole secret bitstream.

INPUT IMAGE



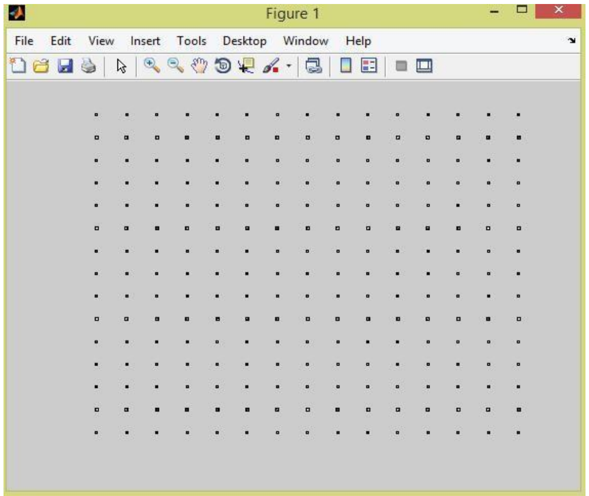
EMBEDDING THE DATA

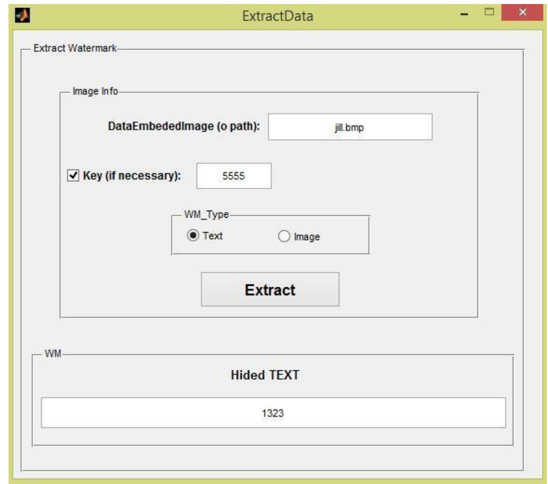


The input image name is entered in the source image. The output image name is entered in the output image. The secret text is entered in the hided text/image. The key is entered and then click splitand chaos.

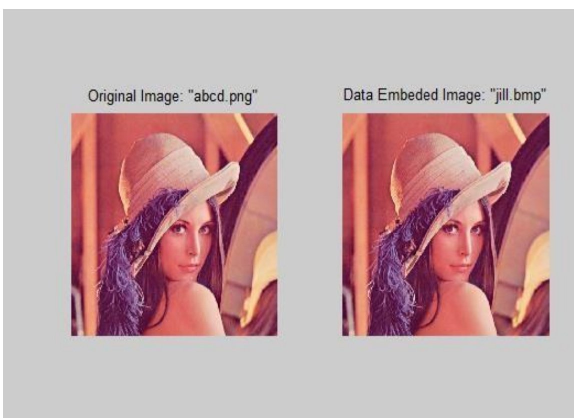
ISSN- 2394-5125 VOL 06, ISSUE 05, 2019

IMAGE SPLITTING





ORIGINAL AND EMBEDDED IMAGE



CONCLUSION

This paper proposed a joint data-hiding and compression scheme by using SMVQ and PDE-based image inpainting. The blocks, except for those in the leftmost and topmost of the image, can be embedded with secret data and compressed simultaneously, and the adopted compression method switches between SMVQ and image inpainting adaptively according to the embedding bits. VQ is also utilized for some complex blocks to control the visual distortion and error diffusion. On the receiver side, after segmenting the compressed codes into a series of sections bythe indicator bits, the embedded secret bits can be easily extracted according to the index values in the segmented sections, and the decompression for all blocks can also be achieved successfully by VQ, SMVQ, and image inpainting. Furthermore, the proposed scheme canintegrate the two functions of data hiding and image compression into a single module seamlessly.

DATA EXTRACTION

REFERENCE

1. Bertalmio. M, Sapiro. G, Caselles.V and Ballester.C (2000) ting" in Proc. 27th Int. Conf. Comput, Graph. Interact. Tech., New Orleans, LA, USA, pp. 417-424.
2. Chan T. F and Shen. J (2001) "Nontexture inpainting by curvature-driven diffusions", J. Vis. Commun. Image Represent., vol. 12,no. 4,pp. 436-449.
3. Chang C. C, Chen T. S and Chung L. Z (2002) "A steganographic method based upon JPEG and quantization table modification", Inf. Sci., vol. 141, no. 1, pp. 123–138.
4. Chang C. C, Chen G. M and Lin M. H (2004) "Information hiding based on search-order coding for VQ indices",Pattern Recognit. Lett., vol. 25, no. 11, pp. 1253–1261.
5. Chang C. C and Wu W. C (2007) "Fast planar-oriented ripple search algorithm for hyperspace VQ codebook",IEEE Trans. Image Process., vol. 16, no. 6, pp. 1538–1547.
6. Chang C. C and Wu W. C (2006) "Hiding secret data adaptively in vector quantisation index tables," IEEE Proc. Vis., Image Signal Process., vol. 153, no. 5, pp. 589–597.
7. Chang C. C, Tai W. L and Lin C. C (2006) "A reversible data hiding scheme based on side match vector quantization," IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 10, pp. 1301–1308.
8. Chen C. C and Chang C. C (2010) "High capacity SMVQ-based hiding scheme using adaptive index," Signal Process., vol. 90, no. 7,pp. 2141–2149.
9. Criminisi. A, Perez . P and Toyama . K (2004) "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1200–1212,.
10. Du W. C and Hsu W. J (2003) "Adaptive data hiding based on VQ compressed images," IEEE Proc. Vis., Image Signal Process., vol.150, no. 4, pp. 233–238.
11. Gersho . A and Gray . M (1992) "Vector Quantization and Signal Compression", Norwell, MA, USA: Kluwer.
12. Hsieh C. H and Tsai J. C (1996) "Lossless compression of VQ index with search-order coding," IEEE Trans. Image Process., vol. 5, no. 11, pp. 1579–1582.
13. Hsieh Y. P, Chang C. C and Liu L. J (2008) "A two-codebook combination and three-phase block matching based image-hiding scheme with high embedding capacity," Pattern Recognit., vol. 41, no.10, pp. 3104–3113.
14. Kim . T (1992) "Side match and overlap match vector quantizersfor images," IEEE Trans. Image Process., vol. 1, no. 2, pp. 170–185.
15. Lee C. C, Ku W. H , and Huang S. Y (2009) "A new steganographic scheme based on vector quantisation and search-ordercoding," IET Image Process., vol. 3, no. 4, pp. 243–248.

ISSN- 2394-5125 VOL 06, ISSUE 05, 2019

16. Lee C. F, Chen H. L and Lai S. H (2010) “An adaptive data hiding scheme with high embedding capacity and visual image quality based on SMVQ prediction through classification codebooks,” Image Vis. Comput., vol. 28, no. 8, pp. 1293–1302.
17. Linde . Y, Buzo . A and Gray R. M (1980) “An algorithm for vector quantization design,” IEEE Trans. Commun., vol. 28, no. 1, pp.84–95.