

International Journal of Statistics and Applied Mathematics

ISSN: 2456-1452

NAAS Rating (2025): 4.49

Maths 2025; 11(7): 117-131

© 2025 Stats & Maths

<https://www.mathsjournal.com>

Received: 09-04-2025

Accepted: 14-05-2025

Penumarthi Parvateesam Murthy
Department of Mathematics,
Guru Ghasidas Vishwavidyalaya
(A Central University)
Bilaspur, Chhattisgarh, India

Amar Pandey
Department of Mathematics,
Guru Ghasidas Vishwavidyalaya
(A Central University)
Bilaspur, Chhattisgarh, India

Dolly Dewangan
Department of Mathematics,
Guru Ghasidas Vishwavidyalaya
(A Central University)
Bilaspur, Chhattisgarh, India

Corresponding Author:
Penumarthi Parvateesam Murthy
Department of Mathematics,
Guru Ghasidas Vishwavidyalaya
(A Central University)
Bilaspur, Chhattisgarh, India

Generalized norm visual cryptography scheme for secure financial data protection

Penumarthi Parvateesam Murthy, Amar Pandey and Dolly Dewangan

DOI: <https://www.doi.org/10.22271/math.2026.v11.i7b.2628>

Abstract

Within the context of banking operations in the “either or survivor” mode, a novel (n, n) visual cryptography scheme has been proposed in this paper. In this scheme, n represents the total number of generated shares. Among these, $n-1$ shares are distributed to the account holders, while one share is retained by the bank authority. Using this approach, the secret image employed for user authentication can be revealed when any one account holder combines (stacks) their share with the bank authority’s share. The share generation process follows this mechanism by considering two consecutive pixels as a single input unit during construction. Comparing this method to other approaches, it generates shares with less overhead space and might offer higher security. Like other visual cryptography methods, it is very simple to use.

This paper presents a generalized approach to norm-based visual cryptography, designed to enhance security and usability for financial applications. The proposed method extends traditional visual cryptography techniques by incorporating multiple norms, advanced encryption, and error correction mechanisms. This generalization ensures the confidentiality, integrity, and authenticity of sensitive financial information.

Keywords: Visual cryptography (VC), share, pixels, images, financial terms etc

Introduction

Through the use of complicated mathematics, visual cryptography is a cryptographic technology that makes it possible to encrypt visual data (such text or images) and decrypt it visually. Moni Naor and Adi Shamir introduced it in 1994^[9]. Visual cryptography works on the basis of dividing an image into a number of shares, each of which can be divided into a sufficient number for the original image to be visually reconstructed. However, nothing about the original image is revealed in any one share. When the shares are properly superimposed, the original image becomes visible after the visual decoding process is performed. This scheme enables the recovery of a hidden secret image without requiring any cryptographic computations. It is both highly secure and simple to implement. The basic framework includes a printed sheet containing the ciphertext (which may be mailed or faxed) and a printed transparency that serves as the confidential key. The original content is revealed by overlaying the transparency onto the ciphertext page, even though each component individually appears random and reveals no information. The approach resembles a one-time pad, as each ciphertext page is decrypted using a unique transparency. Owing to its simplicity, the method can be used by anyone without prior knowledge of cryptography or the need for computational operations.

With the novel form of visual cryptography, no cryptographic computation is required throughout the decryption process^[11]. This method requires that text or a picture be entered into the system as a digital image. The system will then produce ‘ n ’ ($3 \leq n$) distinct images, or shares, that resemble random noise. The user must stack ‘ k ’ shares, where $3 \leq k \leq n$ among ‘ n ’ shares in order to display the hidden image. Visual cryptography makes it possible to encrypt visual data such that it can only be visually decrypted without the need for computer support. While working well for simple pictures, traditional visual cryptography techniques have drawbacks when used with intricate financial data. In order to overcome these drawbacks and make norm-based visual cryptography techniques appropriate for financial data security, this study generalizes these techniques. In 1994, during the “EUROCRYPT” conference, Moni Naor, M. and Adi, S.^[9] made the initial discovery and advancement of Visual Cryptography (VC).

This unique kind of cryptography approach enables the encryption of visual data using a particular encoding style, which requires a human visual system to decipher. A simple image is divided into 'n' shareable images, which are then sent to the participants. Retrieving the plain image involves stacking the permitted subset of shares and less than shares in the stack give away nothing about the secret^[8]. Figure 1 shows a basic Visual Cryptography approach. The structure of the paper unfolds as follows: Section 2 provides some fundamental definitions for visual cryptography. An overview of the associated works is provided in Section 3. The methodology that is suggested is explained in Section 4. Section 5 provides an example of how the suggested technique generates shares. Section 6 presents a performance analysis of the suggested technique. Summaries of the work are included in Section 7.

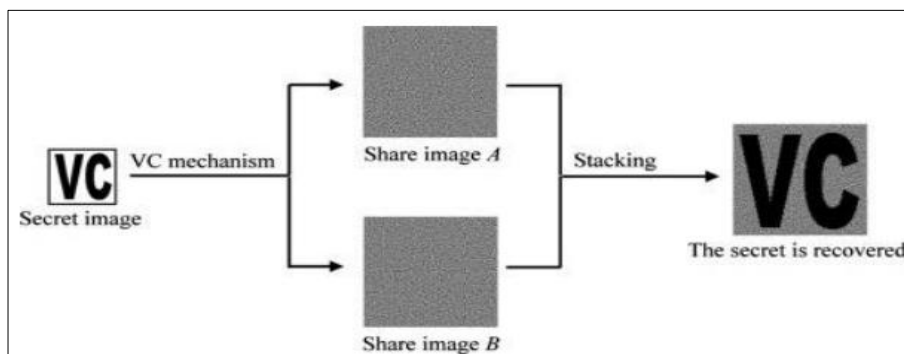


Fig 1: Basic Visual Cryptography

Preliminaries

Numerous visual cryptography systems have already been created for particular use cases. In the future, several methods could be created to address some of the issues with visual cryptography. Secret sharing schemes are another name for the visual cryptography schemes. There are various kinds of visual cryptography techniques, such as : (i) (2, 2) VC (Secret Sharing Scheme) (ii) (2, n) VC (Secret Sharing Scheme) (iii) (k, n) VC (Secret Sharing Scheme) (iv) (n, n) VC (Secret Sharing Scheme). In (2, 2) Visual Cryptography, a secret image is encrypted into two distinct shares in this basic threshold system, which when they are superimposed, discloses the hidden image. The creation of this type of access structure doesn't require any further data. However, under this system, a picture is split into two halves, and it takes both parts to put the original image back together. Lets say we have an image consisting of a black square on a white background. The goal is to divide this image into shares such that when combined, the original image is revealed. This method is often used for secret sharing, where multiple parties each hold one share and cooperation is required to reveal the secret. In (2, n) Visual Cryptography, with the use of this approach, the secret picture is encrypted into n shares, which may be overlaid to disclose the secret image when any two or more of the shares are superimposed. The user will be asked for 'n', which is the total number of people involved. On the other hand, this scheme ensures that individual shares contain no useful information about the original image, making it secure against adversaries who may gain access to only a single share. Only when two or more parties collaborate can the original image be reconstructed, ensuring privacy and security. The procedure is similar but requires more shares for bigger (2, n) visual cryptography systems. The fundamental idea is still that the original image may be seen in any two shares. Applications for this technique include watermarking, secure picture sharing, and visual authentication systems.

In (k, n) Visual Cryptography, the system generates n ($n \geq 2$) shares in this kind of visual cryptography scheme, and at least k ($2 \leq k \leq n$) shares are required to regenerate the secret data. In other words with the help of this approach, the secret image is encrypted into n shares, which can be exposed when any group of at least k shares is superimposed. The threshold k and the total number of participants n will be asked of the user. The (k, n) visual cryptography scheme offers a balance between security and flexibility. It allows for a customizable threshold for reconstruction, where the required number of shares can be adjusted based on the desired level of security and usability in different scenarios. Applications of (k, n) visual cryptography include secure multi-party image sharing, access control systems, and cryptographic key management. To retrieve the secret data, each share in the (n, n) visual cryptography scheme must be piled on top of the other. The system generates n ($n \geq 2$) shares in this type of scheme. The user will be asked for n, the total number of people involved. Stated otherwise, the secret picture is encrypted into n shares via this approach, and the hidden image will only become visible upon the combination of all n shares. We'll ask the user for the number n of participants. Because it makes sure that no single share has any important knowledge about the secret image, this approach is very helpful in situations where extreme security and privacy are necessary. Unauthorized decryption efforts are impossible since all n shares are required to reassemble the original picture. Secure picture transmission, authentication systems, and cryptographic watermarking are just a few of the domains in which perfect visual cryptography finds use.

The count of non-zero elements in a sequence of symbols is known as its weight. In binary form, the Hamming weight specifically refers to the total number of "1" bits present in the binary sequence. "Hamming weight" isn't typically used explicitly in visual cryptography, the concept of the density of information, represented by the number of black pixels in each share, plays a crucial role in ensuring security and privacy. In a (k, n) visual cryptography scheme, where any k shares can be used to reveal the secret image, the Hamming weight of each share becomes significant. By ensuring that each share has an equal number of black pixels (ideally), we can prevent individual shares from revealing any information about the secret image. Only when combined with other shares do the black pixels align to reveal the secret information?

Related works

The most basic form of visual secret sharing assumes that the message is composed of black and white pixels, with each pixel processed independently. Every original pixel is transformed into n altered versions, known as shares, with one share assigned to each transparency. Each share consists of m black and white sub-pixels placed closely together so that the human visual system blends their individual contributions into a single perceived pixel. Most visual cryptography schemes are designed to conceal only one secret image. To enhance the hiding capacity and overall efficiency of the scheme, multiple secret images can be encoded within the shares. In 1994, Moni Naor and Adi Shamir^[9] presented Visual Cryptography for the first time. In this approach, the secret message or image is partitioned into n distinct shares, each distributed to a different participant. The reconstruction of the original information requires the combination of at least k shares out of the total n . Any collection of fewer than k shares (i.e., $k - 1$ or less) is insufficient to recover the secret. For example, consider six thieves who jointly control a bank account but lack mutual trust. They divide the account password in such a way that any group of three or more thieves can retrieve it, while fewer than three cannot access the account. In the $(2, n)$ visual cryptography scheme proposed by Naor and Shamir, the secret image is split into two separate shares, ensuring that an individual share alone provides no information about the original image. However, when the two shares are stacked together, the original image is visually revealed. The method involves the use of matrices, where each share corresponds to a binary matrix. These matrices are carefully designed to ensure that the visual information is concealed in each individual share and revealed only when the shares are combined. The following $m \times n$ matrices can solve their algorithm for $(2, n)$ visual cryptography.

$$C_0 = \text{every matrix produced by switching the columns of } \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 0 & 0 & \dots & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix}$$

$$C_1 = \text{every matrix produced by switching the columns of } \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}$$

The matrices C_0 and C_1 in this instance are used to produce the white and black pixels, respectively.

In the year 2000, Tai-Wen Yue and Suchen Chiang^[15] suggested a neural network-based method for visual encryption. Using two distinct alternatives for each pixel with an equal chance, a combination of two black-and-white pixels is formed in this technique (Shown in Figure 2). In this method, a neural network is employed to create shares for the visual cryptography scheme. The neural network is trained to produce shares that meet the criteria of visual cryptography, ensuring that individual shares reveal no information about the original image, but when combined, the original image is revealed. By generating combinations of black and white pixels with equal probability for each pixel, this approach aims to enhance the security and randomness of the shares, making it more difficult for adversaries to decipher the original image from individual shares. This neural network-based approach represents a novel way to implement visual cryptography, leveraging machine learning techniques to enhance the performance and robustness of the scheme.

Pixel	Share1	Share2	Probability
			.5
			.5
			.5
			.5

Fig 2: Share generation by Tai-Wen Yue and Suchen Chiang

In 2008, Jena and Jena^[7] developed a method of visual cryptography called $(2, 2)$, in which a single pixel produces two or four pixels in each share. This innovative approach aimed to enhance the efficiency and security of visual cryptography schemes by introducing a method where a single pixel in the original image contributes to the generation of multiple pixels in the shares. The technique likely involves carefully designing algorithms or mathematical transformations that allow a single pixel to generate either two or four pixels in each share. In Figure 3, the method for creating pixels using this technology is shown. This design

ensures that the visual information from the original image is effectively distributed across the shares while maintaining the security properties of visual cryptography. By generating multiple pixels from a single pixel, this approach can potentially reduce the size of shares required for encoding the secret image, thereby improving efficiency and reducing storage or transmission overhead.

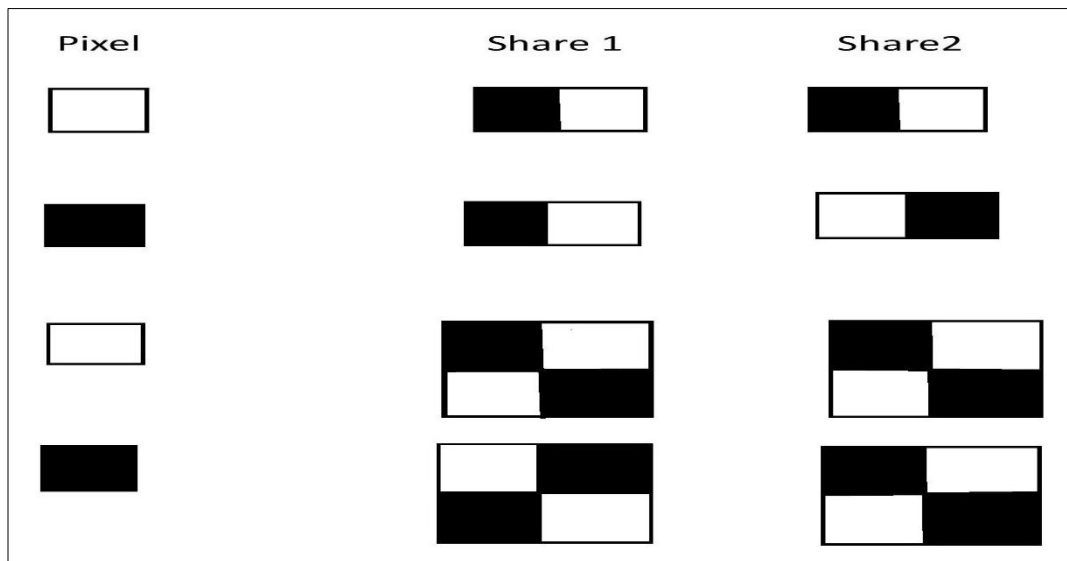


Fig 3: Share generation technique by the proposed algorithm by Jena and Jena

By 2008, Chetana Hegde, Manu S, P Deepa Shenoy, Venugopal K R, and L. M. Patnaik had devised an algorithm for visual cryptography for banking applications^[1, 5]. An effective method for verifying a customer's identity in corebanking and online banking applications was the algorithm's primary goal. Information is represented by the black pixel (represented by 1) and background by the white pixel (represented by 0). The primary objective of using visual cryptography in this context is to securely encode sensitive information, such as customer authentication data, and distribute it among multiple shares. These shares can then be distributed to different entities or parties involved in the banking transactions. For two shares in the (2, 2) scheme, the initial Boolean matrices for the white pixel, S_0 , and the black pixel, S_1 , are provided below.

$$S_0 = \begin{bmatrix} 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \end{bmatrix}$$

$$S_1 = \begin{bmatrix} 1 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 1 \end{bmatrix}$$

On various Constructions of Monochrome Visual Cryptographic Schemes, a (2, n) visual cryptographic scheme, was proposed by Avishek Adhikari and Bimal Roy in 2008^[2]. The term "monochrome" implies that the shares are represented using only one color channel, typically black and white. Each pixel in the shares is either black (representing one part of the secret image) or white (representing another part or background). The construction of the (2, n) visual cryptographic scheme by Adhikari and Roy likely involves carefully designing algorithms or mathematical techniques to divide the secret image into shares while preserving the security properties of visual cryptography.

Using two consecutive images captured at a time as the single input, Jayanta Kumar Pal, J. K. Mandal, and Kousik Dasgupta devised a (2, 2) Visual cryptography in 2010^[10]. This approach likely aimed to enhance the efficiency and security of the visual cryptography scheme by processing pairs of consecutive pixels together. In a traditional (2, 2) visual cryptography scheme, the original secret image is split into two separate shares, and each share typically corresponds to a binary image where each pixel is either black (representing one part of the secret image) or white (representing another part or background). Figure 4 shows a method for creating pixels using this technique.

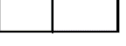
Pixels	Share	Share	Probability
	 	 	.5
			1
			1
	 	 	.5

Fig 4: Share generation method by Jayanta Kumar Pal, J. K. Mandal, and Kausik Dasgupta

There are numerous alternative algorithms accessible for diverse visual cryptography techniques ([6, 17, 4, 12]), wherever security has been improved. These efforts include not only improving security features but also focusing on making the schemes more user-friendly and practical for real-world applications. It is also possible to determine from the literature that attempts have been made to make visual cryptography more user-friendly. Wei-Qi [16], for instance, created a plan to ensure that the shares were aligned properly. This visual cryptography is a more practical and widely applicable solution for secure image sharing, authentication, and other cryptographic applications.

In 2010 [11], A (2, n) visual cryptography technique for banking applications is designed to provide secure access to joint bank accounts, specifically in situations where there are two account holders, and both have equal authority in managing the account. In a (2, n) visual cryptography scheme, the original secret image (representing access credentials or authentication information in this context) is divided into n shares, with each account holder receiving one share. Both shares are necessary to reconstruct the original secret information, ensuring that neither account holder can access the account independently [3]. The (2, n) visual cryptography technique developed for banking applications in 2010 offers a practical and secure solution for managing joint bank accounts, providing equal access to account holders while ensuring the confidentiality and integrity of access credentials.

This study presents a novel visual cryptography method that could be helpful in secure banking scenarios without access to computers. One user may participate in the banking process against an account with the previously proposed algorithm by Chetana Hegde *et al.* [5], but the proposed approach can also be used for accounts held jointly by several account holders. It provides a versatile approach that can accommodate both single-user and joint-account scenarios while maintaining the confidentiality and integrity of banking information.

The Methodology

The proposed technique treats (n=3) consecutive pixels of the source image as a single input, resulting in eight possible cases for those pixel triplets, each representing a combination of different cases : (i) Black, Black and Black, (ii) Black, Black and White, (iii) Black, White and Black, (iv) Black, White and White, (v) White, Black and Black, (vi) White, Black and White, (vii) White, White and Black, (viii) White, White and White.

Similarly the above technique follow the four and five and so on. The technique considers four consecutive pixels as a single input in the source image, resulting in sixteen possible cases for these fourth of Pixels, each representing a combination of different cases: (i) Black, Black, Black and Black, (ii) Black, Black, Black and White, (iii) Black, Black, White and White, (iv) Black, White, White and White, (v) White, Black, White and White, (vi) White, White, Black and White, (vii) White, White, White and Black, (viii) Black, Black, White and White (ix) White, Black, Black and White, (x) White, White, Black and Black, (xi) Black, White, Black and White, (xii) White, Black, White and Black, (xiii) Black, White, White and Black, (xiv) Black, White, Black and Black, (xv) Black, Black, White and Black, (xvi) White, White, White and White.

For a (3, n) visual cryptography scheme, where “n” shares are generated, each share contributes to the reconstruction of the secret information. The position of (3, n) visual cryptography for Hamming weight purposes can be considered in the following aspects:

1. The Hamming weight of corresponding pixels across shares should be consistent. In other words, for a given position in each share, the Hamming weight (number of black pixels or non-zero bits) should be the same.
2. While consistency is important, introducing variations in Hamming weights, especially for certain pixel values (e.g. black pixels), can contribute to security. For instance, if the Hamming weight for a black pixel in one share is significantly different from the corresponding pixels in other shares, it may make it more challenging for an attacker to discern the secret information without possessing the requisite shares.
3. The Hamming weight considerations also impact the clarity of the reconstructed secret image. This scheme should ensure that the Hamming weights contribute to a visually clear and accurate reconstruction when the shares are stacked.
4. While introducing variations in Hamming weights can enhance security, it's essential to balance this with practicality. Excessive complexity may impact the ease of implementation, decoding speed, and the overall user experience.

Also similarly the above points (expects) for the $(4,n)$, $(5,n)$ (n,n) visual cryptography Technique.

Let N represent how many people are involved, or how many account holders there are. Where n is the total amount of shares and m is the integer portion of $(n/2)$. The relation ${}^nC_m \geq \min(N + 1)$, where C is the combination operation, must hold, and the bank authority must determine what number of n to use.

The Hamming weight associated with each block of every share $(H) = \text{Integer part of } {}^nC_m/2$;

Now let us consider the eight possible cases of input pixels:

- **Black, Black and Black:** This case creates a concentrated area of black pixels, contributing to a visually prominent and distinguishable pattern in the output block when shares are stacked.
- **Black, Black and White:** Initial consecutive black pixels create a dark region with a transition to white, influencing the visual structure of the output block.
- **Black, White and Black:** The central white pixel is flanked by black pixels, creating a distinctive pattern with contrast in the output block.
- **Black, White and White:** The black pixel in the first position creates an initial emphasis on a dark region, leading to a distinct pattern in the output block.
- **White, Black and Black:** Two consecutive black pixels contribute to a distinctive and concentrated dark region in the output block.
- **White, Black and White:** The central black pixel is surrounded by white pixels, creating a visually unique structure in the output block.
- **White, White and Black:** The black pixel at the end introduces a concentrated black area, affecting the visual appearance of the output block.
- **White, White and White:** All three consecutive pixels are white. This case results in a fully white segment, providing a unique pattern in the output block.

If the input image contains an odd number of pixels, the final pixel is stored separately in parts.

Proposed Algorithm for (n,n) - Norm VC :

Input:

- Source image (binary or grayscale, assuming binary for simplicity: Black=0, White=1)
- Number of consecutive pixels in a group $k = n$ (block size and shares), which could be 3, 4, 5, etc.
- Number of shares (n) to be generated.
- Number of participants (N) .
- Threshold m for reconstruction (integer portion of $n/2$).

Output: n shares that, when combined, reconstruct the original image.

Procedure

Step 1: Preprocessing

- Convert the image to a binary format (if it's not already). Each pixel should either be 0 (black) or 1 (white).
- Check the number of pixels in the image.
 - (i) If the number of pixels is odd, handle the last pixel separately by padding or storing it in a separate structure.

Step 2: Group consecutive pixels

- Divide the image into blocks of k consecutive pixels:
 - If $k = n = 3$, each block contains three consecutive pixels (e.g., B, B, W for Black, Black, White).
 - If $k = n = 4$, each block contains four pixels, and so on.
 - For each block, store the pixel combination.

Step 3: Generate shares

- **Initialize n empty shares.** Each share will have the same dimensions as the original image.
- **For each block of consecutive pixels:**
 - For each of the 8 possible combinations (in the case of $k = n = 3$) or 16 combinations (in the case of $k = n = 4$), map the combination to an appropriate pattern of black and white pixels for the shares.
 - Example (for $k=n=3$)**
 - If the block is (B, B, B), apply the corresponding black/white distribution across all shares.
 - Use randomness in distributing black and white pixels across shares, making sure the Hamming weight requirements are met (i.e., for each block in each share, the number of black pixels should be consistent across shares for each corresponding position).
- **Ensure consistency in Hamming weight:** For each pixel in the block, ensure that the Hamming weight across all shares is consistent, i.e., the same number of black pixels appears across the shares.
- **Introduce controlled variations (for security):** Add some variation in how black and white pixels are distributed across shares, especially for certain pixel values, to increase the security of the reconstruction process.

Step 4: Hamming weight computation

1. Calculate the Hamming weight:
 - i) For each block across shares, calculate the Hamming weight of corresponding pixel positions in the shares.
 - ii) If using an (n, n) scheme, the formula $H = \text{Integer part of } \binom{n}{m}/2$ can help define the target Hamming weight for each pixel block.

Step 5: Stacking for reconstruction

1. Simulate the combining of shares:
 - i) Regenerate the original image by combining the n shares. Ensure that black pixels from different shares overlap appropriately, and the white regions are maintained in the output.
 - ii) The final reconstructed image should closely resemble the original.

The pseudo code for the above steps are given in Algorithm [Algo-1].

Algorithm 1 Algorithm (n, n) Norm Visual Cryptography Procedure For Financial Use

```

1: procedure Make Functions (VC_k_equals_n ( $n, m, N$ )) -
2:   INPUT :
3:     • Image : binary (0 = Black, 1 = White),  $H \times W$ 
4:     •  $k = n$  : block size = share
5:     •  $n$  : shares to generate
6:     •  $N$  : participants ( $N \geq n$ )
7:     •  $m$  : threshold  $\approx n/2$ 
8:   OUTPUT : shares [1... $n$ ] ( $H \times W$  binary images)
9:   BEGIN
10:    Converted _ image  $\leftarrow$  Convert to Binary (image)
11:    Pad Image (converted _ image,  $n$ ) // Multiple of  $n$  pixels
12:    blocks  $\leftarrow$  Group Into Blocks (converted _ image,  $n$ )
13:    shares[1... $n$ ]  $\leftarrow$  Initialize Empty Shares ( $H, W, n$ )
14:    FOR each block  $\in$  blocks DO // block =  $n$  pixels
15:      block _ pattern  $\leftarrow$  Get Block Pattern (block) //  $2^n$  patterns
16:      // Lookup  $n \times n$  patterns ensuring Hamming weight  $\geq m$  per column
17:      patterns  $\leftarrow$  Lookup Hamming Weight Patterns (block_pattern,  $n, m$ )
18:      for  $i \leftarrow 1$  to  $n$ ; do // Each share
19:        for pos  $\leftarrow 0$  to  $n - 1$ ; do // Each position in block
20:          block _ idx  $\leftarrow$  Calculate Block Index (block, pos)
21:          shares [ $i$ ][block _ idx]  $\leftarrow$  patterns [ $i$ ][pos]
22:        end for
23:      end for
24:      Verify Hamming Weight (shares, block,  $n, m$ )
25:      Apply Controlled Randomness (shares, block _ pattern)
26:    end for
27:    FOR each block  $\in$  blocks;
28:      for pos  $\leftarrow 0$  to  $n - 1$ ; do
29:        hammingweight  $\leftarrow$  Calculate Hamming Weight (shares [1... $n$ ], pos)
30:        if hamming _ weight  $< m$  then
31:          Adjust Share Distribution ( shares, pos,  $m$ )
32:        end if
33:      end for
34:    end for
35:    RETURN shares [1... $n$ ]
36:  END
37:  FUNCTION Reconstruct Image (shares [1... $n$ ]);
38:    reconstructed  $\leftarrow$  Empty Image ( $H, W$ )
39:    for  $i=0$  to  $H-1$ ; do
40:      for  $j=0$  to  $W-1$ ; do
41:        black_count  $\leftarrow 0$ 
42:        for  $p=1$  to  $n$ ; do
43:          if shares [ $p$ ][ $i$ ][ $j$ ] == 0 : black_count ++ then
44:            reconstructed [ $i$ ][ $j$ ]  $\leftarrow$  (black_count  $\geq m$ ) ? 0 : 1
45:          end if
46:        end for
47:      end for
48:    end for
49:    RETURN reconstructed
50:  ENDFUNCTION
51:  FINANCIAL _ AUTH(user _ share -  $n$ , bank _ share [1... $n-1$ ]);
52:  all _ share  $\leftarrow$  bank _ shares + user _ share -  $n$ 
53:  recon  $\leftarrow$  ReconstructImage(all _ shares)
54:  RETURNSSIM(recon, stored _ secret)  $> 0.95$  ? "SUCCESS" :
55:  "FAILED"
56: endprocedure
57: end for

```

Samples illustrative

For better understanding, let's look at an example of the suggested algorithm. Assuming that three people wish to start the account together, four shares will need to be created (three per user plus one for the bank authority). So, here $(N + 1) = 4$, implying $N = 3$, where N is the total number of users or participants in this cryptographic scheme. Now consider $n = 4$, therefore $m =$ integer part of $(4/2) = 2$. Now, $4C_2 = 6 > (N + 1)$. So, this choice of n is appropriate, and a (3,4) visual cryptography scheme will be used. Now, $(N + 1) = 6$, implying $N = 5$, where N is the total number of users or participants in the visual cryptography system. Now consider $n = 6$, therefore $m =$ integer part of $(6/2) = 3$. Now, $6C_3 = 20 > (N + 1)$. So, this value of n is suitable, and a (3,6) visual cryptography technique will be used. $N + 1 = 10$, implying $N = 9$, where N represents how many users or participants there are in the visual cryptography scheme. consider $(n = 10)$, $m =$ integer part of $(10/2) = 5$. Now, $10C_5 = 252 > (N + 1)$, as $(N + 1)$ is 10. The value of $n = 10$ is acceptable for a (3,10) visual cryptographic scheme. We can create shares for banking purposes based on the aforementioned data. The following figure 6 illustrates the share generation for three joint account holders.

The Figure 5 can be represented using a matrix. Let's say there are eight matrices, $C_1, C_2, C_3, C_4, C_5, C_6, C_7$ and C_8 would show how many shares the collection of pixels generated that is black, white and black, black, black and white, white, white and black, white, white and white, black, black and black, white, black and white, white, black and black, 0 for the white pixel and 1 for the black pixel.

Similarly there are sixteen matrices, $C_1, C_2, C_3, C_4, C_5, C_6, C_7$ and C_8 which represent the share generation of the set of pixels black, white and black, black, black and white, white, white and black, white, white and white, black, black and black, white, black and white, white, black and black, 0 for the white pixel and 1 for the black pixel.

$$C_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} C_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} C_3 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} C_4 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$C_5 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} C_6 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} C_7 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix} C_8 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

This is corresponding matrix for (4,n) visual cryptography technique:

$$C_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} C_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} C_3 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} C_4 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$$C_5 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} C_6 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} C_7 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix} C_8 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

$$C_9 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{bmatrix} C_{10} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} C_{11} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix} C_{12} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$C_{13} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix} C_{14} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} C_{15} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \end{bmatrix} C_{16} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Similarly we can find other matrix for (5,n), (6,n)..... below matrix is represent the (n,n) visual cryptography technique.

$$C_1 = \begin{bmatrix} 1 & 0 & 0 & \dots & \dots & \dots \\ 0 & 1 & 0 & \dots & \dots & \dots \\ 0 & 0 & 1 & \dots & \dots & \dots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \dots & 1 \end{bmatrix} \quad C_2 = \begin{bmatrix} 1 & 0 & 0 & \dots & \dots & \dots \\ 1 & 0 & 0 & \dots & \dots & \dots \\ 1 & 0 & 0 & \dots & \dots & \dots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & \dots & \dots & 0 \end{bmatrix} \quad C_3 = \begin{bmatrix} 0 & 1 & 0 & \dots & \dots & \dots \\ 0 & 1 & 0 & \dots & \dots & \dots \\ 0 & 1 & 0 & \dots & \dots & \dots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 1 & 0 & \dots & \dots & 0 \end{bmatrix}$$

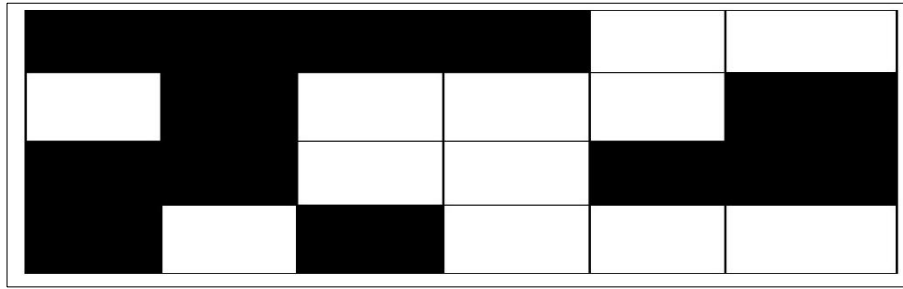
$$\dots\dots\dots C_n = \begin{bmatrix} 0 & 0 & 0 & \dots & \dots & 1 \\ 0 & 0 & 0 & \dots & \dots & 1 \\ 0 & 0 & 0 & \dots & \dots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \dots & 1 \end{bmatrix}$$

we see that all visual technique (2,n), (3,n).....(n,n) can represent the matrix and there are n matrices C_1, C_2, C_3 C_n , which represent the process of generating shares from the black and white pixel, where 0 for the white pixel and 1 for the black pixel.

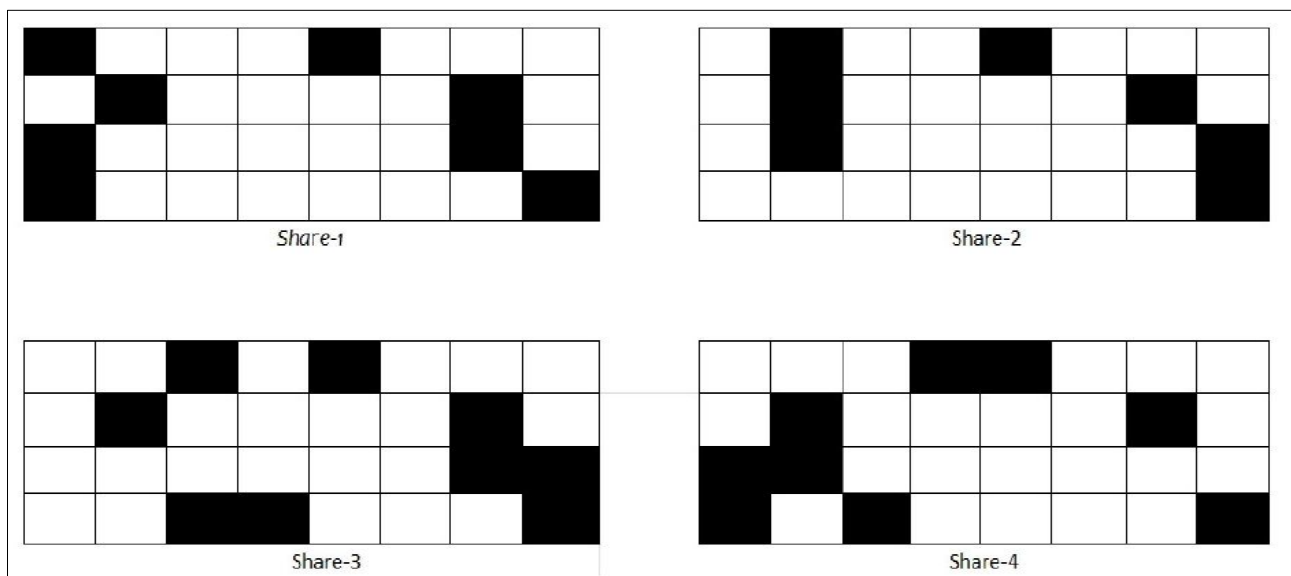
Pixels	Share-1	Share-2	Share-3	Share-4	Probability of occurrence
					1/4
					1/4
					1/4
					1/4
					1
					1
					1
					0.5
					0.5
					0.5
					1/4
					1/4
					1/4
					1/4

Fig 5: Share generation method by (3,n) VC technique

For better understanding, let's look at an example of the suggested algorithm. Take into consideration the image in Figure 6 as the input image.

**Fig 6:** Input Image

If we read the image in Figure 6 in row major order first we are getting three black pixels, followed by one black and two white, among next three, one white, black and white, after that two white and one black pixels, again two black and one white, followed by one white and two black, among next three black, white and black pixels and lastly three white pixels. The generated shares by the proposed algorithm are given in the Figure 7 Share 1 to 4 respectively.

**Fig 7:** Share 1 - 4

Now, let's apply the suggested method to a real-life image as depicted in Figure 8. We'll generate four shares represented by four transparencies following the figure. Upon stacking any two of these shares with precise alignment, it becomes apparent that the concealed image emerges, albeit with some background noise.

**Fig 8:** Input Image**Fig 9:** Share 1



Fig 10: Share 2

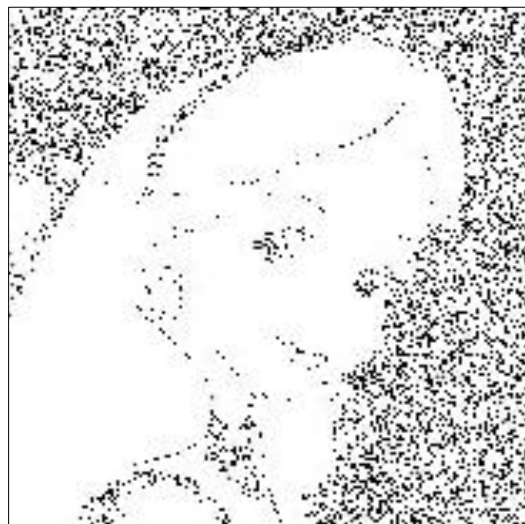


Fig 11: Share 3



Fig 12: Share 4



Fig 13: Share 1 + share 2



Fig 14: Share 2 + share 3



Fig 15: share 3 + share 4



Fig 16: Share 4 + share 1



Fig 17: Reconstruct original image

Similarly we can extend this method for $(4,n)$, $(5,n)$, $(6,n)$ (n,n) . The example of $(4,n)$ visual cryptography pixels are given below in figure 18.

Pixels	Share-1	Share-2	Share-3	Share-4	Share-5	Probability of accuracy
						$1/5$
						$1/5$
						$1/5$
						$1/5$
						$1/5$
						1
						1
						1
						1
						$1/2$
						$1/2$
						$1/2$
						$1/2$
						$1/2$
						$3/4$
						$3/4$
						$3/4$
						$3/4$
						$1/5$
						$1/5$
						$1/5$
						$1/5$
						$1/5$

Fig 18: Share generation method for $(4,n)$ visual cryptography technique

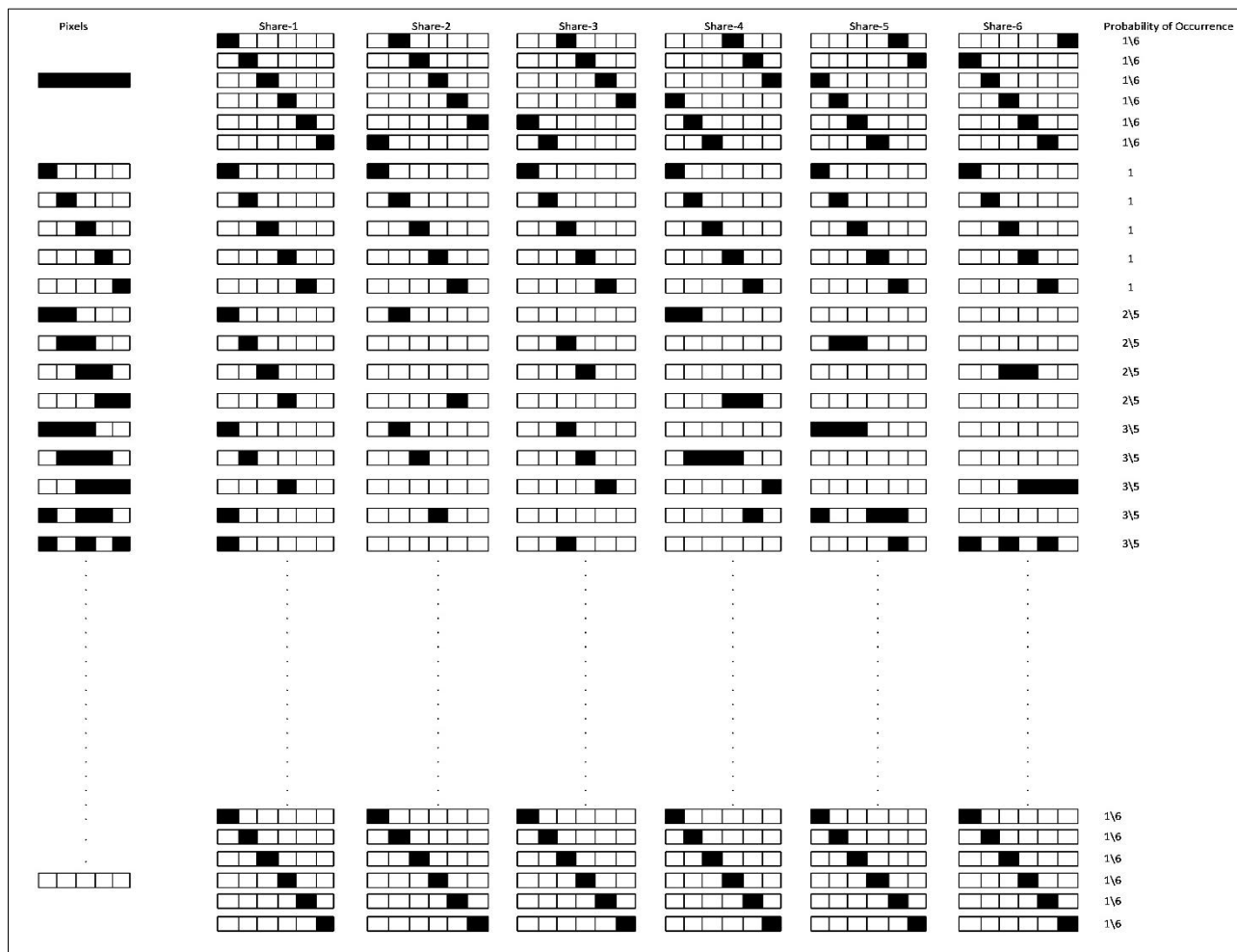


Fig 19: Share generation method for (5,n) visual cryptography technique

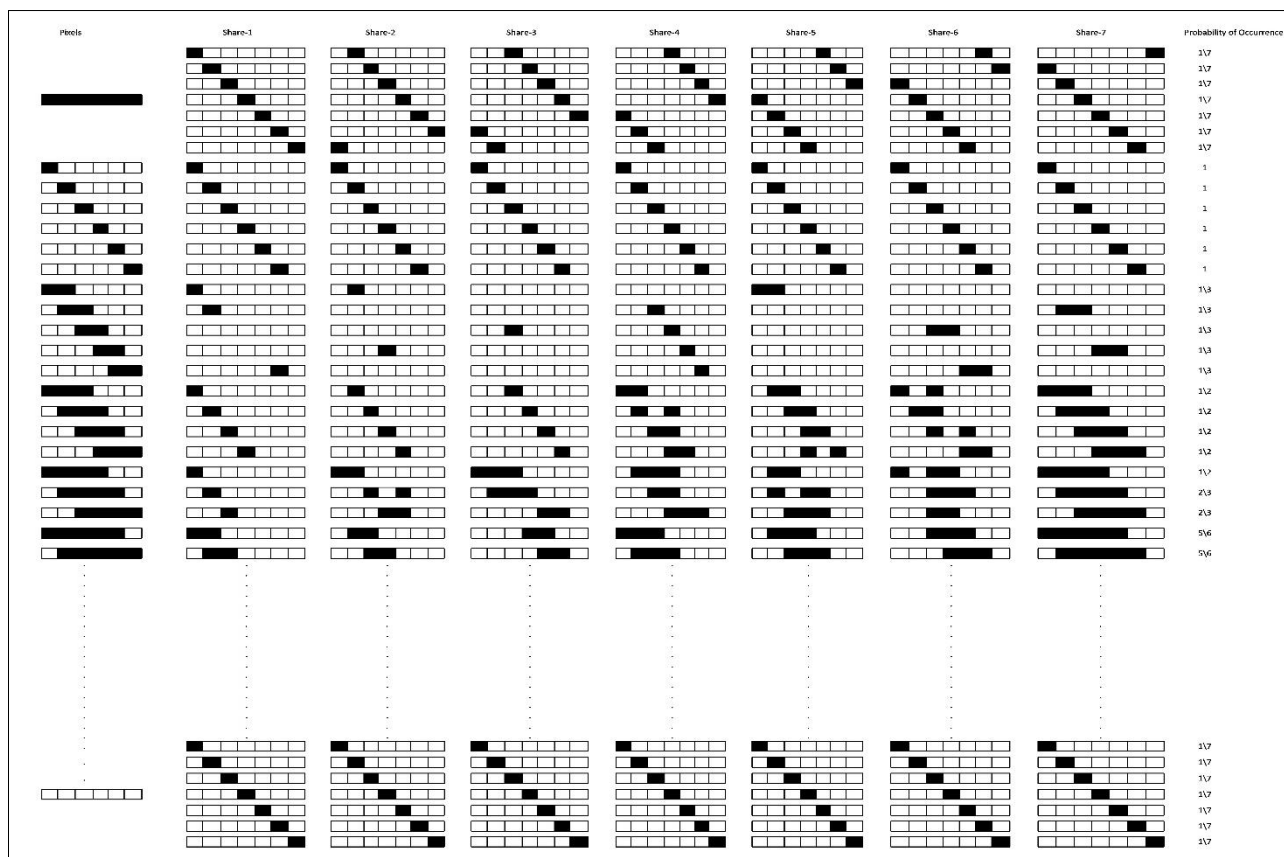


Fig 20: Share generation method for (6,n) visual cryptography technique similarly we give an example of (n,n) share generation method given below:

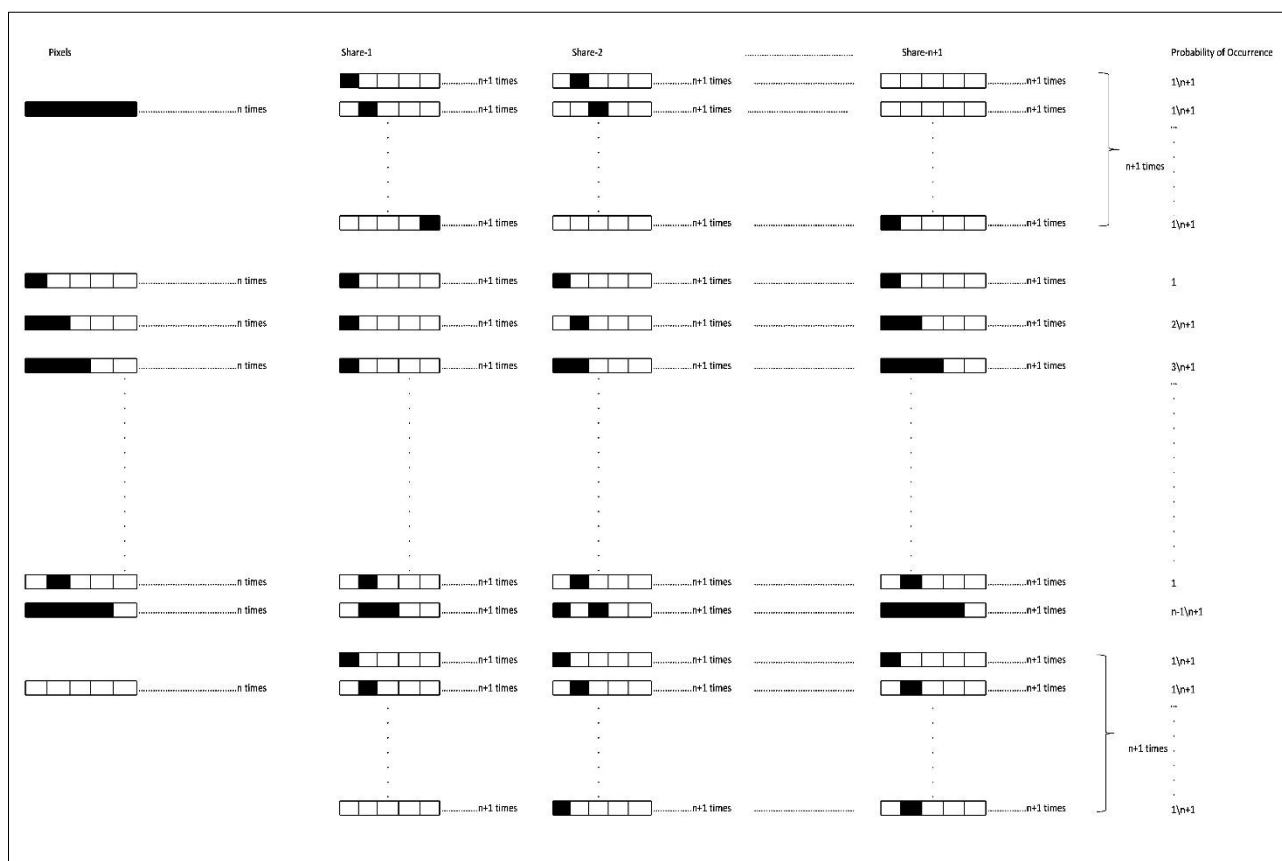


Fig 21: Share generation method for (n,n) visual cryptography technique

Perfect secrecy: The proposed scheme satisfies perfect secrecy in the information-theoretic sense. That is, the probability distribution of any single share remains the same regardless of the secret image being encoded. Consequently, no information about the secret can be inferred from an individual share. In the (n,n) visual cryptography scheme, any subset of shares containing fewer than n elements reveals no information about the secret image. Only when all n shares are combined can the secret image be reconstructed.

Performance analysis

Evaluating the performance of a cryptographic method is crucial for assessing its effectiveness and reliability. In this section, we present an analysis of the implemented approach. Comparative analysis involves comparing the scheme's performance with existing methods, identifying strengths, weaknesses, and optimization opportunities to enhance its efficiency and security while maintaining usability and robustness. Overall, a comprehensive performance analysis provides valuable insights into the scheme's strengths, weaknesses, and areas for improvement, guiding further research and development efforts in visual cryptography. The paper created by Chetana Hegde [5], Manu S, P Deepa Shenoy, Venugopal K R, and L M Patnaik was an effort for secure banking applications utilizing visual cryptography, according to the literature review. Let's examine the similarities and differences between the suggested approach and the algorithm created by Chetana Hegde *et al.*

Table 1: Performance Analysis

Characteristics	Algorithm proposed by Chetana Hegde <i>et al.</i>	Algorithm proposed by J. K. Pal <i>et al.</i>	Proposed Algorithm
Kind of algorithm	(2,2) visual cryptography	(2,n) visual cryptography	(n,n) visual cryptography
Practicality	Maximum number of account user is one	No limit in the number of account user	No limit of account user
Pixel extension	4 times	Variable	Variable
Pixel quantity per input	One	Two	n
Kind of algorithm	(2,2) visual cryptography	(2,n) visual cryptography	(n,n) visual cryptography
Secrecy	Moderate	Lower	Strongest

Table 1 shows a comparison of some key characteristics between the suggested algorithm and the algorithm by Chetana Hegde *et al.* [5]. It is evident that the suggested technique has a lower space overhead because two input pixels are combined and treated as a single pixel for share generation. Additionally, the suggested algorithm is more beneficial than the originally suggested two.

Conclusion

Transferring sensitive data over networks demands a high level of security, and securing shared data is a major challenge. Since the establishment of Visual Security in 1994, numerous forms of visual cryptography have been developed, including halftone, progressive multi-resolution, and segment-based schemes alongside other types of visual cryptographic schemes (2, 2). The

present field of study with the most potential is visual cryptography. Many nations are now using this specific encryption technology to transport data discreetly, including financial records, handwritten documents, text pictures, online votes, and more. The fundamental visual cryptography model that has been shown thus far has numerous potential improvements and additions. We are attempting one such improvement. This work presents a novel (n,n) visual cryptography technique that can be implemented with ease and offers security comparable to that of other existing algorithms. The suggested algorithm's ability to be used to any number of joint account holders is one of its key features. Compared to algorithms that accept a single pixel as the single input, this one may be more space-efficient even though it requires three successive pixels as the one-time input.

Also, Perfect secrecy is achieved by the (n,n) block-based Norm Visual Cryptography; single shares disclose zero information because of the uniform distribution of the Hamming weights throughout the n -pixel blocks. To ensure threshold security, each location needs $\geq m$ black pixels from all n shares for reconstruction. AES pre-encryption of secrets, restricted per-block randomness breaking pattern analysis, and SSIM (Structural Similarity Index Measure) verification (> 95 percent match), which detects manipulation, are examples of financial safeguards. Size-based assaults cannot withstand any pixel growth. Steganographic concealing inside meaningful images is made possible by halftone sharing. Partial shares (statistical independence), known-plaintext (AES), and side-channel (visual-only decryption) are all resistant to attacks. Perfect for multi-party authentication in e-banking.

References

1. Abraham AS, Nair LR, Deepa MS. A novel method for evaluation of visual security of images. In: 2017 International Conference on Networks & Advances in Computational Technologies (NetACT). IEEE; 2017. doi:10.1109/NetACT.2017.8076801.
2. Adhikari A, Roy B. On some constructions of monochrome visual cryptographic schemes. In: 2008 1st International Conference on Information Technology. IEEE; 2008. p. 1-6.
3. Gravano S. Introduction to Error Control Codes. Oxford (USA): Oxford University Press; 2001.
4. Houmansadr A, Ghaemmaghami S. A novel video watermarking method using visual cryptography. In: IEEE International Conference on Engineering of Intelligent Systems; 2006. p. 1-5.
5. Hegde C, Manu S, Shenoy PD, Venugopal KR, Patnaik LM. Secure authentication using image processing and visual cryptography for banking applications. In: 2008 16th International Conference on Advanced Computing and Communications. IEEE; 2008. p. 65-72.
6. Heidarinejad M, Yazdi AA, Plataniotis KN. Algebraic visual cryptography scheme for color images. In: IEEE International Conference on Acoustics, Speech and Signal Processing; 2008. p. 1761-1764.
7. Jena D, Jena SK. A novel visual cryptography scheme. In: 2009 International Conference on Advanced Computer Control. IEEE; 2009. p. 207-211.
8. Kumar R. An Intelligent Secret Image Sharing Scheme Using Visual Cryptography. Thanjavur; 2021.
9. Naor M, Shamir A. Visual cryptography. In: Advances in Cryptology—EUROCRYPT'94: Workshop on the Theory and Application of Cryptographic Techniques, Perugia, Italy, Proceedings 13. Springer; 1995. p. 1-12.
10. Pal JK, Mandal JK, Dasgupta K. A novel visual cryptographic technique through grey level inversion (VCTGLI). In: International Conference on Web and Semantic Technology. Springer; 2010. p. 124-133.
11. Pal JK, Mandal JK, Dasgupta K. A $(2, n)$ visual cryptographic technique for banking applications. International Journal of Network Security & Its Applications. 2010;2(4):118-127.
12. Park GD, Yoon EJ, Yoo KY. A new copyright protection scheme with visual cryptography. In: Second International Conference on Future Generation Communication and Networking Symposia; 2008. p. 60-63.
13. Shankar K, Eswaran P. Sharing a secret image with encapsulated shares in visual cryptography. Procedia Computer Science. 2015;70:462-468. doi:10.1016/j.procs.2015.10.080.
14. Shankar K, Eswaran P. RGB-based secure share creation in visual cryptography using optimal elliptic curve cryptography technique. Journal of Circuits, Systems and Computers. 2016;25(11):1650138. doi:10.1142/S0218126616501383.
15. Yue TW, Chiang S. A neural network approach for visual cryptography. In: Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN 2000): Neural Computing—New Challenges and Perspectives for the New Millennium. IEEE; 2000. Vol. 5. p. 494-499.
16. Yan WQ, Jin D, Kankanhalli MS. Visual cryptography for print and scan applications. In: IEEE International Symposium on Circuits and Systems (ISCAS). Vol. 5. IEEE; 2004. p. 1-4.
17. Zhou Z, Arce GR, Di Crescenzo G. Halftone visual cryptography. IEEE Transactions on Image Processing. 2006;15(8):2441-2453.