Visual Cryptography: Image Sharing and Hiding

A new Visual Cryptographic Scheme for Grey-Scale and Color Images
Introduction

- Earlier, Cryptography was concerned solely with message confidentiality - conversion of messages from a comprehensible form into an incomprehensible one, and back again at the other end, rendering it unreadable by interceptors without secret knowledge.
- Hence most of them require intensive mathematical and algebraic calculations in their encryption and decryption.
- In Visual Cryptography, the secret information (an image) is split into $n$ shares (images), such that the decryption can be performed by the human visual system by simply superimposing the shares.
- Since the reconstruction is done by the human visual system, no computations are involved. But there is certain amount of contrast loss in the recovered image.
Visual Cryptography Scheme for Binary (black & white) Images

- Suppose we have an image made up of only black and white pixels. Then, a pixel P is split into two pixels in each of the two shares.

  If P is white, then we randomly choose one of the first two rows.

  If P is black, then we randomly choose one of the last two rows.
Each of the two possibilities "black-white" or "white-black" is equally likely to occur, independent of whether the corresponding pixel in the secret image is white or black.

Thus just by looking at one of the shares, it is not possible to predict the corresponding pixel in secret image.

Since white pixel is now a combination of white-black pixel, there is contrast loss of 50%.
Grey- Scale Images

- In the current method for grey images, we decompose every pixel into $n \times n$ sub-pixels. The original image is reduced to $n^2$ levels.
- We construct basis matrices for each $n^2$ sublevels in the image, and randomly permute it to get the two parts corresponding to each share.
- For $n=2$, the matrices are as follows:

\[
\begin{align*}
S_{00} &= \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} \quad \text{and} \quad S_{01} = \begin{bmatrix} 3 & 2 \\ 1 & 0 \end{bmatrix} \quad \text{hence the overlap gives} \quad \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad \text{(black),} \\
S_{10} &= \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} \quad \text{and} \quad S_{11} = \begin{bmatrix} 0 & 3 \\ 2 & 1 \end{bmatrix} \quad \text{hence the overlap gives} \quad \begin{bmatrix} 0 & 1 \\ 2 & 1 \end{bmatrix}, \\
S_{20} &= \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} \quad \text{and} \quad S_{21} = \begin{bmatrix} 0 & 1 \\ 3 & 2 \end{bmatrix} \quad \text{hence the overlap gives} \quad \begin{bmatrix} 0 & 1 \\ 2 & 2 \end{bmatrix}, \\
S_{30} &= \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} \quad \text{and} \quad S_{31} = \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} \quad \text{hence the overlap gives} \quad \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} \quad \text{(white)}.
\end{align*}
\]
Results for $n = 3$

Original Image

Reduced Grey Scale Image to 9 Levels

A Closer Look

Share 1

Share 2

Superimposed Image
The contrast levels are reduced to 33%.

The pixel expansion is $n \times n$, so the share size is $n$ times the original image.

When applied using higher values of $n$, finer images are generated since more grey-levels are covered.

Algorithm for generating basis matrix.

1. for $i = 0$ to $i = n^2 - 1$
2.    for $j = 0$ to $j = n^2 - 1$
3.       $s_{0}[j/n][j\%n] = j$
4.       if( $j < i$ )
5.          $s_{1}[j/n][j\%n] = j$
6.       else
7.          $s_{1}[j/n][j\%n] = (n^2) - 1 - (j - i)$
Extended Scheme for Color Images

- All color images can be split into three components Red, Green and Blue. (0-255)
- Each of these components is treated as a separate “grey-scale” image, and two shares for each color are generated.
- R1 & R2 for Red, G1 & G2 for Green, and B1 & B2 for Blue.
- The first shares of R,G,B are merged to get final Share1 and other shares are merged to get Share2.
- Share1 = (R1 + G1 + B1)
- Share2 = (R2 + G2 + B2)
- The contrast levels vary from 25% - 33% of the original image.
Results for a Color Image with $n = 2$

Original Image

Reduced Red Scale Image With 4 levels.

Green Scale

Blue Scale
Another Example

Original

Green Scale

Red Scale

Blue Scale

Share 1

Share 2
General Access Structure

**Visual cryptographic scheme** for a set of $n$ participants is a cryptographic paradigm that enables a secret image to be split into $n$ shares, where each participant in $P$ receives one share.

(k,n) Visual Threshold schemes (VTS)

1. The Secret Image is visible if and only if any ‘$k$’ transparencies are stacked together, but if fewer than ‘$k$’ transparencies are super-imposed, it is impossible to decode the original image.

2. Presently, not many schemes are known about the general access structure for VCS.

3. We will give an example depicting the use of General Access Structure.
Examples of General Access Structure

Suppose that there is a locker in a Bank and the management of the Bank decides that the key of the locker must be distributed among the customers and certain employees of the Bank in such a way that the locker of some customer will be opened only when that customer and designated employees will contribute their keys.

As an example, let B be the Boss and M1, M2, U1 & U2 be 2 managers and 2 users.
1. Suppose that the management of the Bank decides that any user with Boss can open the locker.
2. But due to some reason, if the Boss is absent, then an user can open the locker together with both the managers.
3. To tackle this type of situations, we need General Access Structure.

General Access Structure thus defines the Minimal Qualified Set of the participants depending upon various constraints specified by the Management of the Organization.
Our Software

We have built a Software which implements this General Access Structure in Visual Cryptography.

Software Specifications:-

- **Input-**
  - a. Secret Image
  - b. Text file specifying number of participants and various constraints on the qualifying and forbidden sets.

- **Output-**
  - a. ‘n’ shares (images).

- **Implementation-**
  Implemented in C++ language using Intel’s openCV Libraries in Microsoft Visual Studio.
New Constructions

- We worked on a new Data Hiding Scheme which is not a Visual Cryptographic scheme, but it produced appreciable results.

- This scheme has 100% contrast and 0 pixel-expansion for Grey Images and can be further extended up to Colour-images.

- The following scheme cannot be implemented visually, rather it has to be implemented digitally using computers.

- **Concept Used:**
  - Negative pixel weights have been used for Data-Hiding.

  1. A pixel value of (-2) is equivalent to 253, this means that all pixel values are obtained as (value%255).

  2. Thus to obtain pixel value 10, we use 200, -190 and adding them leads to pixel value 10.

  3. This means that a very bright white pixel(200) added with Gray colour pixel (-190 or 65) superposes to almost black pixel(10). This ensures the secret encryption for the pixel values.
Advantages

The Development of this Data-Hiding technique has many advantages and can be extended up to Color-images as well.

- **100% Color-contrast ratio.**
- **Pixel Expansion: Nil**
- **Construction of Constraint Access Structure from this technique**
  - We will define what is meant by **Constraint Access Structure** with an example which shows its difference from that of General Access Structure.
  - We have created what is called Constraint Access Structure. Here, the constraints imposed by administrator does not specify the Minimal Qualified set like General Access Structure.
  - Instead, it defines the exact set of participants who has the authority to recover the Secret Image.
  - In General Access Structure, any superset of the qualified participants can recover the Image, which does not happen in our structure.
Example depicting use of our Structure

- Suppose that there is an Intelligence agency with 2 Administrators who possess the keys/image shares to unlock a particular system. The unlocking can be done from distant locations through Internet and in that case, they would be required to upload their individual shares.

- Suppose that there is an adversary in the organization who knows the time when the 2 administrators upload their share images to unlock and view some confidential data.

- So, if he/she also uploads his/her image, then by General Access Structure, any superset of the qualified participants would be able to recover the secret image and hence view the confidential data.

- But, in Constraint Access Structure, the unlocking wouldn’t be done because it recovers the secret image only when the exact participants as specified by the administrator superposes their shares/images.
Thank You...