VISUAL CRYPTOGRAPHY

MYRTO ELEFTHERIA GKOGKOU - 3914

What is secret sharing ?

Secret sharing refers to methods for distributing a secret among a group, in such a way that no individual holds any intelligible information about the secret, but when a sufficient number of individuals combine their 'shares', the secret may be reconstructed.

Insecure secret sharing allows an attacker to gain more information with each share.

What is visual cryptography?

It was introduced by Naor and Shamir at the EUROCRYPT conference in 1994.

It’s a cryptographic technique that focuses on solving secret sharing problems where the secrets are hidden in images (image sharing).

In a (k,n) image sharing problem, the image that carries the secret is split up into n shares and the decryption is successful only when at least k shares are stacked together. If fewer than k shares are stacked, then the image remains hidden.

Something interesting about it is that the human eyes do the decryption. This means that it can be implemented without any knowledge of cryptography and also without performing any computations. Plus, it is considered a very secure mechanism.

How it works?

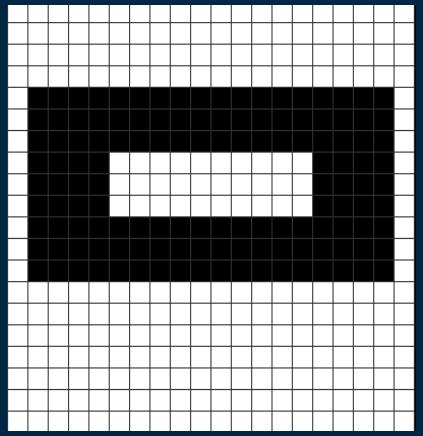
To understand how visual cryptography works, the best way is to do an example. A few things to keep in mind:

- We talked about (k,n) image sharing. The easiest way is when k = n. This means that all n shares are needed to gain the original image. In the example we are going to work with k = n = 2.
- The original image and the shares are a collection of black and white pixels and each pixel is handled individually.
- White pixels in the shares represent the transparent colour.

Example

- Step 1: Choose a monochrome image.

  As a secret image we are going to use:



This is a 20 x 20 image that contains a rectangle.
Every little square is a pixel.
Notice that the pixels that contain information are black.

Example

Before we move on to the next step, we need to know that in the shares we are going to work with sub-pixels. This mean that for every pixel in the original image there is a group of sub-pixels in each share that refers to this pixel. We will use 4 sub-pixels (pixel expansion) which is the most known way. In this case the shares will have a width twice as the width of the secret image.

Example

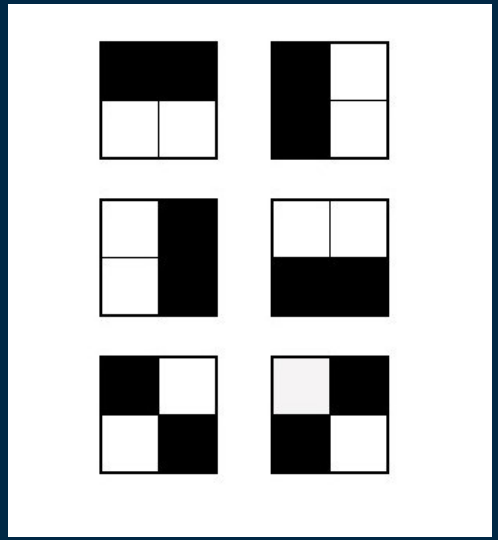
- Step 2: Create the two shares.

  We need to determine the colour of each sub-pixel in both shares such that, if someone has only one share there is no way of decrypting the message, but when we stack together the two shares the original message is revealed.

  How to do that ?

Example

Well, first we need to determine our states. A state is a combination of black and white sub-pixels that could possibly appear in any of the two shares. All states contain two white and two black sub-pixels.



Example

Now that we have our states we are ready to create our two shares.

We look at the colour of each pixel in the original image.

- If the pixel is black, we randomly choose one of the previous states and fill the 4 sub-pixels for this specific pixel in one of the two shares – doesn't really matter which one. For the other share we choose the complementary state.

Let's assume that we randomly chose this state for the first share:

The state in the second share is going to be the complementary:

Example

- If the pixel is white, again we choose a random state for one of the two shares. The difference is that in the other share instead of the complementary state we use the same state.

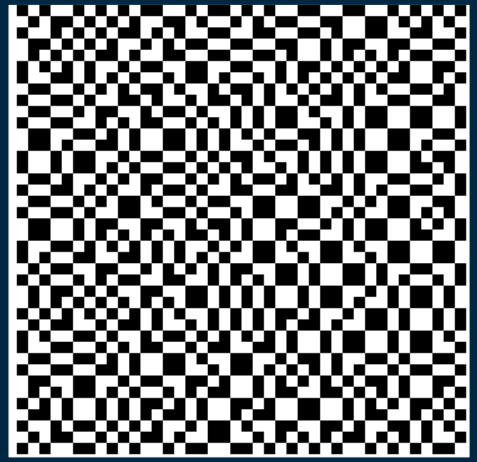
Let’s assume that we randomly chose this state for the first share:

The state in the second share is going to be exactly the same :

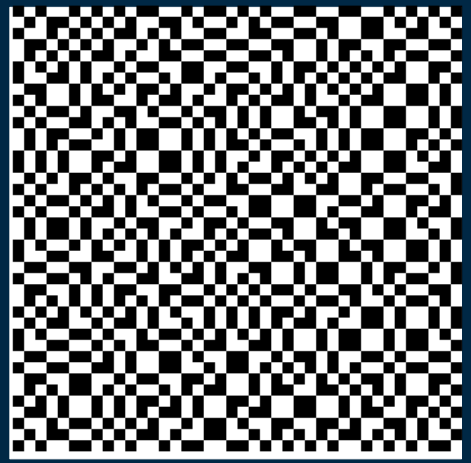
Example

If we fill every sub-pixel of the two shares for each pixel in the original image, our two shares will probably look something like this:

Share 1

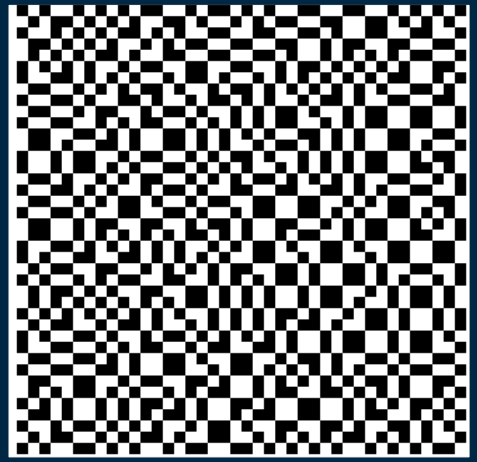


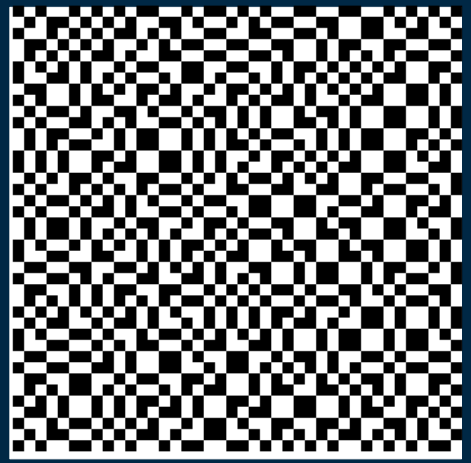
Share 2



Example

What we achieved is that by looking any of these two shares it’s impossible to guess the original image. This system is perfectly secure like OTP because the pixels in the shares are truly random – every state has a probability of 1/6.





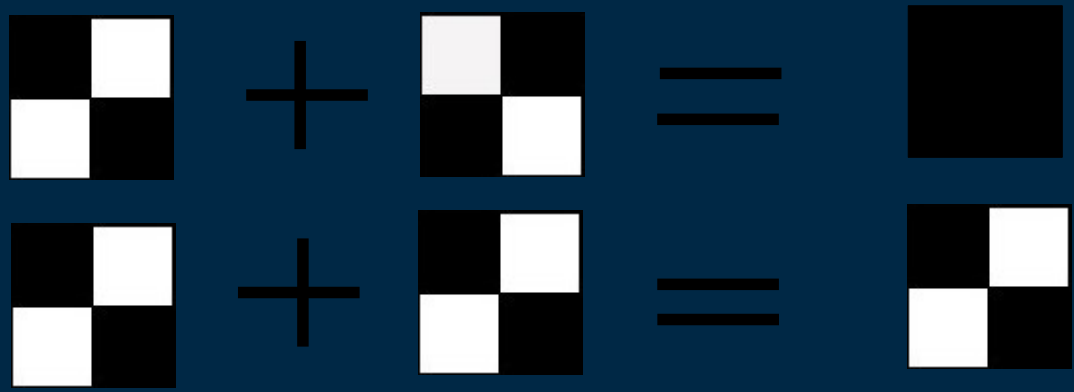
Example

How do we decrypt the original message ?

We carefully align the two shares and the secret is visible by the human eye!

Why is that ?

Because by aligning the two shares it’s like performing the OR operation.

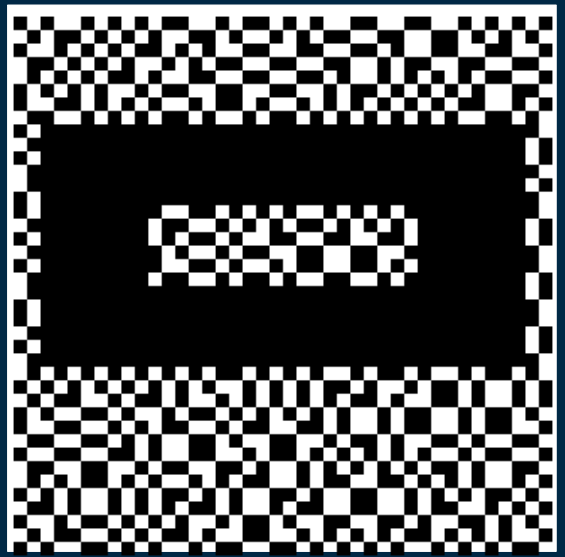


Information Pixel

Empty Pixel

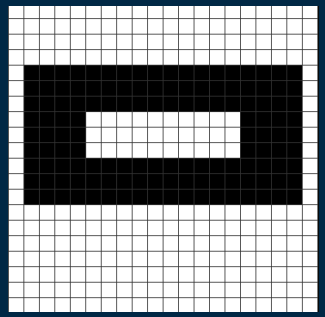
Example

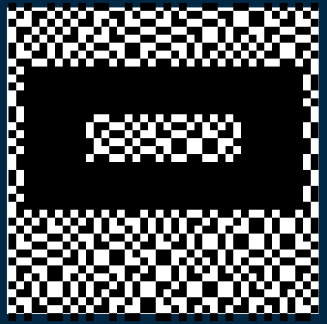
If we align the two previous shares we get the secret image, or at least kind of...



Example

Okay, the black of the secret image remains black but what happened to the white? As we saw in a previous slide, an empty pixel is not actually empty, but it contains two black and two white (or transparent) pixels. That's why we end up having random black pixels or sometimes medium-grey pixels instead of white pixels. Luckily, this is still sufficiently high enough contrast.





Additional

We can implement this system in a more generic way. For instance:

- We can create more than two shares.
- We can modify the states and the shares such that only a subset of shares are needed to be combined to reveal the image.
- We can split a pixel into more sub-pixels.
- We can even encrypt colorful images.

Advantages

- We don’t need a decryption algorithm. We only need an algorithm that gives as an output the shares.
- If it’s correctly implemented it can be considered as perfectly secure.
- It’s easy to implement.

Disadvantages

- The result is not the exact same with the original image but it has noise
- Due to the pixel expansion we have a loss of information – that’s why the best solution is to choose the smallest number of sub-pixels based on the problem we have to solve.
- Perfect alignment of the shares is a little bit of trouble.

Usage

- Steganography
- Watermarking
- Anti-phising systems
- Authentication methods

Do you have any questions?

THANKS

REFERENCES

- [1] M. Naor and A. Shamir, Visual Cryptography
- [2] J. Weir and W. Yan, A Comprehensive Study of Visual Cryptography
- [3] Wikipedia, Visual Cryptography

Links:

- https://www.ciphermachinesandcryptology.com/en/visualcrypto.htm
- https://www.101computing.net/visual-cryptography/