International Journal of Engineering Research in Computer Science and Engineering (IJERCSE) Vol 3, Issue 3, March 2016

Embedded Extended Visual Cryptography Schemes

[1] S. Meghana Reddy, [2] B Rasagna, [3] Prasad B
[1]II/IV, [2][3]Associate Professor
[1][2][3] Department of CSE, Marri Laxman Reddy Institute of Technology and Management (MLRITM)
Hyderabad
[1] meghana1995reddy@gmail.com [2] bheemarasagna@gmail.com [3]bprasad@gmail.com

Abstract **A visual cryptography scheme (VCS) is a kind of secret sharing scheme which allows the encoding of a secret image into shares distributed to participants. The beauty of such a scheme is that a set of qualified participants is able to recover the secret image without any cryptographic knowledge and computation devices. An extended visual cryptography scheme (EVCS) is a kind of VCS which consists of meaningful shares (compared to the random shares of traditional VCS). In this paper, we propose a construction of EVCS which is realized by embedding random shares into meaningful covering shares, and we call it the embedded EVCS. Experimental results compare some of the well-known EVCSs proposed in recent years systematically, and show that the proposed embedded EVCS has competitive visual quality compared with many of the well-known EVCSs in the literature. In addition, it has many specific advantages against these well-known EVCSs, respectively.**

Keywords: **Visual Cryptography Scheme, Data Compression Algorithm, Encoding Algorithm**

I. INTRODUCTION

The basic principle of the visual cryptography scheme (VCS) was first introduced by Naor and Shamir. VCS is a kind of secret sharing scheme that focuses on sharing secret images. The idea of the visual cryptography model proposed in is to split a secret image into two random shares (printed on transparencies) which separately reveals no information about the secret image other than the size of the secret image. The secret image can be reconstructed by stacking the two shares. The underlying operation of this scheme is logical operation OR.

.

a. Project purpose:

Purpose of a VCS with random shares the traditional VCS or simply the VCS. In general, a traditional VCS takes a secret image as input, and outputs shares that satisfy two conditions: 1) any qualified subset of shares can recover the secret image; 2) any forbidden subset of shares cannot obtain any information of the secret image other than the size of the secret image.

b. Project Scope:

System provides a friendly environment to deal with images. Generally tools supports only one kind of image formats. Our application supports .gif and .png (portable network graphics) formatted images and our application has been developed using swing and applet technologies, hence provides a friendly environment to users. Vcs of an evcs, we mean a traditional vcs that have the same access structure with the evcs. Generally, an evcs takes a secret image and original share images as inputs, and outputs shares that satisfy the given three option: any qualified subset of shares can recover the secret image, any forbidden subset of shares cannot obtain any information of the secret image other than the size of the secret image and all the shares are meaningful images.

c. Product Features:

EVCS is flexible in the sense that there exist two trade-offs between the share pixel expansion and the visual quality of the shares and between the secret image pixel expansion and the visual quality of the shares. This flexibility allows the dealer to choose the proper parameters for different applications. Comparisons on the experimental results show that the visual quality of the share of the proposed embedded EVCS is competitive with that of many of the well-known EVCSs in the literature.

II. SYSTEM ANALYSIS

a. Problem definition:

Whenever we transmit the data (image) in the network, any unauthenticated person can read our data (image). In order to provide security to data (image) generally sender will encrypt the data (image) and send it the intended person and the receiver will decrypt the encrypted data(image) and uses it.

b. Existing System:

Visual cryptography is the art and science of encrypting the image in such a way that no-one apart from the sender and intended recipient even realizes the original image, a form of security through obscurity. By contrast, cryptography obscures the original image, but it does not conceal the fact that it is not the actual image.

c. Limitations of Existing System:

The existing system does not provide a friendly environment to encrypt or decrypt the data (images).

d. *Proposed System:*

Proposed system **Visual cryptography** provides a friendly environment to deal with images. Generally cryptography tools supports only one kind of image formats. Our application supports .gif and .png (portable network graphics) formatted images and our application has been developed using swing and applet technologies, hence provides a friendly environment to users.

e. *Advantages of Proposed System:*

EVCS is flexible in the sense that there exist two trade-offs between the share pixel expansion and the visual quality of the shares and between the secret image pixel expansion and the visual quality of the shares. This flexibility allows the dealer to choose the proper parameters for different applications. Comparisons on the experimental results show that the visual quality of the share of the proposed embedded EVCS is competitive with that of many of the well-known EVCSs in the literature.

.

III. SYSTEM MODULES

The system modules include Interface Design Using Applet Frame Work, Visual Cryptography Implementation. Encoding, Decoding, Creating Transparencies, Un-Hiding Image From Transparency, Testing And Integration

a. *Interface design using Applet frame work:*

In this module, we design user interface design using applet frame work. The user interface should be very easy and understandable to every user. So that any one can access using our system. It must be supportable using various GUIs. The user interface also consists of help file. The help file assists on every concepts of the embedded visual cryptography. Help file should clearly depict the details of the project developed in simple language using various screen shoots.

b. *Visual cryptography Implementation:*

This module is the core for the project, where we implement the Visual Cryptography. We used LZW Data Compression algorithm. The LZW data compression algorithm is applied for the gray scale image here. As a pre-processing step, a dictionary is prepared for the gray scale image. In this dictionary, the string replaces characters with single quotes. Calculations are done using dynamic Huffman coding. In compression of greyscale image select the information pixels. Then generate halftone shares using error diffusion method. At last filter process is applied for the output gray scale images. Filters are used to improve the quality of reconstructed image to minimize the noises for sharpening the input secret image.

c. *Encoding:*

A high level view of the encoding algorithm is shown here:

1. Initialize the dictionary to contain all strings of length one.
2. Find the longest string W in the dictionary that matches the current input.
3. Emit the dictionary index for W to output and remove W from the input.
4. Add W followed by the next symbol in the input to the dictionary.
5. Go to Step 2.

A dictionary is initialized to contain the single-character strings corresponding to all the possible input characters (and nothing else except the clear and stop codes if they're being used). The algorithm works by scanning through the input string for successively longer substrings until it finds one that is not in the dictionary. When such a string is found, the index for the string less the last character (i.e., the longest substring that *is* in the dictionary) is retrieved from the dictionary and sent to output, and the new string (including the last character) is added to the dictionary with the next available code. The last input character is then used as the next starting point to scan for substrings.

d. *Decoding:*

The decoding algorithm works by reading a value from the encoded input and outputting the corresponding string from the initialized dictionary. At the same time it obtains the next value from the input, and adds to the dictionary the concatenation of the string just output and the first character of the string obtained by decoding the next input value. The decoder then proceeds to the next input value (which was already read in as the "next value" in the previous pass) and repeats the process until there is no more input, at which point the final input value is decoded without any more additions to the dictionary.

In this way the decoder builds up a dictionary which is identical to that used by the encoder, and uses it to decode subsequent input values. Thus the full dictionary does not need be sent with the encoded data; just the initial dictionary containing the single-character strings is sufficient (and is typically defined beforehand within the encoder and decoder rather than being explicitly sent with the encoded data.)

e. *Creating Transparencies:*

This scheme provides theoretically perfect secrecy. An attacker who obtains either the transparency image or the screen image obtains no information at all about the encoded image since a black-white square on either image is equally likely to encode a clear or dark square in the original image. Another valuable property of visual cryptography is that we can create the second layer after distributing the first layer to produce any image we want. Given a known transparency image, we can select a screen image by choosing the appropriate squares to produce the desired image. One of the most obvious limitations of using visual cryptography in the past was the problem of the decoded image containing an overall gray effect due to the leftover black sub pixel from encoding. This occurred because the decoded image is not an exact preproduction, but an expansion of the original, with extra black pixel. Black pixel in the original document remains black pixel in the decoded version, but White pixel becomes gray. This resulted in a lot of contrast to the entire image. The extra black sub pixel in the image causes the image to become distorted.

D - Secret information. K - Number of shares generated from D. share - piece of information.

Divide data D into n pieces in such a way that D is easily reconstruct able from any k pieces, but even complete knowledge of any k-1 pieces reveals no information about D. Stacking two pixels (each consists of four sub-pixels) can occur for example the following two cases: Secret sharing scheme is a method of sharing secret information among a group of participants. In a secret sharing scheme, each participant gets a piece of secret information, called a share. When the allowed coalitions of the participants pool their shares, they can recover the shared secret; on the other hand, any other subsets, namely non-allowed coalitions, cannot recover the secret image by pooling their shares. In the last decade, various secret sharing schemes were proposed, but most of them need a lot of computations to decode the shared secret information. The basic 2 out of 2 visual cryptography model consist of secret message encoded into two transparencies, one transparency representing the cipher text and the other acting as a secret key. Both transparencies appear to be random dot when inspected individually and provide no information about the original clear text. However, by carefully aligning the transparencies, the original secret message is reproduced. The actual decoding is accomplished by the human visual system. The original is encrypted into 2 transparencies you need both transparencies to decode the message.

f. Un-hiding Image from Transparency:

The simplest form of visual cryptography separates an image into two layers so that either layer by itself conveys no information, but when the layers are combined the image is revealed. One layer can be printed on a transparency, and the other layer displayed on a monitor. When the transparency is placed on top of the monitor and aligned correctly, the image is revealed. For each image pixel, one of the two encoding options is randomly selected with equal probability. Then, the appropriate colorings of the transparency and screen squares are determined based on the color of the pixel in the image.

g. Testing and integration:

This is the final module, which consists of integration of Visual cryptography implementation module into interface design using applet viewer. Then we need to test with various images and formation of transparencies. The transparencies should be able to save and load into the user interface.

IV. SYSTEM DESIGN

The system design is shown in fig 1

Process specification:

a. Input Design:

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy.

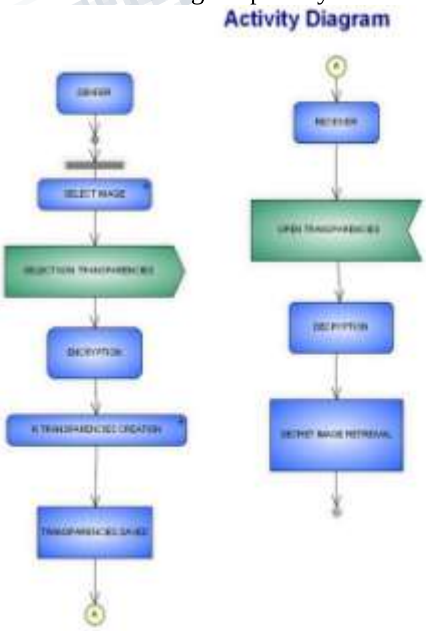


Fig 1: Activity Diagram

Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?

- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

b. Objectives:

Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

c. Output Design:

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

- Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.
- Select methods for presenting information.
- Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
    - Trigger an action.
    - Confirm an action.

d. Techniques and Algorithm Used:

In this technology, the end user identifies an image, which is going to act as the carrier of data. The data file is also selected and then to achieve greater speed of transmission the data file and image file are compressed and sent. Prior to this the data is embedded into the image and then sent. The image if hacked or interpreted by a third party user will open up in any image previewed but not displaying the data. This protects the data from being invisible and hence is secure during transmission. The user in the receiving end uses another piece of code to retrieve the data from the image.

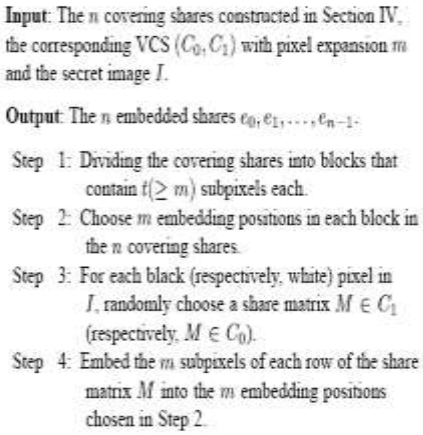
Algorithm:

Input: The c x d dithering matrix D and a pixel with gray-level g in input image I.

Output: The halftoned pattern at the position of the pixel

```
For i=0 to c-1 do
For j=0 to d-1 to do
If g<=Dij  then print a black pixel at position (i,j);
Else print a white pixel at position (i,j);
```

For embedding



Input: The n covering shares constructed in Section IV, the corresponding VCS (C_0, C_1) with pixel expansion m and the secret image I.

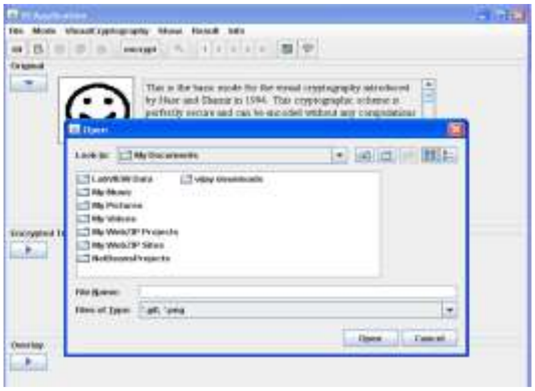
Output: The n embedded shares $e_0, e_1, \dots, e_{n-1}$.

```
Step 1: Dividing the covering shares into blocks that
        contain $t(\geq m)$ subpixels each.
Step 2: Choose $m$ embedding positions in each block in
        the $n$ covering shares.
Step 3: For each black (respectively, white) pixel in
        $I$, randomly choose a share matrix $M \in C_1$
        (respectively, $M \in C_0$).
Step 4: Embed the $m$ subpixels of each row of the share
        matrix $M$ into the $m$ embedding positions
        chosen in Step 2.
```

V. SCREEN HOTS

Fig 2: User interface - Steganography tool

User interface which allows the users to work with Steganography tool is shown in fig 2.



This is the basic mode for the visual cryptography introduced by Naor and Shamir in 1994. This cryptographic scheme is perfectly secure and can be decoded without any computations

Fig 3 : Load Image

To encrypt a image is shown in fig 3 proceed with the following procedure: Select file menu, Select load file sub menu and Load .gif or .png formatted images

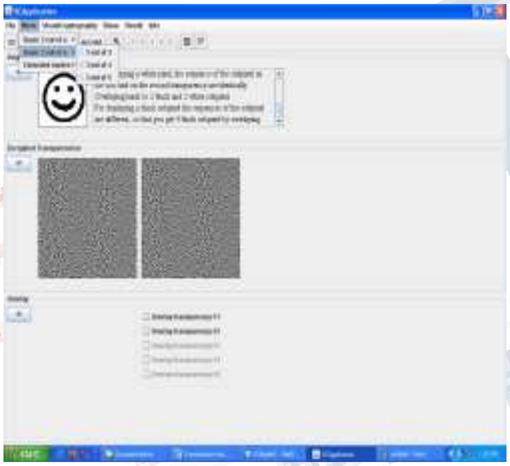


Fig 4: Mode of Encryption

We can select mode of encryption by selecting Mode menu as shown in fig 4.

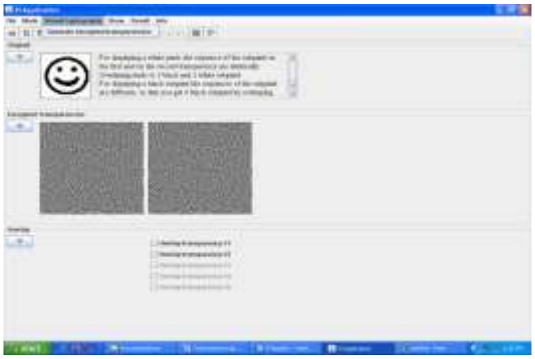


Fig 5: Generate encrypted transparencies submenu generates transparencies

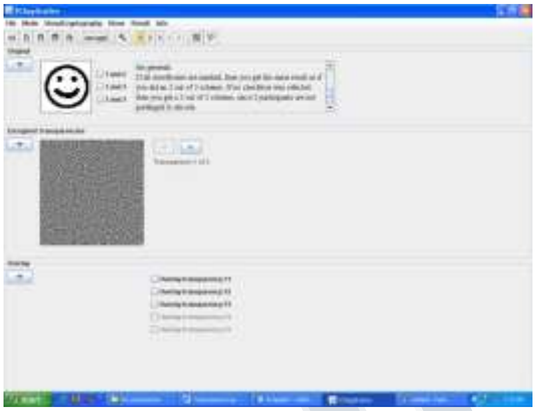


Fig 6: Generate encrypted transparencies submenu generates transparencies more.

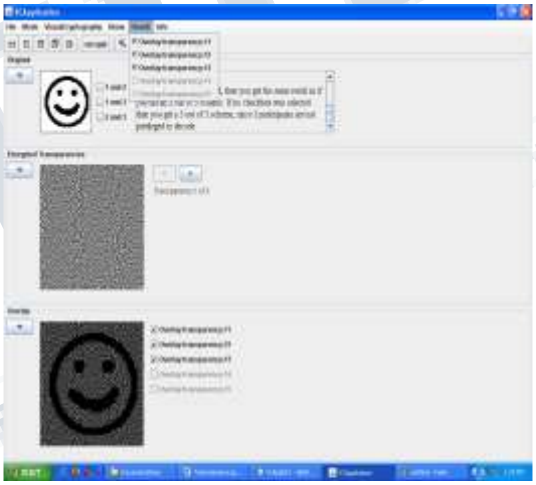


Fig 7: Decrypted Image

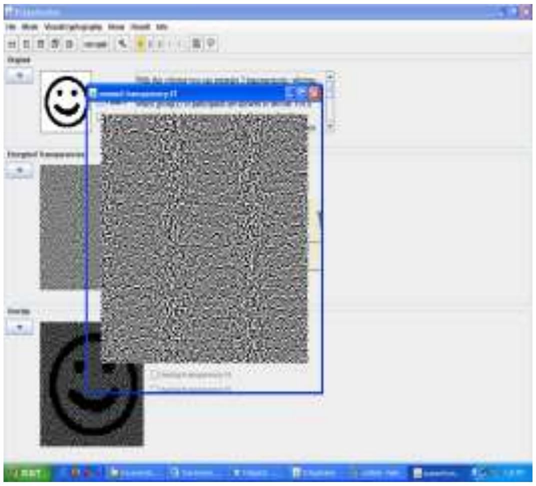


Fig 8: Zooming option supports zooming of transparencies

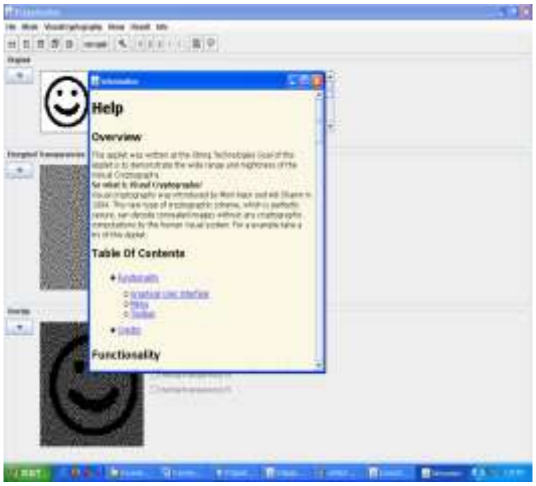


Fig 9 : Help dialogue provides information about usage of this application

VI. CONCLUSION

In this paper, we proposed a construction of EVCS which was realized by embedding the random shares into the meaningful covering shares. The shares of the proposed scheme are meaningful images, and the stacking of a qualified subset of shares will recover the secret image visually. We show two methods to generate the covering shares, and proved the optimality on the black ratio of the threshold covering subsets. We also proposed a method to improve the visual quality of the share images. According to comparisons with many of the well-known EVCS in the literature the proposed embedded EVCS has many specific advantages against different well-known schemes, such as the fact that it can deal with gray-scale input images, has smaller pixel expansion, is always unconditionally secure, does not require complementary share images, one participant only needs to carry one share, and can be applied for general access structure. Furthermore, our construction is flexible in the sense that there exist two trade-offs between the share pixel expansion and the visual quality of the shares and between the secret image pixel expansion and the visual quality of the shares.

Limitations & Future Enchantements:

In this paper, we propose a construction of EVCS which is realized by embedding random shares into meaningful covering shares, and we call it the embedded EVCS. Experimental results compare some of the well-known EVCSs proposed in recent years systematically, and show that the proposed embedded EVCS has competitive visual quality compared with many of the well-known EVCSs in the literature. In addition, it has many specific advantages against these well-known EVCSs, respectively.

REFERENCE

[1] A. Shamir, “How to share a secret,” Commun. ACM, vol. 22, no. 11, pp. 612–613, 1979.

[2] G. R. Blakley, “Safeguarding cryptographic keys,” in Proc. National Computer Conf., 1979, vol. 48, pp. 313–317.

[3] M. Naor and A. Shamir, “Visual cryptography,” in Proc. EUROCRYPT’ 94, Berlin, Germany, 1995, vol. 950, pp. 1–12, Springer-Verlag, LNCS.

[4] M. Naor and B. Pinkas, “Visual authentication and identification,” in Proc. CRYPTO’97, 1997, vol. 1294, pp. 322–336, Springer-Verlag LNCS.

[5] T. H. Chen and D. S. Tsai, “Owner-customer right protection mechanism using a watermarking scheme and a watermarking protocol,” Pattern Recognit., vol. 39, pp. 1530–1541, 2006.

[6] P. Tuyls, T. Kevenaar, G. J. Schrijen, T. Staring, and M. Van Dijk, “Security displays enabling secure communications,” in Proc. First Int. Conf. Pervasive Computing, Boppard Germany, Springer-Verlag Berlin LNCS, 2004, vol. 2802, pp. 271–284.

[7] C. Blundo, A. De Bonis, and A. De Santis, “Improved schemes for visual cryptography,” Designs, Codes and Cryptography, vol. 24, pp. 255–278, 2001.

[8] G. Ateniese, C. Blundo, A. De Santis, and D. R. Stinson, “Visual cryptography for general access structures,” Inf. Computat., vol. 129, pp. 86–106, 1996.

[9] N. K. Prakash and S. Govindaraju, “Visual secret sharing schemes for color images using halftoning,” in Proc. Int. Conf. Computational Intelligence and Multimedia Applications (ICCIMA 2007), 2007, vol. 3, pp. 174–178.

[10] H. Luo, F.X.Yu, J. S. Pan, and Z. M. Lu, “Robust and progressive color image visual secret sharing cooperated with data hiding,” in Proc. 2008 Eighth Int. Conf. Intelligent Systems Design and Applications, 2008, vol. 3, pp. 431–436.