Digital Image Processing Final Project

Lu, Zhi-Bao	Pao, Yu-Wen	Yun-Ye, Cai
R12922196	B09902016	R12922104
r12922196@ntu.edu.tw	b09902016@csie.ntu.edu.tw	r12922104@csie.ntu.edu.tw

Abstract

Visual cryptography (VC) is an encryption technique that divides an image into multiple shares, such that the original image can only be revealed when a sufficient number of shares are combined. This proposal outlines an experiment aimed at comparing various visual cryptography methods specifically designed for gray-scale images. The study will delve into the visual quality and the efficiency of these methods, providing valuable insights into their real-world applications.

1 Motivation

In Lecture 6 on halftoning, we acquired skills related to the technique. The professor briefly linked visual cryptography skills to halftoning, highlighting its relevance. Subsequently, we uncovered numerous security applications for these skills. The more we delved into visual cryptography's capabilities, the more curious we became about its actual implementation. Our final project focuses on studying visual cryptography in the detail.

2 Introduction

Visual cryptography, first proposed in Naor and Shamir[18], is an encryption technique to encrypt the secret images into multiple share images. The VC are composed of two parts, encryption and decryption. The encryption process is to generated the unrelated share file. The share file can be random dots or natural images based on different algorithms. The decryption process is stacking two share image or sufficient amount of share images. The human can extract the hidden information from the decrypted images. Compared with the conventional cryptographic methods[23, 13, 6], it puts the human perception into the decryption process to acquire the information in the original secret images, instead of relying on the complex mathematical theories and computation. VC have already show some potentials in the real-world applications[21], like watermarking[24, 9, 22], digital signature[4], secret communication[25], authentication[10].

3 Problem description

3.1 Definition

Fig. 1 shows the default pipeline how a visual cryptography works. The VC scheme can be viewed as a k out of n secret sharing problem, where n represents the number of shares generated during encryption. The secret image can be recovered only if at least k shares are combined. During the encryption phase, we can categorize it into pixel expansion, type of shares , and quality, among other factors. We will discuss these aspects further in the following section 3.2.

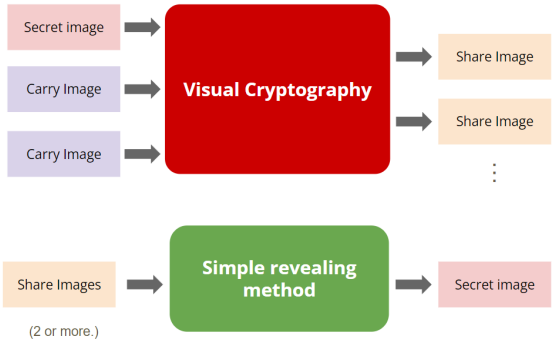


Figure 1: The overflow chart of visual cryptography scheme.

In the aforementioned framework, the secret image can be categorized as binary, grayscale, or color image. However, due to time constraints, we want to simplify the secret image range without making it overly simplistic. Therefore, we will focus on discussing **binary** and **grayscale** images.

3.2 Visual cryptography metrics

In this section, we will discuss the metrics we use for this project. We will present a detailed analysis of how each metric contributes to evaluating the performance and effectiveness of our VC approach.

Pixel expansion refers to number of sub-pixels m in generated shares that represent a single pixel in an original image. This parameter presents in loss of resolution from an original image to share image in VC procedure.

The **type of shares** of a VC scheme can either be meaningless or meaningful.A meaningless share resembles noise whereas a meaningful share is a discernible image used to embed information from the secret image. Depending on the application, the use of one or the other may be preferred. For example, a meaningless share may rouse the suspicion of an adversary whereas a meaningful share may not.

Contrast refers to the distinction between the black and white pixels in binary images or the variation in color tones in colored images. It is an indicator of an image's clarity or sharpness. In visual cryptography (VC) schemes, the contrast of images that have been decrypted is assessed as a quality metric. Often, the process of encrypting and decrypting images in VC can result in diminished contrast.

The term **number of secrets** refers to how many secret images are encrypted using a visual cryptography (VC) technique. In cases where multiple secrets exist, these images are encrypted together into a single collection of shares. The decryption is achieved by superimposing these shares, which may be rotated or flipped, to reveal different secret images.

The **accuracy** of a visual cryptography (VC) scheme is determined by the fidelity of the recovered image to the original secret image. The aim of a VC scheme is to achieve the highest possible precision in reproducing the original image. This accuracy is typically assessed using metrics such as Peak Signal-to-Noise Ratio (PSNR), Mean Square Error (MSE), and Correlation Coefficient (CC).

4 Methodology

In this project, we will implement 4 different methods: **Pattern-based**, **Error Diffusion**, **Hierarchical Visual Cryptography**, and **Absolute Moment Block Truncation Coding Compression**. We will thoroughly discuss each method using the aforementioned metrics, evaluating their pros and cons in detail.

4.1 Pattern Based

Pattern based visual cartography is one of the most simplest method for encrypting pictures. The concept involves replacing each pixel with a predefined specific pattern. Based on the type of shares, pattern-based cryptography can be categorized into two types: meaningless shares and meaningful shares.

4.1.1 Meaningless share

A pioneer work is mentioned in [18], whose main idea is the earliest is regarded as earliest visual cryptography method. Consider the basic example for meaningless share shown in Fig. 2. If the origin pixel is black, our go is to fill the entire block with black after stacking the shares. Conversely, if the origin pixel is white, the share pattern will be identic for each share. This approach ensures that some white pixels appear in the resulting shares. This design lows us to control the placement of black and white pixels in the resulting image as desired.

4.1.2 Meaningful shares

Meaningful shares were developed to enhance the undetectability during the transfer of shared images. Transferring a noisy image could raise suspicions of conceing secret information. Generally speaking, our shares should be like ordinary images before hiding the secret information. Chen et al.[15] use 2 input share meaningful images to cover 1 secret images. The core idea behind their design involves using four specific parameters to control the contrast of each block in the output image, ensuring they conform to the specified pattern simultaneously. Benefiting from the dynamic hyperparameter threshold. Here we summarize their implementation method

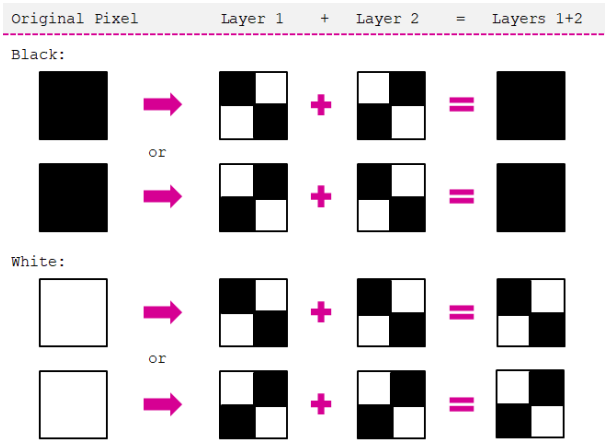


Figure 2: A simple demo for (2, 2) pattern based visual cryptography [5].

and simplified into the pseudo code showed in Alg. 1. By doing so, we can embed secret image information into visually meaningful pictures easily.

Algorithm 1 Chen's pattern based method

Ensure: The black pixel parameter for black and white block in share and secret images.
for `block` **in** `image` **do**
0. Do the preprocessing for both secret and share image block.
1. Calculate the number of black pixels in the block(secret and share).
2. Record the position for all black pixel position for each block.
3. Make sure the number of black pixels in stacked share image are same for secret image block.

end for

For grayscale image, Blesswin et al.[2] presents an Improved Grayscale Visual Secret Sharing (IGVSS) method for securely sharing grayscale images. The overall scheme flow is shown in Fig. 3. It uses error diffusion to convert a secret image into a halftone image and then generates two share images using one or two meaningful visual images (MVIs). These shares appear as random noise individually but can reconstruct the secret image when combined. This method ensures high-quality image reconstruction with minimal pixel expansion, making it ideal for secure applications like medical imagery and confidential data transmission. The IGVSS scheme enhances both security and visual quality compared to previous methods.

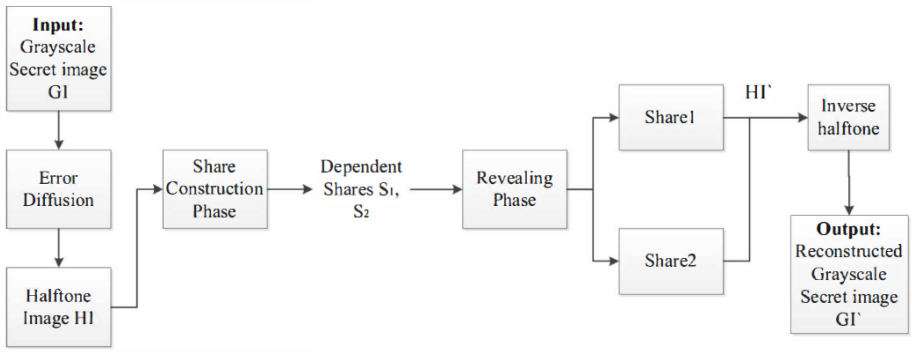


Figure 3: General flowchart of IGVSS scheme

4.2 Error diffusion

Since we focus on the visual cryptography scheme for grayscale images and the visual cryptography is based on the boolean operation, halftoning techniques come in handy as a preprocess to convert the grayscale images into binary images for the ongoing encryption. Error diffusion[14] is one of the most common halftoning techniques where quantization errors from pixel values are dispersed to neighboring pixels based on the designed kernel[8, 12]. The error-diffusion is one of the main streams of the visual cryptography for grayscale image and they for focusing on embedding the image through the diffusion process, instead of using the halftoning as a pre-processing. Due to the nature of the error-diffusion, it also come with some benefit that the share image can be two natural images instead of random pattern in most of the other VC works.

M. Nakajima et al.[17] is one of the pioneers in the error-diffusion-based visual cryptography with high quality. The inputs of their algorithm are two carry images and one secret image, which is grayscale. The output are two carry images, which is similar to the input carry images respectively. First, they introduce the fundamentals of the visual cryptography, especially for extended visual cryptography, which means using stacking (AND operation) in the decryption process. It suggest the proper value range of the share images and the secret image that can be properly by stacking. The follow is an example. The pie chart of (a) and (b) means the pixel values of the same position in the two share images, and A_1 and A_2 means the white portion (transparent part). A_Ω means the maximum value of a pixel. After the stacking of the two share images, the value that they can decrypted is in the range of $[max(0, (A_1 + A_2) - A_\Omega), min(A_1, A_2)]$. The maximum value happens when two black parts from the share images overlap as much as possible. In the other hand, the minimum value happens when two black parts from the share images don't overlap with each other.

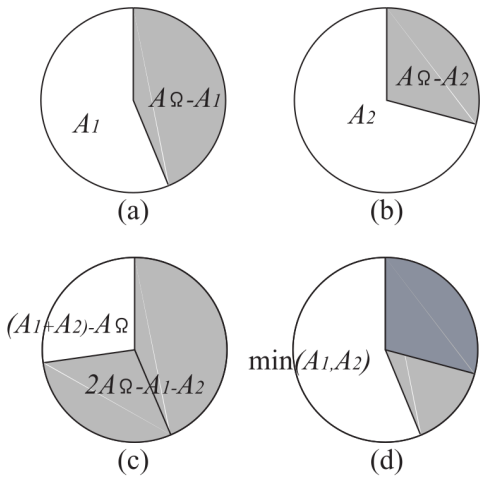


Figure 4: An example of the range of the share images and the secret images in the extended visual cryptography

To deal with this characteristic of the EVC, the authors increase the pixel value of the carry image and decrease the pixel value of the secret image to enhance the contrast. After modify the image to the proper value range, they use the error-diffusion for visual cryptography to

diffuse the error in three directions, the secret image and the two share images. The following image is a more detail of their error-diffusion algorithm.

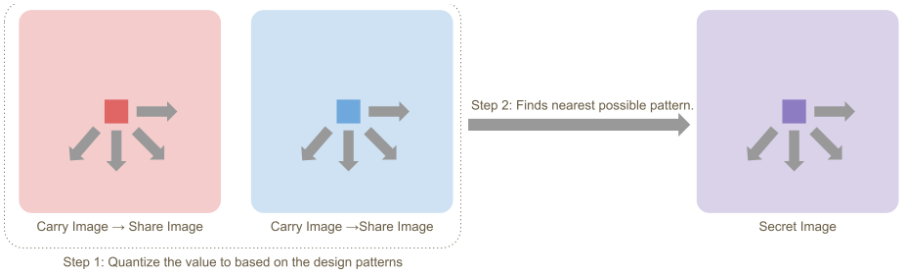


Figure 5: The workflow of the error diffusion for EVC.

The first step is to quantize the value of the two carry image and create the pattern based on the quantize value. In this work, they quantize the value into 9 bins. The second step is to find the find the pattern that can be generated by stacking the pattern from two share images and is also the nearest to the value of the secret image. Then, the next is the compute the error between the quantize values calculated from the assigned 3×3 patterns and the original value of the two carry images and the secret image. Finally, diffuse the error for each image based on the designed weight.

However, the pixel expansion problem still remain in the error-diffusion of the error-difussion-based methods. Following the previous work[16] dealing with pixel expansion on binary image VC, [1] use the error-diffusion to extend the VC scheme to grayscale image for the share images only. In more recent works, Qing et al.[11] also apply the algorithm powered by error diffusion on the gray-scale share images to generate the final share image. In this project, we implemented the algorithm from Qing et al.[11]. The detail of their algorithm is following.

Algorithm 1 Visual cryptography by error diffusion

Require: a grayscale image f, parameters α and γ
Ensure: share images $S_1 = [s_{1ij}]$ and $S_2 = [s_{2ij}]$

Compute the normalized image $\tilde{f} = [\tilde{f}_{ij}]$ by (1);
Initialize arrays as $e_{ij} = \varepsilon_{1ij} = \varepsilon_{2ij} := 0$;
Initialize arrays as $c_{1ij} = c_{2ij} := \gamma$;
Initialize vectors as $\boldsymbol{b}_0 := \begin{bmatrix}0\\0\end{bmatrix}$, $\boldsymbol{b}_1 := \begin{bmatrix}1\\0\end{bmatrix}$, $\boldsymbol{b}_2 := \begin{bmatrix}0\\1\end{bmatrix}$;
for $i = 0$ to m **do**
  for $j = 0$ to n **do**
    if $\tilde{f}_{ij} + e_{ij} \geq \theta$ **then**
      $b_{ij} := 1$;
      $d_{ij} := \tilde{f}_{ij} + e_{ij} - b_{ij}$;
      $s_{1ij} = s_{2ij} := 1$;
      $\boldsymbol{\delta}_{ij} = \begin{bmatrix}c_{1ij} + \varepsilon_{1ij}\\c_{2ij} + \varepsilon_{2ij}\end{bmatrix} - \begin{bmatrix}1\\1\end{bmatrix}$;
    else
      $b_{ij} := 0$;
      $d_{ij} := \tilde{f}_{ij} + e_{ij} - b_{ij}$;
      Compute

$$t^* = \arg\min_{t\in\{0,1,2\}} \left\| \begin{bmatrix}c_{1ij} + \varepsilon_{1ij}\\c_{2ij} + \varepsilon_{2ij}\end{bmatrix} - \boldsymbol{b}_t \right\|_1 ;$$

      $\begin{bmatrix}s_{1ij}\\s_{2ij}\end{bmatrix} := \boldsymbol{b}_{t^*}$;
      $\boldsymbol{\delta}_{ij} = \begin{bmatrix}c_{1ij} + \varepsilon_{1ij}\\c_{2ij} + \varepsilon_{2ij}\end{bmatrix} - \boldsymbol{b}_{t^*}$;
    end if
    for (k, l) in $\{(0,1),\ (1,-1),\ (1,0),\ (1,1)\}$ **do**
      if $(i + k, j + l) \in \Omega$ **then**
        $e_{i+k,j+l} := e_{i+k,j+l} + w_{kl} d_{ij}$;
        $\begin{bmatrix}\varepsilon_{1,i+k,j+l}\\\varepsilon_{2,i+k,j+l}\end{bmatrix} := \begin{bmatrix}\varepsilon_{1,i+k,j+l}\\\varepsilon_{2,i+k,j+l}\end{bmatrix} + w_{kl}\boldsymbol{\delta}_{ij}$;
      end if
    end for
  end for
end for

The algorithm can be described as the following steps. First, normalize the secret image based on the following formula, which will convert the secret image to the proper range of the pixel value.

$$\tilde{f} = \alpha \frac{f - min(f)}{max(f) - min(f)}$$

where f is the secret image, and α is a parameter, whose default value is 0.5. Secondly, initialize the share image as a grayscale image and the error for each image as zero. Then, the core step of this algorithm is assign the pixel value for two share images, which can be decomposed into two part. The first part is to determine the value of the decrypted secret image by the original secret image, error values, and the threshold. The second part is to find the binary values of the two share images that can generate the pixel value of the secret image and are also create the smallest error for both share images. Finally, compute the error for the secret image and the two share image, and diffuse the error based on the weights.

4.3 Hierarchical Visual Cryptography(HVC)

The code concept of Hierarchical Visual Cryptography involves encrypting a secret image using multiple levels, creating hierarchical structure [3, 28]. In this structure, the secret image

undergoes decomposition into several shares, and each of shares is further divided into additional shares, as illustrated in figure 6.

Additionally, in the hierarchical structure, specific shares can be overlapped to create a *key* share, which is then combined with final share to recovery secret image. This method allows us to provide the *key* administrators while giving the other share to users. Thereby, HVC is particularly suitable for authentication systems or cryptosystem [7] due to its complexity involved in obtaining the shares, making it challenging for hackers for accessing sensitive information.

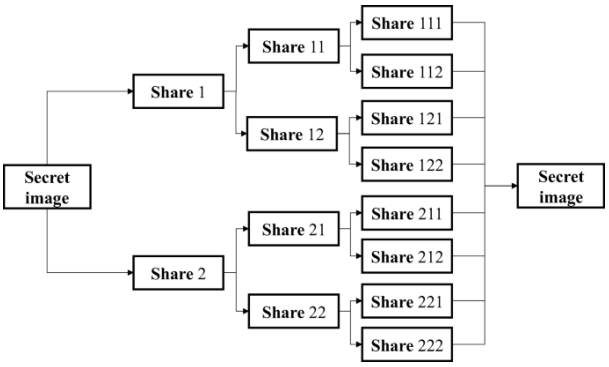


Figure 6: Hierarchical Visual Cryptography

4.4 Absolute moment block truncation coding(AMBTC) compression

Absolute Moment Block Truncation Coding (AMBTC) compression is another technique used to implement visual cryptography for grayscale images. It operates by dividing the image into non-overlapping blocks and encoding each block using a simple binary representation based on the comparison of pixel values with a threshold. However, unlike methods suck as HVC or error-diffusion, AMBTC doesn't support the recovery secret image with multiple shares.

It's worth noting that Ou and Sun [20] were the first to introduce a reversible AMBTC scheme. The secret image can be reconstructed by OR or XOR decryption, and the AMBTC shares can be reverted to their original image.Furthermore, C.N. Yang et al. [27] expanded this scheme into a more general (k, n) scheme, allowing for a wider range of applications and more flexibility in the sharing and reconstruction of secret images using AMBTC-based visual cryptography.

In this scheme, as illustrated in Figure 7, the encryption process starts with the requirement for p compressed AMBTC reference images for encoding. During encryption, the binary secret image is encoded into multiple AMBTC shadows using a threshold scheme denoted as (k, n). These shadows encapsulate meaningful content derived from the original image and are subsequently distributed among participants using predefined algorithms, ensuring confidentiality.

However, in this scheme, each participant is required to hold p shares, resulting in increased memory costs. This requirement was subsequently eliminated in an improved scheme proposed by X. Wu [26], which only requires one share per participant. Although there are minor variations, in the encoding phase, a binary secret image is divided into n AMBTC shadow images based on the base matrices generated by two proposed constructions. During the decoding phase, the secret image is recovered by stacking sufficient bitmaps, and partially recovering

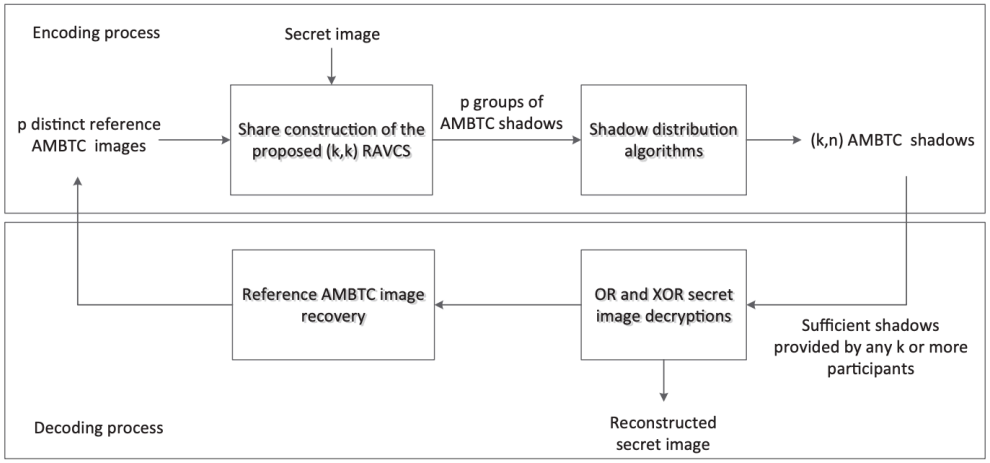


Figure 7: Framework of (k, n) AMBTC method

the AMBTC reference image. When n AMBTC shares are used, lossless reconstruction for the reference image is achieved.

5 Experimental Result

In this section we showed our resultant image using the re-implementation method.

5.1 Pattern-based

5.1.1 Naor and Shamir's

For the Naor and Shamir'pattern, all we need to do is to follow the principle mentioned in Sec. 4.1. We have to make sure the the pattern can meet our requirement to generate the correct visual effect after stacking. The following picture are the secret, share, and decrypted image for this method.

Experimental setting:

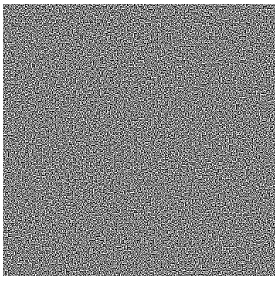
- image size: (192 × 192)
- color: binary (only black & white)

Computing resource:

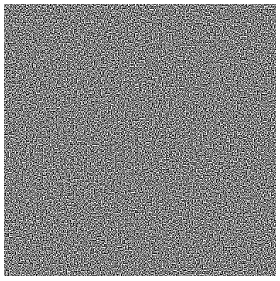
- time: 0.6286346912384033 sec.
- size: (384 × 384)
- storage: 34.6 KB × 2 shares.

(a) secret1.png

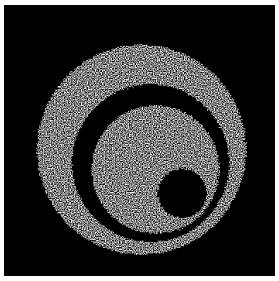
(b) secret2.png



(c) share1.png



(d) share2.png



(e) resultant1.png

(f) resultant2.png

Figure 8: Result of Naor and Shamir's method.

5.1.2 Chen's pattern

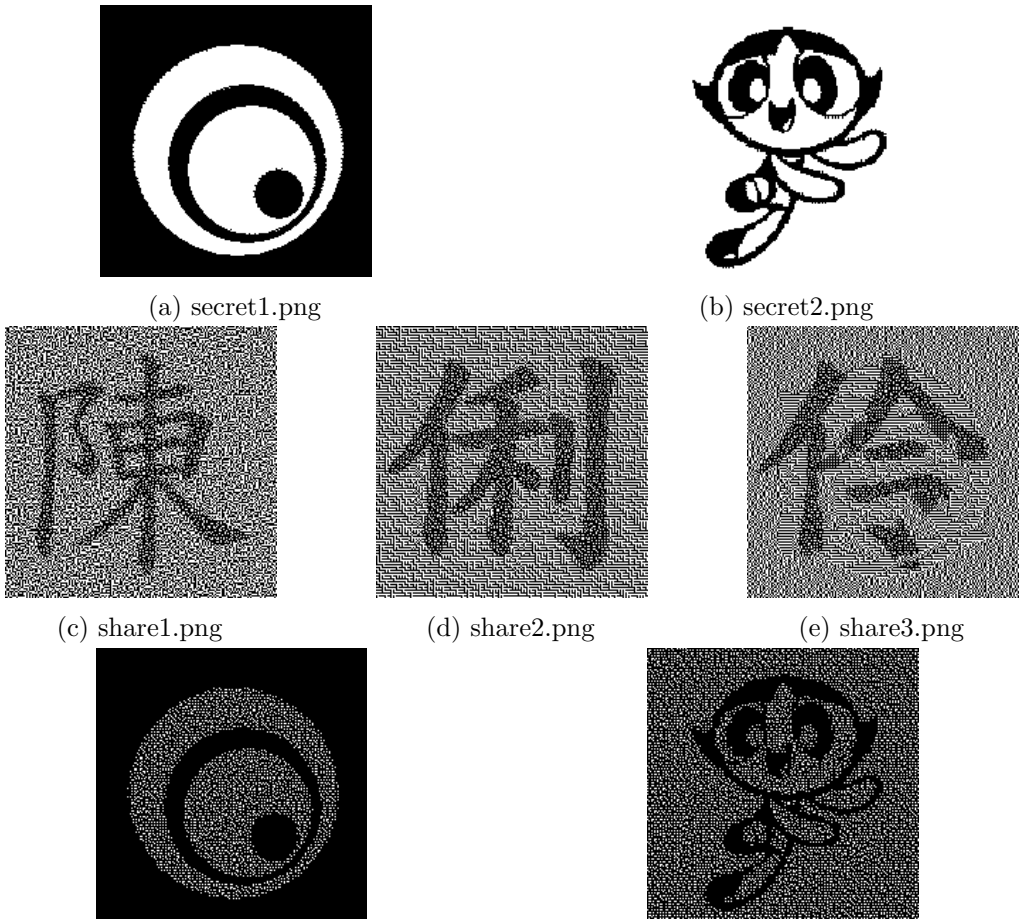
Next is Chen's pattern. We also the follow the statement mentioned in their original paper. Just like the Alg. 1, we need to maintain the black pixels number for each region to keep the the contrast for both black and white blocks. The following picture are the secret, share, and decrypted image for this method.

Experimental setting:

- image size: (192×192)
- color: binary (only black & white)

Computing resource:

- time: 1.5325465202331543 sec.
- size: (192×192)
- storage: $\sim$8 KB $\times$ n shares.



(a) secret1.png (b) secret2.png

(c) share1.png (d) share2.png (e) share3.png

(f) resultant1.png (g) resultant2.png

Figure 9: Result of Chen's pattern method.

5.1.3 IGVSS

We have different experimental setting for grayscale visual secret share method. Just like mentioned in Fig. 3, we need to do the halftoning and inverse halftoning. Error diffusion with Floyd kernel and Gaussian blur is applied here. Other details have already mentioned in the Sec. 4.1. The following picture are the secret, share, and decrypted image for this method.

Experimental setting:

- image size: (512× 512)
- color: grayscale

Computing resource:

- time: 5.696638584136963 sec.
- size: (512 × 512)
- storage: ∼155 KB × 2 shares.

(a) secret.png

(b) share1.png

(c) share2.png

(d) stacked halftoning image

(e) after inverse halftoning.

Figure 10: Result of IGVSS method.

5.2 Error-diffusion-based

5.2.1 Extended Visual Cryptography for Natural Images[17]

The following is the results of the EVC.

Experimental setting:

- image size: (256× 256)
- color: grayscale


carry image 0


carry image 1


secret image


share image 0


share image 1


decrypted secret image

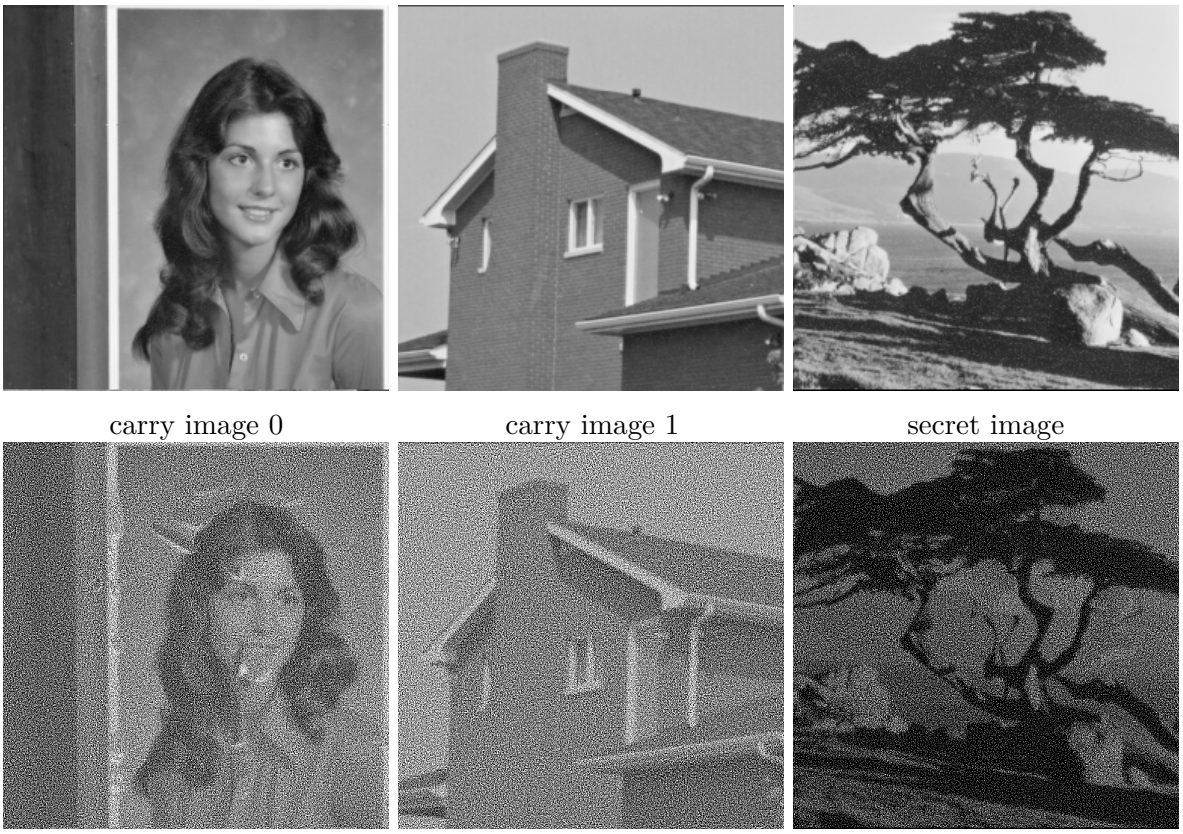
Figure 11: The result of EVC

Computing resource:

- time: 1.773 sec.
- storage: ∼131 KB × 2 shares (768 x 768).

Experimental setting:

- image size: (256× 256)
- color: grayscale



carry image 0 | carry image 1 | secret image

share image 0 | share image 1 | decrypted secret image

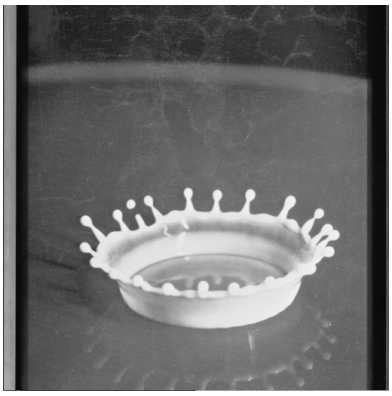
Figure 12: The result of EVC

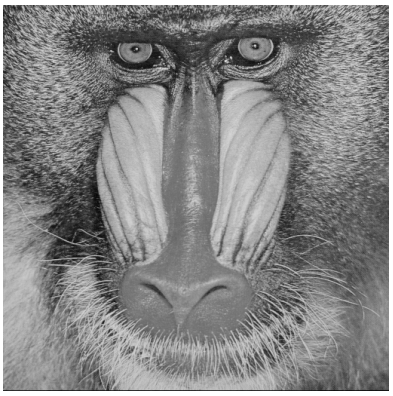
Computing resource:

- time: 1.723 sec.
- storage: ∼134 KB × 2 shares (768 x 768).

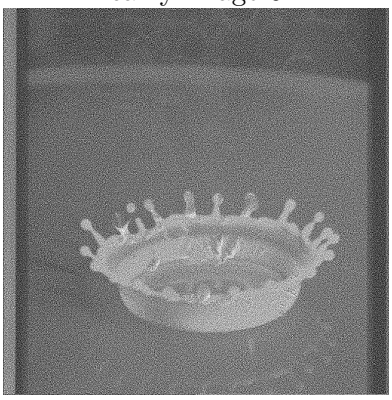
Experimental setting:

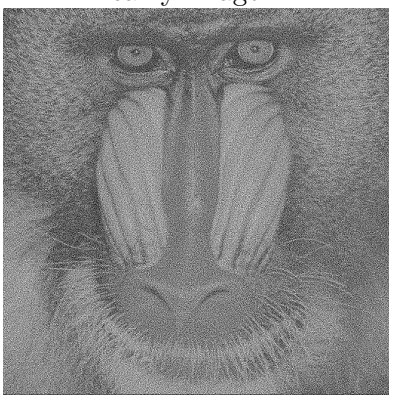
- image size: (512× 512)
- color: grayscale


carry image 0


carry image 1

secret image


share image 0


share image 1

decrypted secret image

Figure 13: The result of EVC

Computing resource:

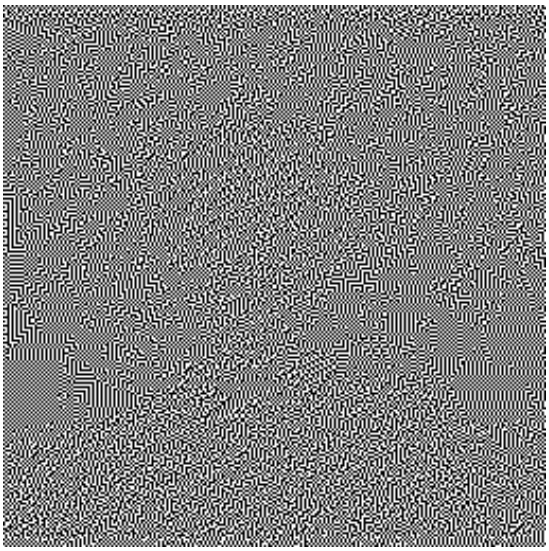
- time: 6.304 sec.
- storage: ∼533 KB × 2 shares (1536 x 1536).

5.2.2 Image Size-Preserving Visual Cryptography by Error Diffusion[11]

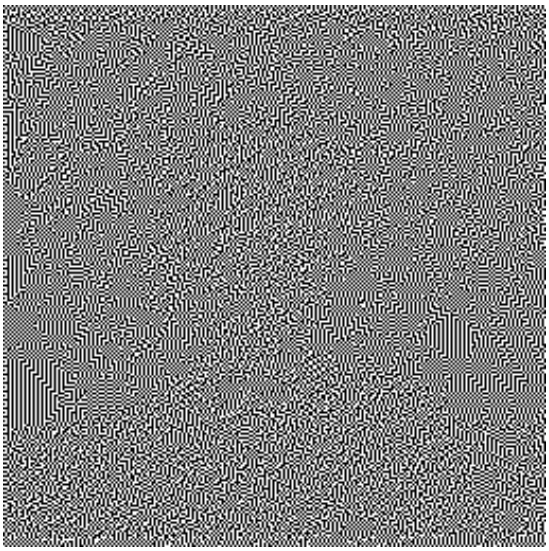
The following is the results of the Qing's method[11].

Experimental setting:

- image size: (256× 256)
- color: grayscale



share image 0



share image 1

secret image

decrypt secret image

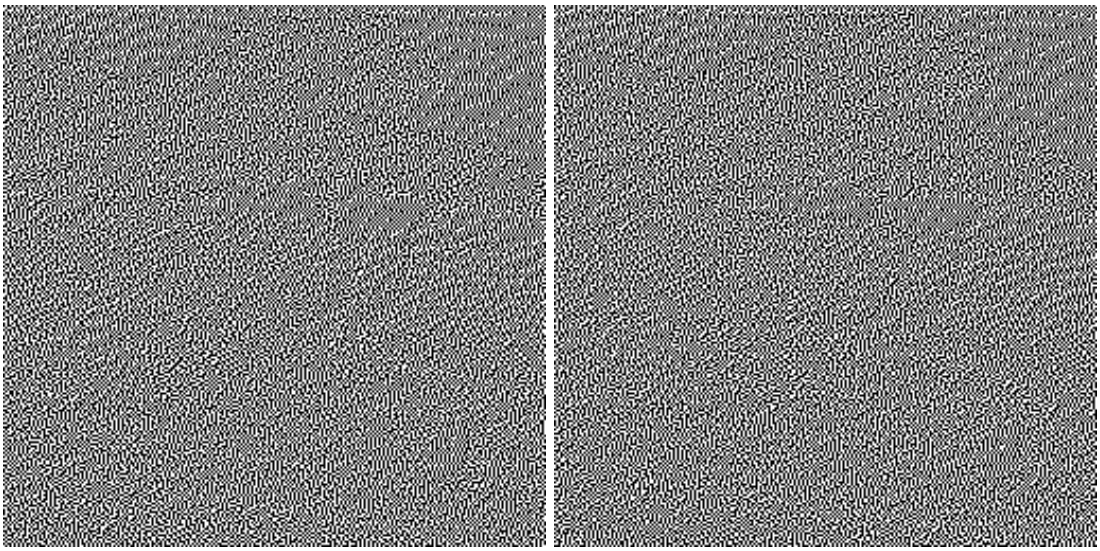
Figure 14: The result of Qing's method

Computing resource:

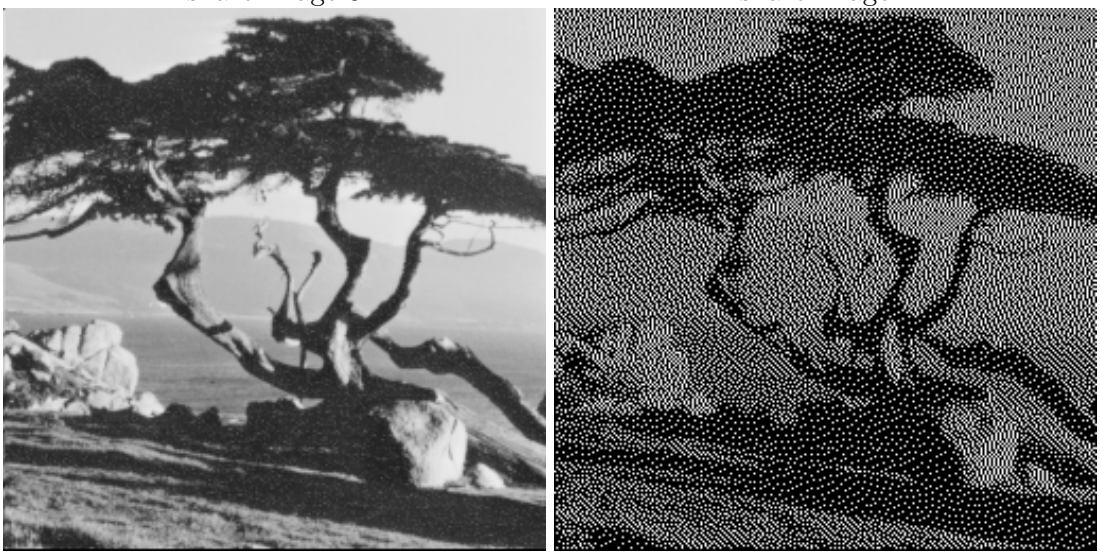
- time: 1.405 sec.
- storage: ∼15 KB × 2 shares (256 x 256).

Experimental setting:

- image size: (256× 256)
- color: grayscale



share image 0 share image 1



secret image decrypt secret image

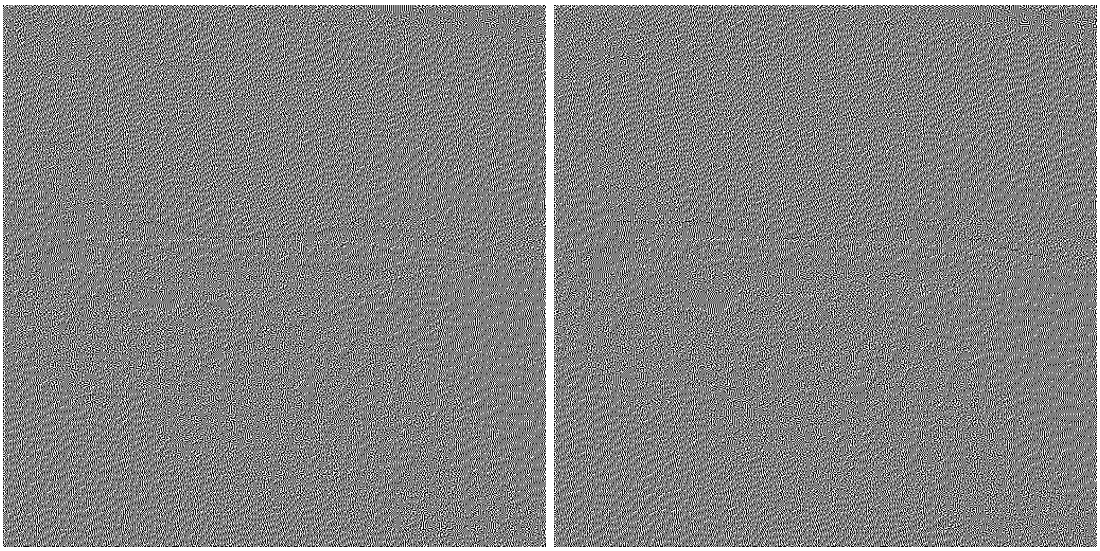
Figure 15: The result of Qing's method

Computing resource:

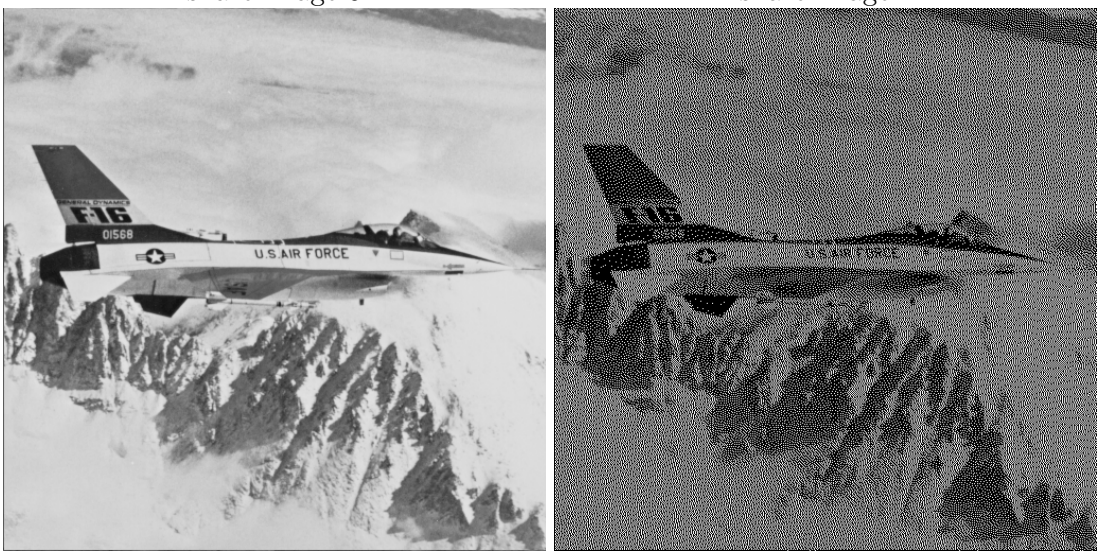
- time: 1.285 sec.
- storage: ∼15 KB × 2 shares (256 x 256).

Experimental setting:

- image size: (512× 512)
- color: grayscale



share image 0 share image 1



secret image decrypt secret image

Figure 16: The result of Qing's method

Computing resource:

- time: 4.578 sec.
- storage: ∼57 KB × 2 shares (512 x 512).

5.3 Hierarchical Visual Cryptography(HVC)

Paper [7] presents a hierarchical structure for encrypting and decrypting secret images, utilizing Vector Quantization (VQ) techniques to reduce storage overhead. In our experiments, following the experimental settings of the original paper [7], we use an image size of 512 × 512 and Shamir's (3, 6) scheme as applied to the secret image. To save space and avoid displaying meaningless shares, we have omitted the shares in our presentation.

Experimental setting:

- Secret image size: (512× 512)
- Color: Grayscale
- Shamir's (3, 6) threshold scheme

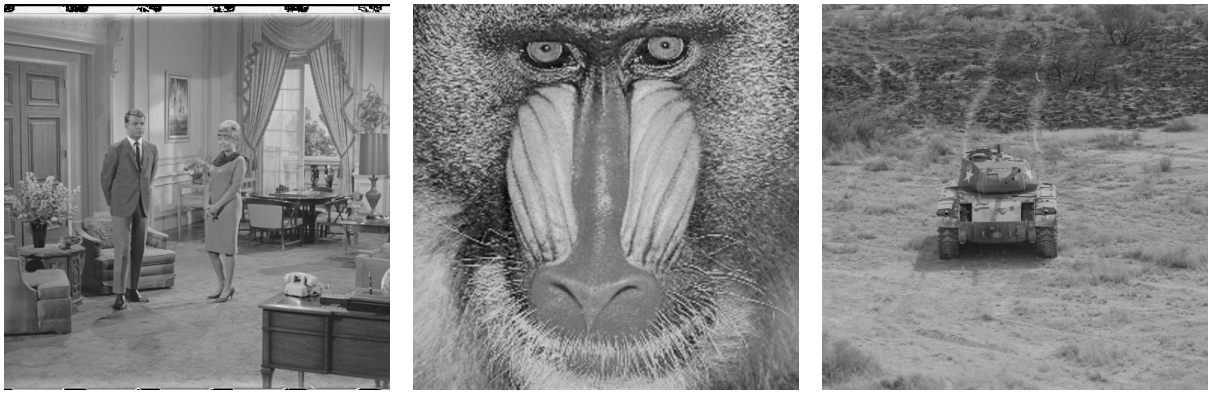


Figure 17: Secret images

couple mandril tank

Figure 18: Decypted images

Computing resource: For calculating running time, we measure only the algorithm's execution, excluding image loading and saving, and only include the encryption and decryption processes without reconstructing the codebook into shares.Additionally, we randomly select 4 level-2 shares to decrypt the secret images.

Image	Time (encrypt + decrypt)	Original	Shares (w/o VQ)	Codebook (w/ VQ)
couple	22.36 + 21.12 (sec)	257	2*6*258 (kb)	2*319 (kb)
mandril	25.31 + 20.72(sec)	257	2*6*258 (kb)	2*319 (kb)
tank	22.23 + 20.42 (sec)	257	2*6*258 (kb)	2*319 (kb)

Table 1: Execution Time and Storage

Table 1 illustrates that the encryption and decryption processes take longer compared to alternative methods due to the hierarchical structure. Furthermore, the hierarchical structure

necessitates more storage to save each share without a codebook. However, applying vector quantization to level 2 shares can significantly reduce the storage requirements.

5.4 Absolute moment block truncation coding(AMBTC) compression

Paper [20] demonstrates the use of the AMBTC method in combination with secret images. In our experiments, we use a 512 × 512 grayscale image and a 128 × 128 binary image(see figures 20 and 19). For AMBTC compression, we compress using 4 × 4 blocks, saving the AMBTC shares as PNG files.

Experimental setting:

- Carry image: Grayscale(512× 512)
- Secret image : Binary (128× 128)
- AMBTC Block: 4 × 4

Figure 19: Secret image

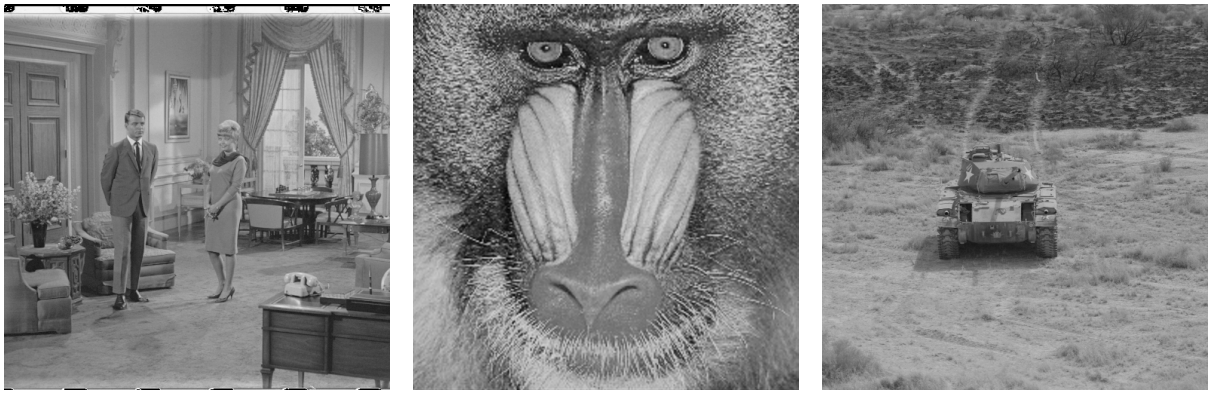
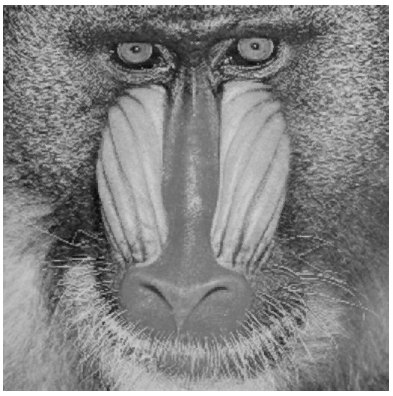


Figure 20: Carry images

After observing Figures 21 and 23 , it is evident that despite the shares having the secret image embedded, they show no noticeable difference from the original carrier images to the naked eye. Additionally, the recovery process for the carrier images (see 23) also shows no visible difference from the original carry images. However, intriguingly, the PSNR value of the shares and decrypted carrier images are identical, which may suggest the inherent properties of AMBTC coding, as mentioned in the paper [20].

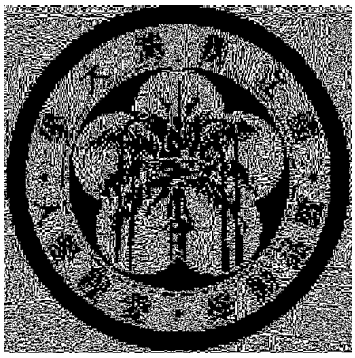
PSNR: 32.89



PSNR: 28.66

PSNR: 32.84

Figure 21: Shares

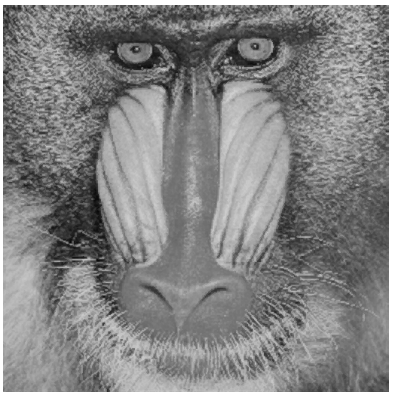


Stacking (512 × 512)

XOR (128 × 128)

Figure 22: Decrypted secret images

PSNR: 32.89



PSNR: 28.66

PSNR: 32.84

Figure 23: Recovery carry images

Furthermore, both decrypted secret image methods (see figures 22) successfully reveal the original secret image. However, the size of the decrypted image by the stacking method matches the size of the shares. In contrast, the decrypted image by the XOR method restores it to the original secret image's size and appearance.

Computing resource: For calculating running time, we measure only the algorithm's execution time, excluding image loading and saving. We include the encryption and decryption processes using both the stacking and XOR methods, as well as the recovery process for the carrier image.

Carry image	Time (encrypt+decrypt+recovery)	Original	AMBTC shares
couple	2.38+0.21+1.08 (sec)	257 (kb)	2*149 (kb)
mandril	2.30+0.22+1.07(sec)	257	2*160 (kb)
tank	2.35+0.24+1.07 (sec)	257	2*146 (kb)

Table 2: Execution Time and Storage

Table 2 demonstrates that the process and recovery times should be longer than the decrypt process because the decrypt process only requires simple operations like stacking or XOR. Additionally, it can be observed that even when there are two AMBTC shares, their memory size remains similar to that of the original carrier image. This suggests that when transmitting memory, it does not incur significant overhead.

6 Discussion

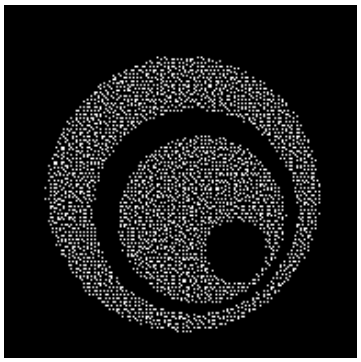
In this section we do some discussion and summarization for our report.

6.1 Most of the methods are sensitive to parameters.

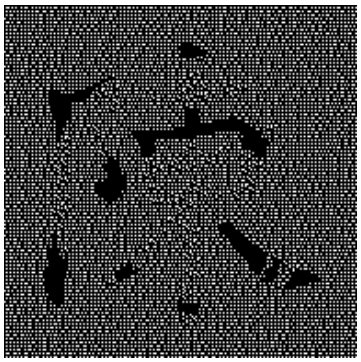
We found the result will be dramatically changed if we adjusted some specific parameter. We showed the fault error in the Fig. 24. We followed the original setting in Chen's pattern method and found the resultant image is broken. The way we repaired the error is fin-tuning the corresponding parameter with trail and error.

What's more, we find that error-diffusion-based method is highly sensitive to the value range of the input images. For example, the author from EVC[17] use the contrast enhancement to deal with this problem. For Qing et al.[11], they use the configurable parameter, α, to find the best value range as mentioned in 4.2.

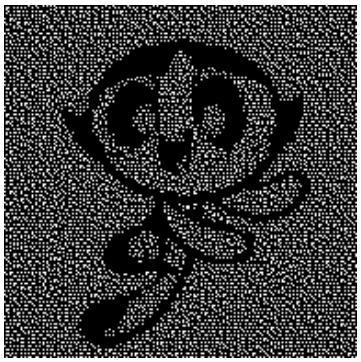
We do the ablation study about the contrast enhancement in EVC[17]. The following is the visual results. We can seen that without the contrast enhancement, we cannot properly decrypt the secret image from the share images.



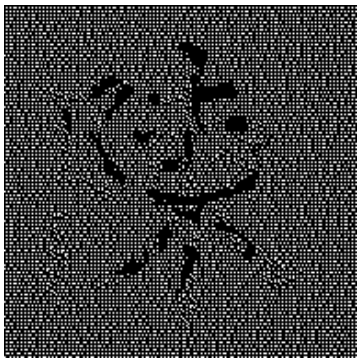
(a) correct image 1.



(b) broken image 1.



(c) correct image 2.



(d) broken image 2.

Figure 24: Some broken example for parameter setting.

carry image 0

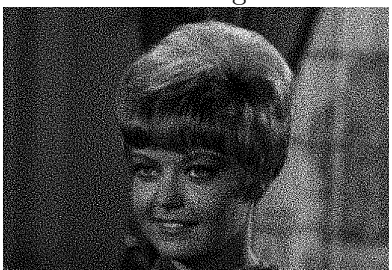
carry image 1

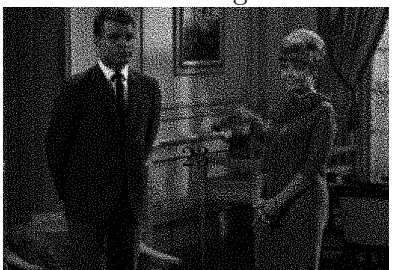
secret image

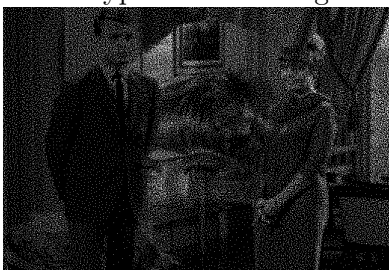
share image 0

share image 1

decrypted secret image







We have also try different α in Qing's method[11]. The following is the visual result. We can see that if α is too small, the decrypted secret image will be vague. However, if α is too large we may spot some part of the secret image from the share images.

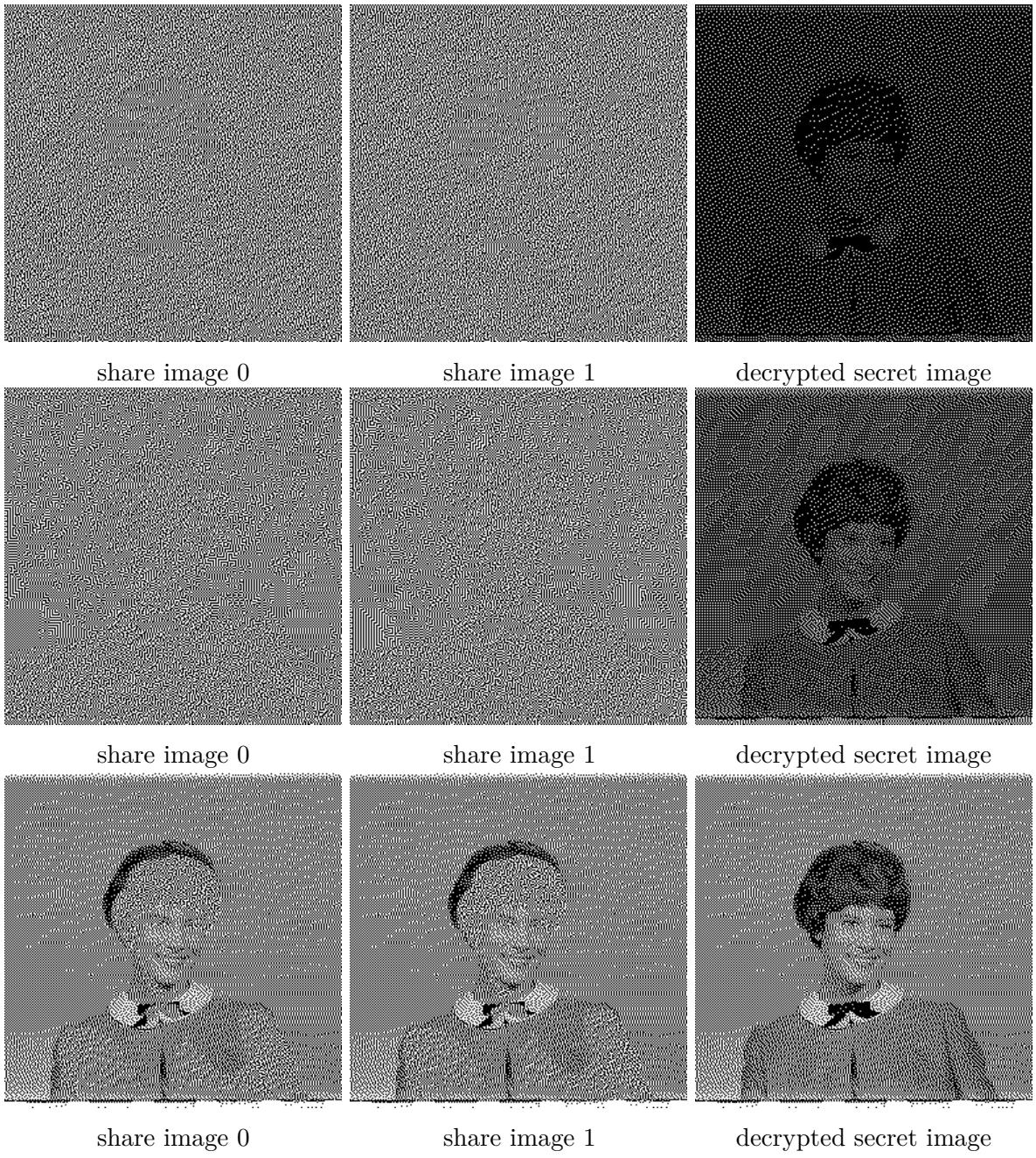


Figure 26: The effect of α in Qing's method[11]

6.2 Lack of metric for the robustness and privacy-preserving of the share images.

In past visual cryptography documents, we found there is lack of metric for the robustness and privacy-preserving of the share images. The share and decrypted image quality relied on the subjective of viewer too much. Although there are some common metrics, like Peak Signal-to-Noise Ratio (PSNR), Unified Average Changing Intensity (UACI), and Number of Pixel

Change Rate (NPCR), we still need a stronger protocol to judge whether the VC method is good enough.

6.3 The type of the image may be not mutable.

While we are conducting experiment, we found some method are not compatible i.e. some gray-scale image can not apply on the method designed for binary image, and vice versa. If we insisted on doing this, some wrong pattern will be discovered.

For example, if we apply EVC[17] on the the secret binary image with two grayscale carry images. We cannot decrypt the secret image from it. If we apply EVC on the the secret grayscale image with one binary carry image and one grayscale image, the information of the secret image can be easily spot from the share image. The 27 is the visual result.

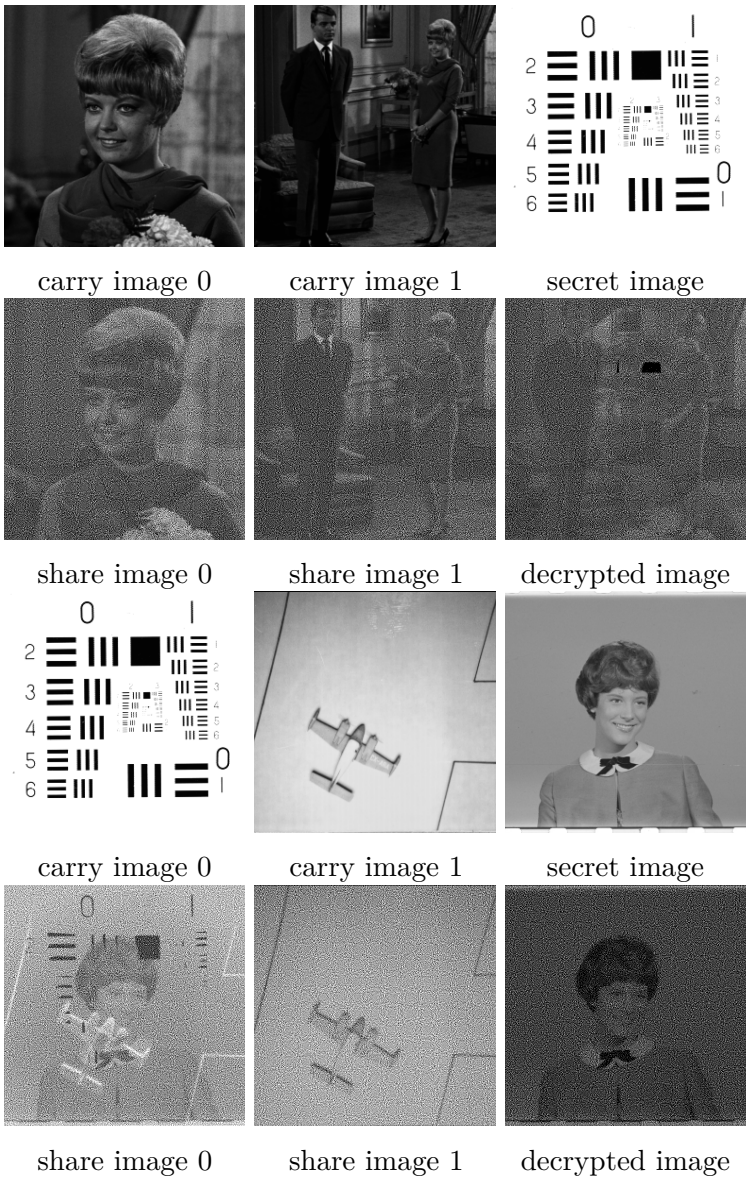
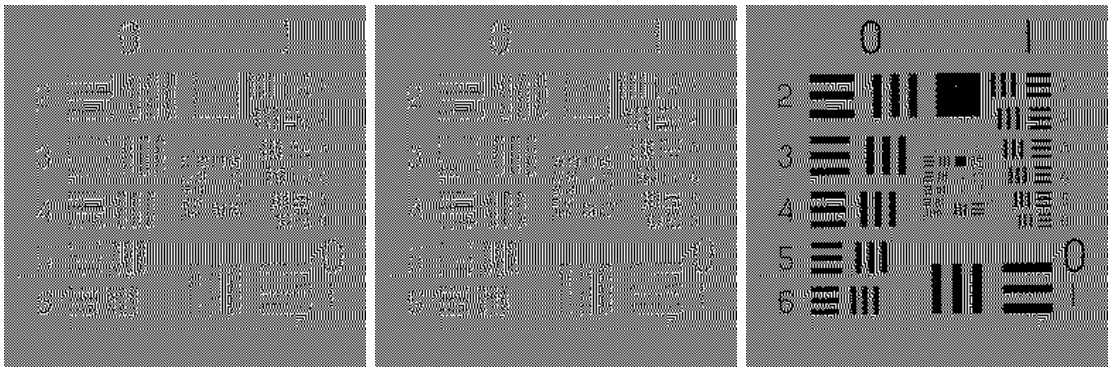


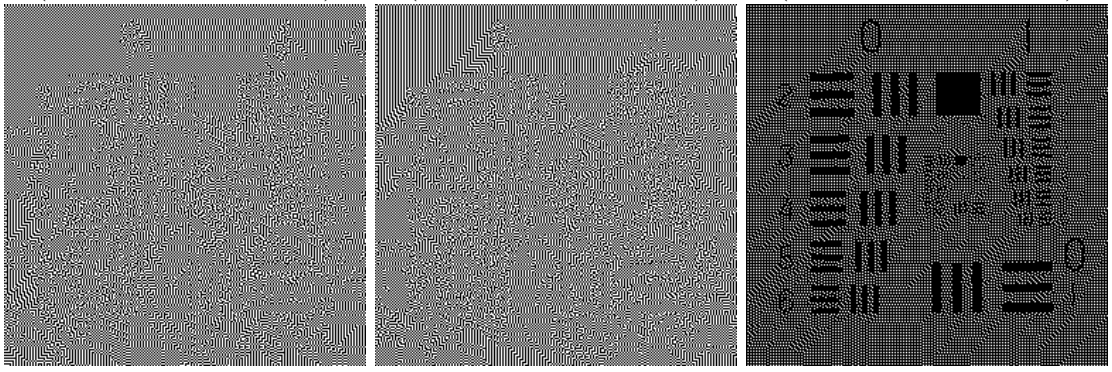
Figure 27: The result of EVC on the binary images

We have also apply Qing's method[11] on the binary secret images. We can see that the

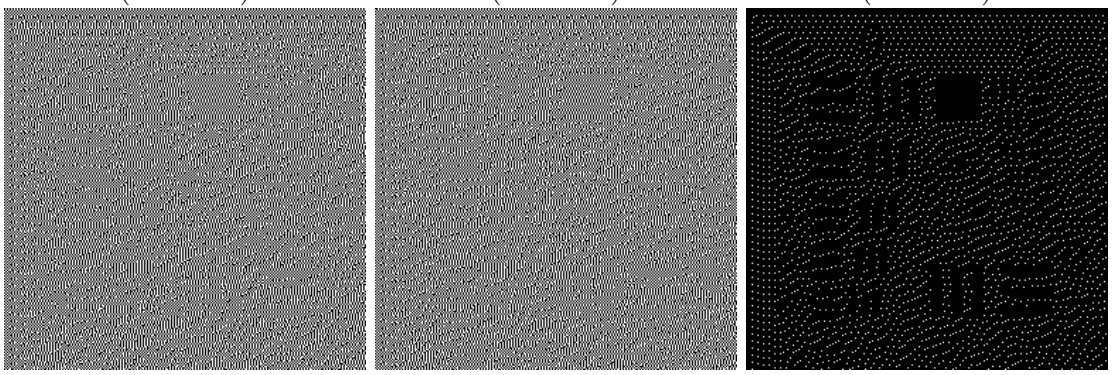
proper value of the *alpha* should totally different from the grayscale image. Even with the parameter selection, there may still be the noticeable different patterns between the white and the black block, which is not guarantee of privacy-preserving. The 28 is the visual results.



share image 0 ($\alpha = 0.5$, default value) | share image 1 ($\alpha = 0.5$, default value) | decrypt secret image ($\alpha = 0.5$, default value)



share image 0 ($\alpha = 0.25$) | share image 1 ($\alpha = 0.25$) | decrypt secret image ($\alpha = 0.25$)



share image 0 ($\alpha = 0.05$) | share image 1 ($\alpha = 0.05$) | decrypt secret image ($\alpha = 0.05$)

Figure 28: The result of Qing's method on the binary image

6.4 Comparisons

Method	Image type	(k, m)-scheme	meaningful share	pixel expansion
Naor and Shamir's[19]	binary	(2, 2)	meaningless	Yes
Chen's pattern[15]	binary	(m, m)	meaningful	No
IGVSS[2]	grayscale	(2, 2)	meaningful	Yes
EVC[17]	grayscale	(2, 2)	meaningful	yes (×3)
Qing Ye's method[11]	grayscale	(2, 2)	meaningful	no
Hierarchical VC	grayscale	2*(k, m)	meaningless	no
AMBTC	grayscale	(2, 2)	meaningful	no

Table 3: The comparision between each method

7 Conclusion

In conclusion, this study thoroughly investigates various visual cryptography (VC) methods for grayscale images, highlighting their respective strengths and weaknesses. Through detailed experimentation with pattern-based, error diffusion, hierarchical VC, and AMBTC compression techniques, it is evident that each method has unique advantages in terms of image quality, security, and computational efficiency. However, the sensitivity to parameters and the need for robust metrics to evaluate the privacy and robustness of the share images are critical areas for further research. Overall, this project provides valuable insights into the practical applications of VC, emphasizing the importance of selecting appropriate methods based on specific requirements.

8 Contributions

Name	ID	Work
Zhi-Bao, Lu	R12922196	Implementation, slide, and report of the HVC and AMBTC-based VC.
Yun-Ye, Cai	R12922104	Implementation, slide, and report of the pattern- based methods.
Pao, Yu-Wen	B0902016	Implementation, slide, and report of the error-diffusion-based methods

References

[1] L Jani Anbarasi, M Jenila Vincent, and GS Anandha Mala. A novel visual secret sharing scheme for multiple secrets via error diffusion in halftone visual cryptography. In *2011 International Conference on Recent Trends in Information Technology (ICRTIT)*, pages 129–133. IEEE, 2011.

[2] A. John Blesswin and P. Visalakshi. An improved grayscale visual secret sharing scheme for visual information security. In *2013 Fifth International Conference on Advanced Computing (ICoAC)*, pages 560–564, 2013.

[3] Pallavi V. Chavan and Mohammad Atique. Design of hierarchical visual cryptography. In *2012 Nirma University International Conference on Engineering (NUiCONE)*, pages 1–3, 2012.

[4] Pallavi Vijay Chavan, Mohammad Atique, and Latesh Malik. Signature based authentication using contrast enhanced hierarchical visual cryptography. In *2014 IEEE Students' Conference on Electrical, Electronics and Computer Science*, pages 1–5. IEEE, 2014.

[5] 101 Computing.net. Visual cryptography, 2020. `https://www.101computing.net/visual-cryptography/`.

[6] Joan Daemen and Vincent Rijmen. Aes proposal: Rijndael. 1999.

[7] Surya Das, Kaushik Das Sharma, Jayanta Chandra, and Jiten Bera. *A Hierarchical Image Cryptosystem Based on Visual Cryptography and Vector Quantization: Proceedings of ICCAIAIT 2018*, pages 3–11. 01 2019.

[8] Floyd and L. Steinberg. An adaptive algorithm for spatial grayscale. *SID Symposium.Digest of Papers,*, pages 36–37, 1975.

[9] L Hawkes, Alec Yasinsac, and C Cline. An application of visual cryptography to financial documents. *Florida State University, Florida*, pages 1–7, 2000.

[10] Chetana Hegde, S Manu, P Deepa Shenoy, KR Venugopal, and LM Patnaik. Secure authentication using image processing and visual cryptography for banking applications. In *2008 16th International Conference on Advanced Computing and Communications*, pages 65–72. IEEE, 2008.

[11] Kohei Inoue, Kenji Hara, and Kiichi Urahama. Image size-preserving visual cryptography by error diffusion. ITC-CSCC, 2018.

[12] John F Jarvis, Charles N Judice, and William H Ninke. A survey of techniques for the display of continuous tone pictures on bilevel displays. *Computer graphics and image processing*, 5(1):13–40, 1976.

[13] Don Johnson, Alfred Menezes, and Scott Vanstone. The elliptic curve digital signature algorithm (ecdsa). *International journal of information security*, 1:36–63, 2001.

[14] Keith T Knox. Evolution of error diffusion. *Journal of Electronic Imaging*, 8(4):422–429, 1999.

[15] Chen Li-Ling and Wang Shuenn-Shyang. Visual cryptography for meaningful shares. Master's thesis, Tatung University, Jan 2007.

[16] Tsung-Lieh Lin, Shi-Jinn Horng, Kai-Hui Lee, Pei-Ling Chiu, Tzong-Wann Kao, Yuan-Hsin Chen, Ray-Shine Run, Jui-Lin Lai, and Rong-Jian Chen. A novel visual secret sharing scheme for multiple secrets without pixel expansion. *Expert systems with applications*, 37(12):7858–7869, 2010.

[17] Mizuko Nakajima and Yasushi Yamaguchi. Extended visual cryptography for natural images. 2002.

[18] Moni Naor and Adi Shamir. Visual cryptography. In Alfredo De Santis, editor, *Advances in Cryptology — EUROCRYPT'94*, pages 1–12, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[19] Moni Naor and Adi Shamir. Visual cryptography. In *Advances in Cryptology—EUROCRYPT'94: Workshop on the Theory and Application of Cryptographic Techniques Perugia, Italy, May 9–12, 1994 Proceedings 13*, pages 1–12. Springer, 1995.

[20] Duanhao Ou and Wei Sun. Reversible ambtc-based secret sharing scheme with abilities of two decryptions. *Journal of Visual Communication and Image Representation*, 25, 07 2014.

[21] Anjney Pandey and Subhranil Som. Applications and usage of visual cryptography: A review. In *2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, pages 375–381, 2016.

[22] Anjney Pandey and Subhranil Som. Applications and usage of visual cryptography: A review. In *2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*, pages 375–381. IEEE, 2016.

[23] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[24] JH Saturwar and DN Chaudhari. Review of models, issues and applications of digital watermarking based on visual cryptography. In *2017 International Conference on Inventive Systems and Control (ICISC)*, pages 1–4. IEEE, 2017.

[25] Pim Tuyls, Tom Kevenaar, Geert-Jan Schrijen, Toine Staring, and Marten van Dijk. Visual crypto displays enabling secure communications. In *Security in Pervasive Computing: First International Conference, Boppard, Germany, March 12-14, 2003. Revised Papers*, pages 271–284. Springer, 2004.

[26] Xiaotian Wu, Dong Chen, Ching-Nung Yang, and Yi-Yun Yang. A (k , n) threshold partial reversible ambtc-based visual cryptography using one reference image. *Journal of Visual Communication and Image Representation*, 59, 02 2019.

[27] Ching-Nung Yang, Xiaotian Wu, Yung-Chien Chou, and Zhangjie Fu. Constructions of general (k, n) reversible ambtc-based visual cryptography with two decryption options. *J. Vis. Commun. Image Represent.*, 48:182–194, 2017.

[28] Tieyu Zhao and Yingying Chi. Hierarchical visual cryptography for multisecret images based on a modified phase retrieval algorithm. *Multimedia Tools and Applications*, 79, 05 2020.