ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 10, March 2016

Data Hiding in an Encrypted Image Using Huffman Coding Algorithm

V.Manimala[1], J.Mohanapriya[2], V.Leelavathi[2], P.Mohanapriya[2], S.Nandhagopal[2]
Assistant professor[1], Department of Electronics and Communication Engineering,
Erode Builder Educational Trust's Group of Institutions, Kangayam-638108, Tirupur, India
Final Year Students[2], Department of Electronics and Communication Engineering,
Erode Builder Educational Trust's Group of Institutions, Kangayam-638108, Tirupur, India

Abstract:− ***To prevent data from unauthorized access, encryption technique is widely used. This work proposes to hide data into an encrypted image. Reversible data hiding is a type of data hiding techniques whereby the host image can be recovered exactly. In first phase the content owner encrypts the image by reversible data hiding using encryption key. The data hider then hides some information into the encrypted image by Huffman coding algorithm, making use of data hiding key. At the receiver side, if the receiver has only an encryption key, we can get the original image, but cannot read the hidden data. If the receiver has only data hiding key, we can extract the data, but cannot get the original image. If the receiver has both keys, it is also desired that the receiver can extract the hidden data and the original image in a separable manner.***

Keywords*: Reversible data hiding, Image encryption, Huffman coding*.

Introduction:

Existing work:

In an existing system the random key generation method has been used for encryption purpose, there is a chance for easy cracking of the key and hacking the encrypted image. In this system the original image cover is first encrypted, and then secret data are embedded by modifying a small proportion of the encrypted image. The receiver first decrypts the encrypted image, and then extracts the embedded data and recovers the original cover image based on the decrypted version.

Proposed work:

The rapid development of communication technology due to the global spread of the internet and the digital information revolution has given rise to a huge increase in the use and transmission of multimedia information, there are some privacy violations, information theft and so on. Digital images are widely used now a days so its security is very vital. In this work, discrete data hiding in cipher images deals with security and authentication. Content owner encrypts the original uncompressed image using an encryption key. Data hider compresses the least significant bits of the encrypted image using a data hiding key to create a sparse space to accommodate some additional data. In this work, authentication has been done using the Huffman Coding Algorithm. Based on the keys the receiver can extract the hidden data and the original image.

Reversible Data Hiding (RDH):

Data hiding is a group of methods that are used to insert a secure data in a host media(like images). Reversible data hiding inserts information bits by modifying the host data and enable lossless reconstruction of original host data after extracting the embedded information, such as military or medical images. In such a way that the original cover content can be perfectly restored after extraction of the hiding message. In order to securely share a secret image,the content owner can encrypt the image before transmitting it to the receiver. There are different techniques in RDH. Reversible data hiding in image aims to exactly recover both the embedded secret information and the original cover image.

Huffman coding:

The algorithm used in this process for providing security and authentication is Huffman Coding Algorithm. The idea behind Huffman coding is to find a way to compress the storage of

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 10, March 2016

data using variable length codes and to give less frequent characters and groups of characters longer codes. Also, the coding is constructed in such a way that no two constructed codes are prefixes of each other. This property about the code is crucial with respect to easily deciphering the code. For example, each character for Huffman coding we need to create a binary tree for each character that also stores the frequency with which it occurs.

Algorithm steps:

1. Take the characters and their frequencies, and sort this list by increasing frequency.
2. All the characters are vertices of the tree.
3. Take the first 2 vertices from the list and make them children of a vertex having the sum of their frequencies.
4. Insert the new vertex into the sorted list of vertices waiting to be put into the tree.
5. If there are at least 2 vertices in the list, go to step 3.
6. Read the Huffman code from the tree.

Both the .mp3 and .jpg file formats use Huffman coding at one stage of the compression. Reduce size of data by 20%-90% in general. If no characters occur more frequently than others, then no advantage over ASCII. In encoding process based on the characters and their frequencies perform the algorithm and generate a code. Write the characters using the code. And in decoding process the Huffman tree figure out each character(prefix property). A code is called a prefix or free code if no codeword is a prefix of another one. This coding gives less frequent characters and groups of characters longer codes. Also, the coding is constructed in such a way that no two constructed codes are prefixes of each other. This property about the code is crucial with respect to easily deciphering the code. Huffman coding finds the optimal way to take advantage of varying character frequencies in a particular file. On average, using Huffman coding on standard files can shrink them anywhere from 10% to 30% depending to the character distribution.

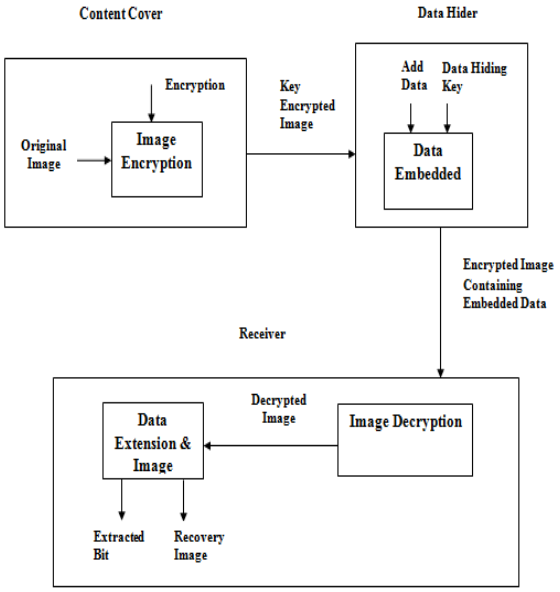


Fig 1. Block diagram

Module Description:

There are three modules,

A. Image Encryption
B. Data Hiding
C. Image Decryption

A. Image Encryption:

Three phases in image encryption:

1) Sparse representation
2) Self-reversible embedding
3) Stream encryption

Given a cover image, we first divide it into patches that are then represented according to an over complete dictionary via sparse coding. Then, the smoother patches with lower residual errors are selected for room reserving. Inorder to reserve room several patches will be selected to construct smoother area. The Room allocation is based on LSB were atmost 3LSB planes will be used $3N^2$ bits for each patch. These selected patches are represented by the sparse coefficients, and the corresponding residual errors are encoded and reversibly embedded into the other non selected paths with a standard RDH algorithm. Splitting the images into patches and training the dictionary completely for reserving room to hide data is called as sparse Finally, the room

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 10, March 2016

preserved and self-embedded image is encrypted to generate the finally version.

B. Data Hiding:

Once the encrypted image is received, the data hider can embed secret data for management or authentication requirement. The embedding process starts with locating the encrypted version of area. Since the image owner has embedded the position of the first room preserving patch and the room size for each patch in the encrypted image, it is effortless for the data hider to know where and how many bits they can modify. After that, the data hider scans each selected patch in the encrypted image, and simply makes use of bit replacement to substitute the corresponding bits reserved for secret data.

C. Image Decryption and Data extraction:

The next step is data extraction process which is the reverse of the data embedding process. Here encrypted data is extracted from the encrypted image in the reverse order by employing the RDH decryption algorithm. Then the Huffman encoded data is retrieved and then Huffman decoding is performed to retrieve the original data. With the encrypted image containing additional embedded data, the receiver faces three situations depending on whether the receiver has data hiding and/or encryption keys. The data extraction and image decryption can be processed separately.

1) Data Extraction with Only Data Hiding Key: For the receiver who only has data hiding key, he can extract the embedded data.

2) Image Decryption with Only Encryption Key: In this case, the receiver has the encryption key only. Then he can decrypt the original image.

3) Data Extraction and Image Recovery with Both Data Hiding and Encryption Keys: If the receiver has both the data hiding key and encryption key, the data extraction and image recovery achieve full reversibility. On the one hand, with the data hiding key, one can extract the hidden secret data without any error.

OUTPUTS:

Image Encryption:

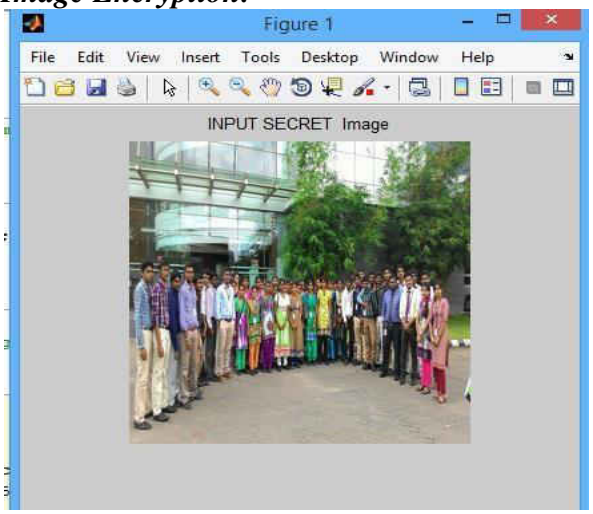


Fig.2 Input image

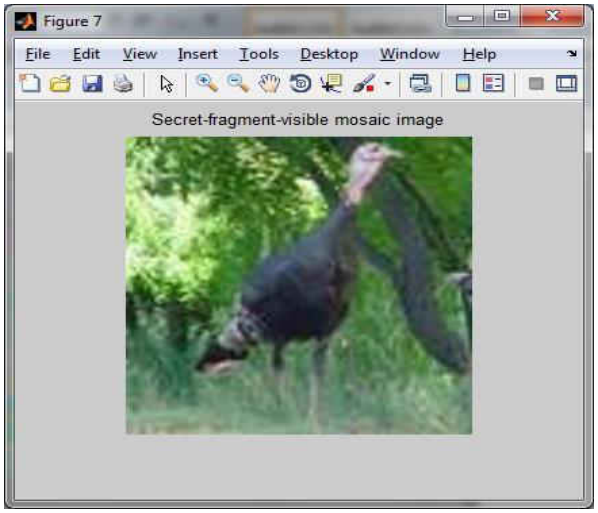
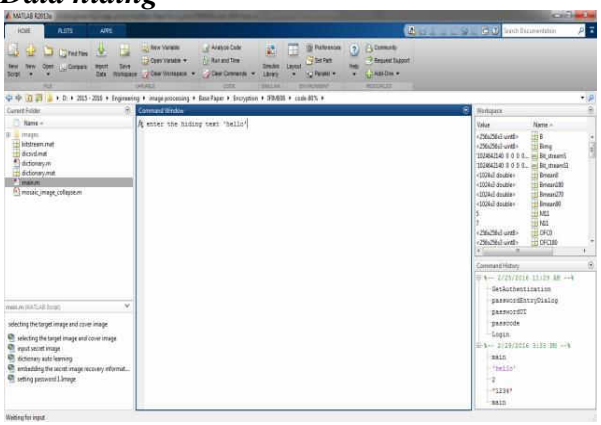


Fig.3 Encrypted Image

Data hiding



ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 10, March 2016

Fig.4 Enter the text to hide

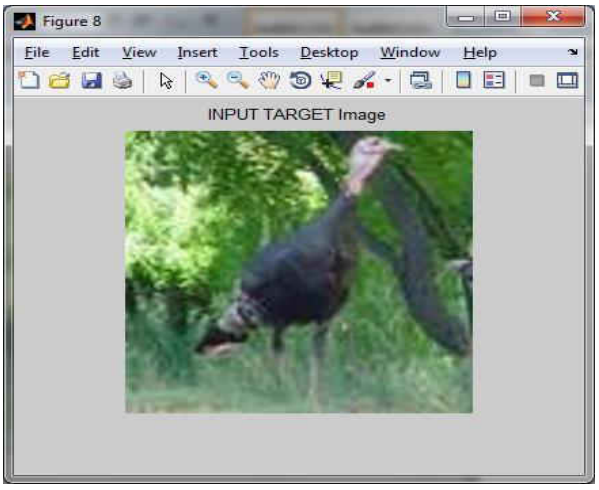


Fig.5 Data hidden image

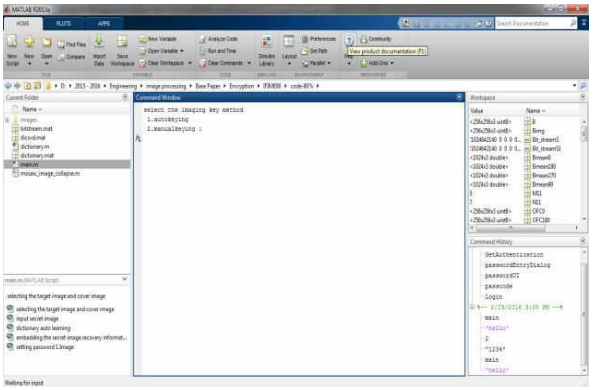


Fig.6 Key Selection

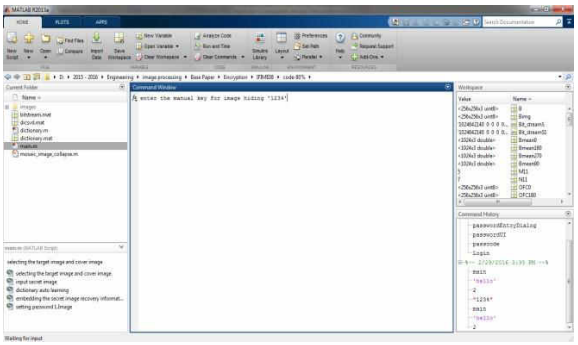


Fig.7 Enter the key value for image

Conclusion:

This work combines data encryption with image encryption. RDH is used to encrypt the image and Huffman coding is to hide the data in it. Huffman coding is more secure to transmit the secret data. The changes could not be find by others. Since the manual keying technique has been employed there is no chance for hacking of the data. The image have to be decrypted and the data will be retrieved as per the key selection.

Reference:

[1] B.ou X.Lt,Y.Zhao,R.N1, and Y-Q. Shi,"pairwise prediction-error expansion for efficient reversible data hiding,"IEEE Irans. Image process, vol.22,no.12,pp.5010-5021,dec.2013.

[2] W.L.Tai, C.M.Yeh and C.C.Chang, "Reversible data hiding based on Histogram modification of pixel differences", IEEEIrans .Circuits syst.Video technology, vol.19,no.6,pp.906-910,jun.2009.

[3] Horng, Y.H. Huang, C.C. Chang and Y.Liu, "(k, n) image reversible data hiding" J. Inf. Hiding Multimedia Signal Process, vol.5, no.2, pp. 152-164, Apr. 2014.

[4] Dabrowski, E.R.Weippl and I. Echizen, "Framework based on pri-vacy policy hiding for preventing unauthorized face image processing" in Proc. IEE Int. Conf. Syst. Man Cybern (SMC), Manchester, U.K, Oct. 2013, pp. 455-461.

[5] X.Zhang, "Reversible Data Hiding in an Encrypted Image" IEEE Signal Process. Lett, vol.18, no. 4,pp. 199-202, Apr. 2012

.

[6] W.Hong, T.S.Chen and H. Wu, "An improved reversible data hiding in an encrypted images using side match" IEEE Signal Process. Lett, vol.19, no. 4,pp. 199-202, Apr. 2012.

ISSN 2395-695X (Print)
ISSN 2395-695X
(Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 10, March 2016