Visual Cryptography

Shahzan Ahmad - 2020117

Motive:

- To introduce a new cryptographic scheme which relies on the human visual system instead of cryptographic computations to decode secret images.
- We'll do so by looking at the problem as a k out of k secret sharing problem.
- Given an image and an integer k we create k shares (k new images) and each one of them would seem like a collection of "random" black and white pixels.
- Overlap any r < k images and the result should again appear as a random bunch of black and white pixels.
- But overlapping exactly k shares will give you the secret image

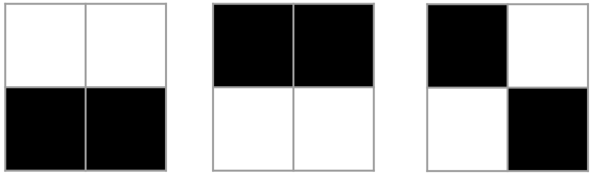
Basic Model:

- For each pixel in the secret image, we replace it with a rectangular block of m pixels in each of the k shares.
- The information regarding this replacement will be stored in k x m matrices where k is the number of shares.
- For each pixel in the secret image we choose a random matrix from a set of such matrices and substitute the corresponding location in the ith share with ith row of the chosen matrix for all 1 <= i <= k

0	0	1	1	Share1
1	1	0	0	Share2
1	0	0	1	Share3

Matrix chosen for any pixel

Replacement blocks:



share1 share2 share2

Correctness and Security:

1. Correctness:
   - Overlaying the k shares we should get the secret image

2. Security:
   - Overlaying any r < k of the shares the probability of getting the secret image remains the same as if one had no shares(images) at all

Correctness(Contrast):

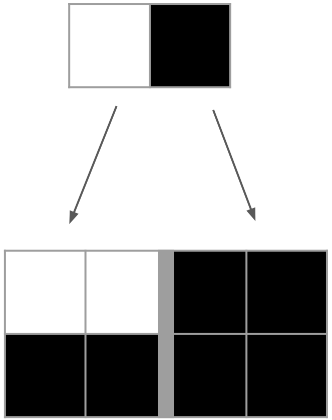
The logical ‘or’ of the k pixel-blocks in the k shares corresponding to any pixel in the secret image should represent the pixel in the original image.

i.e. if the original pixel is black then on overlapping k pixel-blocks it should look black to human eyes and the same for white.

NOTE: the replacement blocks will have black and white pixels and hence it’s impossible to reproduce the color white by overlapping the blocks so the white pixels will appear as gray.

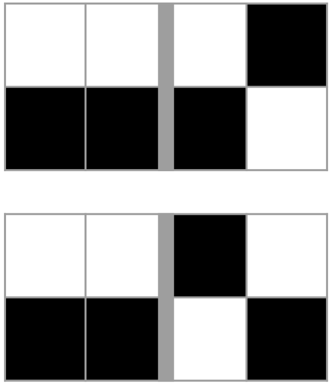
Correctness(Contrast):

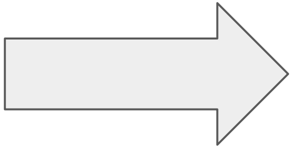
Secret image:



Share1 :

Share2:





Matrix chosen for left pixel:

0	0	1	1
0	0	1	1

Matrix chosen for right pixel:

0	1	1	0
1	0	0	1

Notice that in the matrix used for replacing white pixel the hamming weight of the ‘or’ of all the rows is closer to 0 than the matrix used to replace black pixels

Correctness condition:

Matrix for a white pixel: a random matrix from C0 (a set of m x k matrices)

Matrix for a black pixel: a random matrix from C1 (a set of m x k matrices)

Then for any S1 in C1 the ‘or’ V of all the rows in S1: $H(V) >= d$

and for any S0 in C0 the ‘or’ V of all the rows in S0: $H(V) <= d - ma$

(*Visual Cryptography, Moni Naor and Adi Shamir*)

a is the relative difference between H(V) for any S1 and S0
Assuming that $H(V) = f(r)$ where $r<k$ is the number of rows

Security:

- **Idea:** If one sees any r < k shares then it should be equally likely for any block to represent black or white pixel of the original image.
- Hence for any {i1, i2 . . . iq} ⊂ {1, 2, . . . k}: q < k the two collections of matrices D0 and D1 obtained by restricting matrices of C0 and C1 to rows i1, i2, . . . iq should have the same matrices with same frequencies(*Visual Cryptography, Moni Naor and Adi Shamir*)
- Which will model the real life scenario where someone has q < k shares and upon seeing any particular block one’s confused about whether the block is used to represent a white or a black pixel.

2 out of 2 case

C0 = set of all the matrices we get by permuting the columns of:

0	0	1	1
0	0	1	1

C1 = set of all the matrices we get by permuting the columns of:

0	0	1	1
1	1	0	0

K out of K case:

We build a scheme with C0 and C1 containing matrices of size: (k x 2^(k-1)) and hence each block will contain 2^(k-1) pixels

- Define E as the set containing all the even cardinality subsets of {0, 1. . . k-1 }
- And O as the set containing all the odd cardinality subsets of {0, 1 . . . k-1}
- Define S0 as: (S0[i, j] = 1) iff (i ϵ E(j))
- Define S1 as: (S1[i, j] = 1) iff (i ϵ O(j))

S0 for any k will have only zeroes in its first column

S1 for any k will have no column with all zeroes

Hence d = 2^(k-1) for correctness criteria

Proof of correctness (contrast):

S0 for any k will have only zeroes in its first column

S1 for any k will have no column with all zeroes

Hence d = 2^(k-1) for correctness criteria