Multimedia Tools and Applications (2021) 80:14609–14635
https://doi.org/10.1007/s11042-020-10352-3

XOR-based visual secret sharing scheme using pixel vectorization

Suresh Prasad Kannojia[1] · **Jasvant Kumar**[2]

Received: 12 August 2019 / Revised: 19 October 2020 / Accepted: 22 December 2020 /
Published online: 27 January 2021

Abstract
The XOR operation has improved the recovery effect of the visual secret sharing schemes. Various visual secret sharing schemes based on XOR operation are available in the literature with some limitations, e.g. direct non-applicability of the scheme to the gray-scale images and reduced quality of the reconstructed secret image. Many existing visual secret sharing schemes in the literature, firstly converts grayscale secret image into a halftone image then creates the shares. This paper proposes XOR-based $(n, n) - VCS_{XOR}$ visual secret sharing scheme using pixel vectorization. It diffuses and disguise secret information into multiple meaningless shares prior to allocating participants of the group having no clue about the secret information. The scheme also resolves various problems like pixel expansion, contrast-loss, explicit codebook requirement, limitation on number of participants and lossy recovery of the secret image. The proposed scheme makes use of implicit codebook and pixel vectorization to encode gray secret image directly without converting to halftone image, into multiple random looking shares. Also applicable to binary image. Reveal a secret image perfectly by XOR-ing all share. The efficacy of the proposed scheme is verified by numerical illustrations, experimental results and comparative analysis. We found that proposed scheme outperform in comparison to the state-of-the-art approaches.

Keywords Pixel vectorization · XOR · XOR-based visual cryptography · Security · Secret sharing · Visual cryptography · Visual secret sharing schemes

1 Introduction

Cryptographic paradigm used for image encryption is visual cryptography [28], first introduced by Moni Naor and Adi Shamir in 1994. The traditional (k, n) visual secret sharing

✉ Jasvant Kumar
er.jaswantsingh786@gmail.com

Suresh Prasad Kannojia
spkannojia@gmail.com

[1] Department of Computer Science, University of Lucknow, Lucknow 226007, (U.P.), India

[2] ICT Research Laboratory, Department of Computer Science, University of Lucknow, Lucknow 226007, (U.P.), India

scheme encodes a binary secret image into n shares using two basis matrices generated explicitly. Further, original secret is revealed by stacking any k or more shares. Here, k out of n shares are printed on transparencies and stacked together to reveal the secret so it is called as (k, n)-threshold scheme of visual cryptography. The stacking operation is analogous to the OR operation that is why traditional visual cryptography is referred as OR-based visual cryptography. Informally, after creating the shares printed on transparencies and assigned to equal number of participants in such a way that each and every participant receives only a single share. Only secret will be revealed if any k or more than k shares are stacked together else individual or less than k participants are not able to reveal the secret, even if they are allowed to use infinite computational power as well as cryptographic knowledge.

The traditional visual cryptography was suffering from various issues such as meaningless shares, perfect reconstruction of black pixels, contrast of the revealed secret image, cheating by the participants, pixel expansion, explicit codebook requirement and direct non-applicability of the scheme to the gray-scale image. Based on the pioneer work of Moni Naor and Adi Shamir, many issues such as quality of shares [37, 43], perfect reconstruction of black pixels [1, 19], contrast of the reconstructed secret image [2, 15] and cheating prevention [9, 10, 16, 17, 20, 23, 31, 38] are extensively studied by researchers. To resolve the problem of pixel expansion two approaches, namely probabilistic and random-grids are available in the literature. Probabilistic approach [11, 25] have no pixel expansion, but recovered secret image suffers from contrast-loss due to which quality of revealed secret image is poor.

Random grids approach to encrypt binary secret image into two noise-like share images first introduced by Kafri and Keren [18]. Extended capabilities for random grids visual cryptography such as general access structures [3, 21, 44],multiple random grids [33], multiple secrets [5, 8, 32], (k, n) threshold schemes [6, 49], verifiable shares [4, 48], improving visual quality of revealed secret image [41, 46], user-friendly shares [7, 14, 24, 30] are investigated. Poor quality of revealed images is still persisted because contrast of the OR-based visual secret sharing schemes can achieve at most 1/2.

To further improve the visual quality of the revealed images some XOR-based schemes were developed. An XOR-based visual secret sharing schemes decode the secret image using the XOR operator among the shares. These schemes use, light weight, computational devices for decoding purpose instead of the human visual system. Such scheme reconstructs secret with better visual quality as well as resolves the problem of pixel alignment [27]. Recently, many research works [12, 13, 26, 29, 34–36, 39, 40, 42, 45, 47] based on XOR operation has been reported.

Tuyls et al. [39, 40] proposed a visual cryptography system that uses polarization of light in which operation of decryption is mathematically described by XOR operation. Wu et al. [42] proposed a visual secret sharing using random grids which generate meaningful shares for a binary secret image. Wu et al. [47] presented another work using random grids to share the binary secret with two decoding (OR and XOR) options. Wu and Son [45] generalized random grid based visual secret sharing for the binary secret image with an XOR operation based decoding. In 2014, Liu et al. [26] proposed an optimal $(2, n)$ scheme based on XOR operation in which various issues such as meaningless, explicit codebook requirement, pixel expansion still remain. In 2015, Ou et al. [29] proposes an XOR-based visual secret sharing scheme which generates non-expansible meaningful shares for a binary or halftone image. This scheme is not directly suitable for grayscale images because firstly it pre-process grayscale images by halftone techniques [51], due to which quality of secret image degrades prior to encryption. Deshmukh et al. [12] proposed a scheme to encrypt

multiple secrets by using an XOR operation and arithmetic modulo to create shares. Singh [34] performed comparative analysis between XOR of messages with the LSB and XOR of the message with a secret key, concluded that the XOR of LSB is more efficient than another. Singh et al. [35] proposed an XOR-based visual secret sharing scheme for binary secret image with unique meaningful shares. Fu et al. [13] proposed visual secret sharing scheme to realize the perfect recovery of the secret image. It is based on random generation of intermediate images which are further used to encrypt a secret image. Tan et al. [36] proposes a visual secret sharing scheme based on QR codes which generates robust and meaningful shadows.

Literature analysis shows that most of the above mentioned scheme deals with binary or halftone image and the quality of the revealed secrets is still an issue.

In this paper, we focus on how to encrypt a gray secret image directly without converting into halftone image. Proposed XOR-based visual secret sharing scheme using pixel vectorization, diffuses and disguise secret image into a number of random looking shares distributed to participants in such a way that each participant receives only one share and have no clue about the secret. Secret image can be revealed by XOR-ing all share received from participants. Also applicable to binary image, reconstructs the original secret image without clarity-loss when all shares are available without tamper.Three novel algorithm construction matrix generation (Algorithm 1), division of a construction matrix into basis matrices (Algorithm 2) and $(n, n) - VCS_{XOR}$ visual secret sharing scheme (Algorithm 3) have been proposed to achieve the intended objective.

The highlighted contributions of the proposed scheme are as follows:

- **Pixel vectorization:** The idea of pixel vectorization is introduced, used with implicit codebook to encrypt a secret image.
- **Applicability:** Unlike other visual secret sharing schemes which converts grayscale image into a halftone image prior to feeding as input to the scheme, the proposed scheme directly applicable to the grayscale secret image.
- **No limit on number of participants:** The proposed scheme is not restricted to a specific number of participants. It is suitable for any n, $n \geq 2$ number of participants to resolve any related real world problem.
- **No pixel expansion:** The proposed scheme diffuses and disguise secret image into multiple number of random looking images of the same size, hence resolves the pixel expansion consequently minimum storage and transmission time required.
- **Implicit codebook generation:** There is no requirement of explicit codebook generation because the proposed scheme generates codebook at run time depending upon the number of participants, hence no need to store codebook.
- **Reveals secret image perfectly:** The proposed scheme reconstructs each and every pixel of secret image from the respective pixels of the shares without clarity-loss if they are not tempered. Hence, the contrast of the reconstructed secret image is 100% which fulfills the intended objective of secret sharing.
- **Reveals quality secret image:** The quality of an image depends on the number of bits associated with each and every pixel of the image. Since binary and halftone images are made of single bit per pixels, hence not sufficient to represent the high quality phenomenon. While grayscale images associates 8 bits per pixel which are sufficient to represent any high quality phenomenon. The proposed scheme takes binary as well as 8 bit grayscale images as secret image. And in revealing process, reconstructs each and every bit (0 or 1) associated with all the pixels of the secret image perfectly. Hence, revealed secret image is same as input secret image.

The rest of the paper is organized as follows: Section 2 describes the background and preliminaries. Proposed work is explained in Section 3. Experimental results and analysis are presented in Section 4. Comparison with the state-of-the-art approaches is given in Section 5 followed by conclusion in Section 6. Various notations used and their descriptions are given in Table 1.

2 Background and preliminaries

The brief overview of the concepts such as fundamental concept of visual cryptography, bit representation of pixel values, its vectorization and preliminary definitions which have been used in the proposed approach are discussed in Sections 2.1, 2.2, 2.3 and 2.4 respectively.

2.1 Fundamental concept of visual cryptography

Visual Cryptography deals with methods and techniques to securely share a secret image between the participants of a group. Let S be a secret image and $\mathcal{P} = \{P_1, P_2, ..., P_n\}$ be a group of n participants. Objective of visual cryptography is to encode the secret image S into n random images called shares in such a way that no clue about original secret image can be drawn from individual shares. Each participant of the group $\mathcal{P}$ receives only one share. Whenever any k, $2 \leq k \geq n$ shares are processed in a particular way, information about the secret image can be revealed [28].

2.2 Bit representation of pixel value

Let S be a $N_r \times N_c$ digital image and its pixel values are represented by K bits, K=1 for binary image and K=8 for gray image. Let $S(i, j)$, $1 \leq i \leq N_r$, $1 \leq j \leq N_c$, be a pixel of the digital image S, represented by $S(i, j) = [V_k....V_2V_1]$, $k = 1$ to 8. To compute the integer pixel value of the pixel $S(i, j)$, (1) is used.

$$S(i, j) = \sum_{k=1}^{8} (V_k \times 2^{k-1}) \tag{1}$$

2.3 Pixel vectorization

Various existing visual secret sharing schemes which deals with grayscale secret image firstly converts them into halftone images, then consider the resultant image as a secret image. The halftone image is a single bit per pixel image, but creates illusion of being grayscale. Thus, the quality of the sharing image is reduced in comparison to the original secret image. To overcome the problem of the reduction in quality of secret image which occurs due to converting grayscale secret image into a halftone image we use the concept of pixel vectorization. Pixel vectorization is a K bit binary representation of a pixel of an image, where K is the maximum number of bits that can represent all the gray-scale levels. For example, if S be an 8 bit grayscale image, it means the integer pixel value of each pixel of image S is in the range from 0 to 255. Now suppose we have to vectorize the pixel with integer pixel value 30. Here, 8 bit vector representation of the integer pixel value 30 is $V = [0\ 0\ 0\ 1\ 1\ 1\ 1\ 0]$. In case of binary secret image pixel vectorization of 0 or 1 will be represented as $V = [0]$ or $V = [1]$ respectively. Because only one bit per pixel is required to represent a binary image.

Table 1 Notations and their descriptions

Notation		Description
S	:	Secret image
N_r	:	Number of rows in S
N_c	:	Number of Columns in S
n	:	Number of participants
C_n	:	Construction matrix for n participants
$\mathcal{P}$	:	Set of participants
P_i	:	$P_i \in \mathcal{P}, 1 \leq i \leq n$
C_n^{even}	:	Matrix for row vectors of C_n w.r.t even hamming weight
C_n^{odd}	:	Matrix for row vectors of C_n w.r.t odd hamming weight
d	:	Integer pixel value of $(i, j)^{th}$ pixel
$S_1, S_2, ..., S_n$	:	Shares
V	:	Vector
V_k	:	k^{th} bit of vector V,$k = 1$ to 8
$A^1, A^2, ..., A^n$	:	Vectors
A_k^i	:	k^{th} bit of $A^{i^{th}}$ Vector
t	:	Random number
t^o	:	Random number when $V_k = 1, 1 \leq k \leq 8$
t^e	:	Random number when $V_k = 0, 1 \leq k \leq 8$
$S(i, j)$	:	$(i, j)^{th}$ pixel of S
$S_1(i, j), ..., S_n(i, j)$	:	Pixel shares of the secret pixel$S(i, j)$
$X(0)$	:	Area covered by white pixels in image X
$X(1)$	:	Area covered by black pixels in image X
$Y(0)$	:	Area covered by white pixels in image Y
$Y(1)$	:	Area covered by black pixels in image Y
$w_h(.)$	:	Hamming function
w	:	Hamming weight of a particular row in C_n
R^d	:	Reconstructed value for d
$\oplus$	:	Exclusive OR operation
α_{xor}	:	Contrast of the reconstructed secret image

2.4 Preliminary definitions

Some preliminary definitions are incorporated to analyze the performance of the proposed scheme.

Definition 1 (Average light transmission) [29] For a certain pixel $S(i, j), 1 \leq i \leq N_r, 1 \leq j \leq N_c$ in the binary secret image S which is $N_r \times N_c$ in size, the probability of pixel $S(i, j)$ being is white, called $prob(S(i, j) = 1)$, represents the light transmission of pixel $S(i, j)$, which is denoted as $T(S(i, j)) = 1$ and $T(S(i, j)) = 0$, for black pixel. Average light transmission of S is given by

$$T(S) = \frac{\sum_{i=1}^{N_r} \sum_{j=1}^{N_c} S(i, j)}{N_r \times N_c} \tag{2}$$

Definition 2 (Area representation) [29] Let $X(1)$ (resp.$X(0)$) be the area of all the white (resp. black) pixels in image X, where $X = X(1) \cup X(0)$ and $X = X(1) \cap X(0) = \phi$. Therefore $Y(1)$ (resp.$Y(0)$) is the corresponding area of all the white (resp. black) pixels in image Y.

Definition 3 (Contrast of the revealed secret image) [29] The contrast of the revealed secret image $S_{\{\oplus,1,....,n\}} = S_1 \oplus .... \oplus S_n$ with respect to the original secret image S is

$$\alpha_{xor} = \frac{T(S_{\{\oplus,1,....,n\}}[S(1)]) - T(S_{\{\oplus,1,....,n\}}[S(0)])}{1 + T(S_{\{\oplus,1,....,n\}}[S(0)])} \tag{3}$$

where $\oplus$ is a Boolean exclusive-OR operation, and S_1, ..., S_n are shares of the secret image S.

3 Proposed work

Proposed scheme provides an elegant method to diffuse and disguise a binary or grayscale secret image into random looking shares. It eradicates the major limitations of the state-of-the-art approaches such as direct non-applicability of the scheme to securely share grayscale images, poor contrast, pixel expansion and explicit codebook, demarcation on the number of participants. The most of the existing visual secret sharing schemes used for encrypting binary or halftone images which are not sufficient to convey more information. The OR based visual secret sharing schemes suffers with the contrast - loss. The expanded shares pass overhead of storing and requires more storage. The restriction on the number of participants, limits the applicability and scope of the scheme. The explicit codebook used for encrypting posses overhead of storing it. The proposed scheme resolves all these problems. The proposed scheme encrypts grayscale image using implicitly generated codebook, reveals secret image perfectly (no contrast-loss), generates non-expansible shares and there is no demarcation on the number of participants. Although proposed approach generates overhead of converting pixels of the sharing image into integer pixel values before the encryption. The framework of the proposed model is depicted in Fig. 1. The methodology of the proposed approach involves: implicit construction matrix generation (Algorithm 1), division of a construction matrix into basis matrices (Algorithm 2) and proposed $(n, n) - VCS_{XOR}$ visual secret sharing scheme (Algorithm 3) in the Sections 3.1, 3.2, 3.3 respectively. Further, revealing process (Algorithm 4) and verification and analysis is traced in the Sections 3.4 and 3.5 respectively.

3.1 Construction matrix generation

The proposed scheme is flexible enough for any number of participants. So as per the need of the problem, the number of participants can be decided. Let secret image S be to be shared between n participants. To share the secret image S between n participants a matrix of order $2^n \times n$ is required, which will be used to construct basis matrices for the scheme. Steps for construction matrix generation are given in Algorithm 1.

In *Algorithm* 1 function $de2bi(i-1, n)$ is used to convert decimal number $(i-1)$ into a row vector of n bits. This row vector is n bit binary equivalent of decimal number $(i-1)$ in reverse order. For example $de2bi(5, 8) = [1\ 0\ 1\ 0\ 0\ 0\ 0\ 0]$. After executing *Algorithm* 1, obtained matrix C_n contains the binary codes of decimal number from 0 to $2^n - 1$. Next step is to divide matrix C_n into two $2^{n-1} \times n$ sub-matrices C_n^{even} and C_n^{odd}, where matrix C_n^{even} includes those row vectors of C_n whose hamming weight $w_h(.)$ is an even number while C_n^{odd} includes row vectors whose hamming weight is an odd number.

Algorithm 1 Construction matrix generation.

Input: Parameter n, $n \geq 2$ the number of participants .
Output: A $2^n \times n$ matrix C_n

function MATRIX(n) ▷ Where, n-Number of participants
  for $i = 1$ to 2^n **do**
    $C(i, 1 \text{ to } n) = de2bi(i-1, n)$ ▷ Convert decimal number $i-1$ to n bit binary number
  end for
  for $j = 1$ to n **do**
    $C_n(1 \text{ to } 2^n, j) = C(1 \text{ to } 2^n, n-j+1)$ ▷ Reverse all the row vectors of matrix C.
  end for
  return C_n
end function

3.2 Division of construction matrix

This section provides pseudo-code to divide construction matrix C_n into two sub matrices C_n^{even} and C_n^{odd}. For illustration, matrix C_3 for three participants, generated by Algorithm

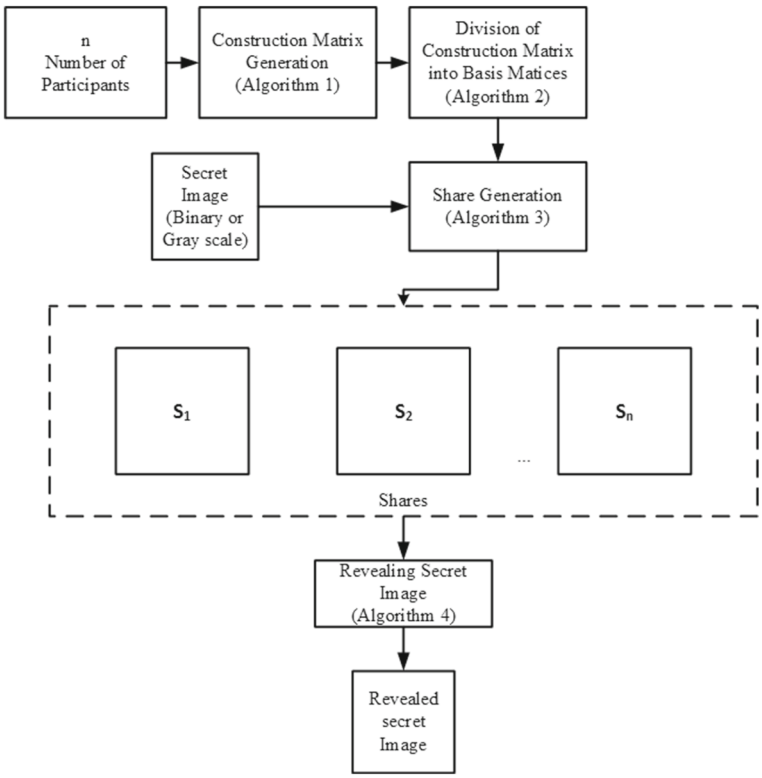

Fig. 1 Framework of the proposed approach

1 and its sub-matrices C_3^{even} and C_3^{odd} generated by Algorithm 2 are given below. $C_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}_{8\times 3}$, $C_3^{even} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}_{4\times 3}$ and $C_3^{odd} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}_{4\times 3}$

Algorithm 2 Algorithm to divide C_n into C_n^{even} and C_n^{odd}.

Input: A $2^n \times n$ matrix C_n generated by Algorithm 1.
Output: The matrices C_n^{even} and C_n^{odd}.

function DIV(C_n, n) ▷Where n - Number of participants
  $counter_1 \leftarrow 1$
  $counter_2 \leftarrow 1$
  for $i = 1$ to 2^n **do**
    for $j = 1$ to n **do**
      $w \leftarrow w_h(C_n(i, 1\ to\ j))$ ▷Hamming weight of vector i having j columns
      if w is an even number **then**
        $C_n^{even}(counter_1, 1\ to\ j) = C_n(i, 1\ to\ j)$
        $counter_1 = counter_1 + 1$
      else
        $C_n^{odd}(counter_2, 1\ to\ j) = C_n(i, 1\ to\ j)$
        $counter_2 = counter_2 + 1$
      end if
    end for
  end for
  return C_n^{even}, C_n^{odd}
end function

3.3 Proposed: (n, n) − VCS_{XOR} visual secret sharing scheme

The flow diagram of the proposed secret sharing scheme is depicted in Fig. 2. Firstly, a pixel of the secret image is picked out, converted to integer value called as an integer pixel value of that pixel. Later on pixel vectorization is carried out. Now, pick out a single bit from the vector representation of the pixel to encrypt it with the help of basis matrices. The choice of basis matrices depends on the color 0 (white) and 1 (black) of the bits. After that, on the basis of a random number a row the selected basis matrix is chosen. Side by side, assign value under the first column of that row to first vector A^1, value under the second column of that row to second vector A^2 and so on. Likewise, all the bits are to be encrypted of that pixel. After that values of the vectors are calculated using bit representation (1). Next step is to assign these vector values to corresponding positions of the shares are generated. This procedure is repeated for all the pixels of the secret image to obtain final shares. The intuitive idea of the proposed secret sharing scheme is given in *Algorithm* 3. The procedure for sharing secret pixel $S(i, j) = d$, where d is an integer pixel value of secret pixel $S(i, j)$ of the secret image S, illustrated here. First of all converts d into K bit binary numbers, where K is the number of bits required to represent all the grayscale of the secret image.

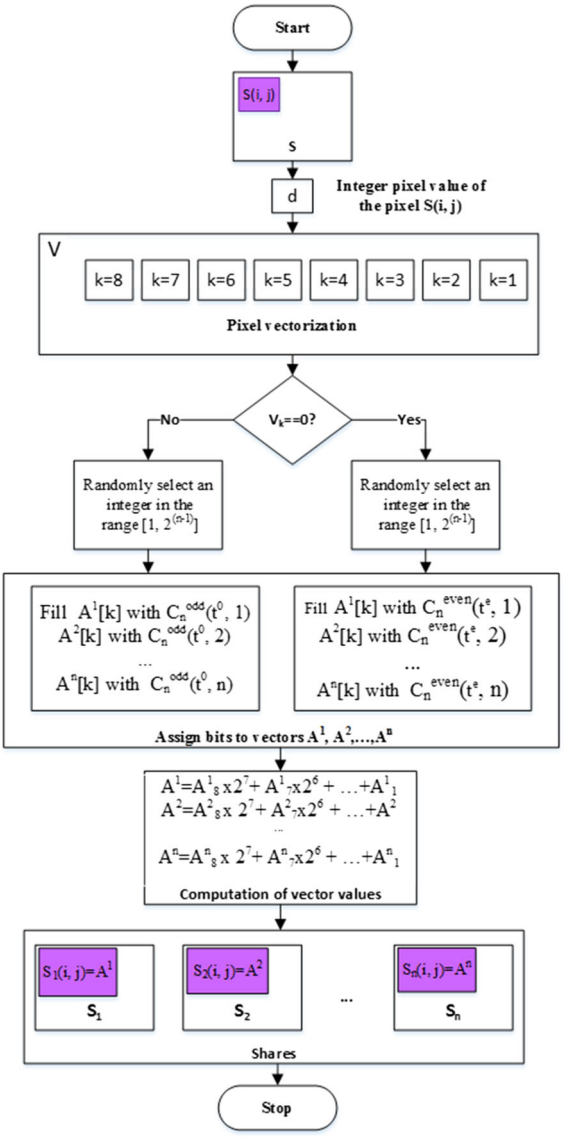

Fig. 2 Flowchart of proposed $(n, n) - VCS_{XOR}$ visual secret sharing scheme using pixel vectorization

Value of K is 1 for binary secret image and 8 for grayscale secret image. Here the function $bitget()$ is used to convert d into k bit vector V. The function $bitget(d, k)$ provided by MATLAB to extract k^{th} bit of the binary representation of decimal number d from LSB. For example $bitget(204, 1) = 0$, where, 0 is the first bit from the LSB side. Now consider k^{th} bit V_k of vector V, let us assume $V_k = 0$, then choose C_n^{even} to construct vector bits $A_k^1, A_k^2, ..., A_k^n$. Further to select the row vector of the C_n^{even} matrix, function randi(2^n) is used, where $randi(2^n)$ is MATLAB function which generates random integer numbers with equal probability from 1 to 2^n. Let $randi(2^{n-1}) = t^e$, where, $n - 1$ is the number of row vectors in the C_n^{even} matrix, subsequently compute $A_k^1, A_k^2, ..., A_k^n$ as, $A_k^1 = C_n^{even}(t^e, 1)$, $A_k^2 = C_n^{even}(t^e, 2)$, ..., and $A_k^n = C_n^{even}(t^e, n)$.

On the other hand, when $V_k = 1$, then according to Algorithm 3, choose C_n^{odd} matrix to construct vector bits $A_k^1, A_k^2, ..., A_k^n$, after that to choose row vector from C_n^{odd} matrix, generate a random number t^o, such that $randi(2^{n-1}) = t^o$, then compute $A_k^1, A_k^2, ..., A_k^n$ as, $A_k^1 = C_n^{odd}(t^o, 1)$, $A_k^2 = C_n^{odd}(t^o, 2)$, ..., and $A_k^n = C_n^{odd}(t^o, n)$. After processing all the bits of the vector V, compute integer values of the vectors A^1, A^2,...., A^n by using (1) as, $A^1 = \Sigma_{k=1}^{8}(2^{k-1} \times A_k^1)$, $A^2 = \Sigma_{k=1}^{8}(2^{k-1} \times A_k^2)$,..., and $A^n = \Sigma_{k=1}^{8}(2^{k-1} \times A_k^n)$. Next is to assign values of A^1, A^2,...., and A^n to respective location of the shares S_1, S_2,..., and S_n as, $S_1(i, j) = A^1$, $S_2(i, j) = A^2$,..., and $S_n(i, j) = A^n$.

Algorithm 3 $(n, n) - VCS_{XOR}$ visual secret sharing scheme.

ASSUME:
1. Gray scale image is composed of eight bits only.
2. logical values 1 and 0 converts to decimal 1 and 0 respectively

INPUT: Secret image S with N_r rows and N_c columns, matrices C_n^{even} and C_n^{odd}
OUTPUT: Shares $S_1, S_2, ..., S_n$

function $(n, n) - VCS_{XOR}S, N_r, N_c$ ▷Where S - Secret Image, N_r - rows, N_c - columns
  for $i = 1$ to N_r **do**
    for $j = 1$ to N_c **do**
      $d = S(i, j)$
      for $k = 1$ to 8 **do** ▷ Every pixel is represented by 8 bit binary number
        $V_k = bitget(d, k)$ ▷ k gradually increases from LSB to MSB
      end for
      $A^1 \leftarrow 0$ ▷$A^1, A^2, ..., A^n$ are vectors
      $A^2 \leftarrow 0$
      ....
      $A^n \leftarrow 0$
      for $k = 1$ to 8 **do**
        if $V_k == 0$ **then**
          $t^e = rand(1, 2^{n-1})$ ▷ t-random number from 1 to 2^{n-1}
          $A_k^1 = C_n^{even}(t^e, 1)$, $A_k^2 = C_n^{even}(t^e, 2)$,..., $A_k^n = C_n^{even}(t^e, n)$
        else
          $A_k^1 = C_n^{odd}(t^o, 1)$, $A_k^2 = C_n^{odd}(t^o, 2)$,..., $A_k^n = C_n^{odd}(t^o, n)$
        end if
      end for
      for $k = 1$ to 8 **do**
        $A^1 = A^1 + \sum_{k=1}^{8}(A_k^1 \times 2^{k-1})$
        $A^2 = A^2 + \sum_{k=1}^{8}(A_k^2 \times 2^{k-1})$
        ...,
        $A^n = A^n + \sum_{k=1}^{8}(A_k^n \times 2^{k-1})$
      end for
      $S_1(i, j) = A^1$
      $S_2(i, j) = A^2$
      ....,
      $S_n(i, j) = A^n$
    end for
  end for
  return $S_1, S_2, ..., S_n$
end function

The same process is repeated for every pixel of the secret image S to get the final shares S_1, S_2,..., and S_n.

3.4 Revealing secret image

To reconstruct the secret image responsible authorized entity collects all the shares from the participants and ensures that they are not tampered. After that Algorithm 4 can be used, which performs XOR operation between all the shares and reveals the secret image.

Algorithm 4 Revealing algorithm.

Input: Shares $S_1, ..., S_n$
Output: Secret image S
1: $temp = S_1$
2: **for** $i = 2$ to n **do**
3: $temp = temp \oplus S_i$ $\triangleright \oplus \rightarrow$ exclusive-OR operation.
4: **end for**

3.5 Verification and analysis of the proposed approach

To ensure efficacy of a newly proposed visual secret sharing scheme, it must satisfy the basic conditions which are security and contrast condition. To ensure effectiveness of the proposed scheme, theoretical analysis of Algorithm 3 is given bellow. Theorems 1 and 2 demonstrates that Algorithm 3 is a valid construction for the proposed scheme.

Lemma 1 *Given n shares $S_1, ..., S_n$, generated from Algorithm 3, each of which is a random-noise like image and gives no clue about the secret image S: $T(S_k[S(0)]) = T(S_k[S(1)]) = 1/2$, where k=1,....,n.*

Proof Every column vector in $C_n^{even}(C_n^{odd})$ generated by Algorithm 2 always has hamming weight equal to $2^{n-2}(2^{n-2})$, it means number of 1 and 0 are same in every column vector. It implies matrices $C_n^{even}(C_n^{odd})$ have equal number of 1 and 0 with probability of being 1 or 0 is 1/2 for each element in the matrix. According to Algorithm 3, secret pixel $S(i, j)$ assigned to do, firstly converted to a K bit binary number(pixel vectorization). To construct n vector bits $A_k^1, ..., A_k^n$ for every bit V_k of the secret pixel $S(i, j)$, a row vector would be arbitrarily selected from $C_n^{even}(C_n^{odd})$ with equal probability $1/2^{n-1}$. Thus a bit 0 or 1 are assigned to $A_k^1, ..., A_k^n$ with probability $1/2$ and no matter that, the bit V_k is 1 or 0, hence we have prob$(A_k^n = 1) = 1/2$ and prob$(A_k^n = 0) = 1/2$. By Definitions 1 and 2 , we have $T(A_k^i[V(0)]) = T(A_k^i[V(1)]) = 1/2, (i = 1, ..., n)$. After that, using vector bits $A_k^n, k = 1, ..., 8$, values $A^1, ..., A^n$ are computed by using (1), assigned to share pixels $S_1(i, j), ..., S_n(i, j)$, implies that $T(S_k[S(0)]) = T(S_k[S(1)]) = 1/2$. Therefore have no clue about the secret pixel, except random noise like image. □

Lemma 2 *Given n shares $S_1, ..., S_n$, each of which is random noise like image, generated using Algorithm 3, the XOR-Ed result of any p $(p < n)$ shares $S_{\{\oplus, x_1, ..., x_p\}} = S_{x_1} \oplus .... \oplus S_{x_p}$ gives no clue about the secret image S: $T(S_{\{\oplus, x_1, ..., x_p\}}[S(1)]) = T(S_{\{\oplus, x_1, ..., x_p\}}[S(0)]) = 1/2$.*

Proof For a K bit secret image $S(i, j)$(K=1 for binary and K=8 for gray-scale image). Let n be the number of participants. According to Algorithm 3, $S(i, j)$ assigned to variable d, which is further converted to vector V such that $V = [V_8V_7V_6V_5V_4V_3V_2V_1]$. Every bit V_k, where $k = 1, 2..., 8$, encoded into vector bits $A_k^1, ..., A_k^n$. Let p vector bits from $A_k^1, ..., A_k^n$ are denoted by $A_k^{j_1}, ..., A_k^{j_p}$, where $\{j_1, ..., j_p\} \subsetneq \{1, ..., n\}$ and row vector V_r with p elements $A_k^{j_1}, ..., A_k^{j_p}$ given as $V_r = [A_k^{j_1}, A_k^{j_2}..., A_k^{j_p}]$, $j = 1, 2, ..., 8$.

When $V_k = 0$, a row vector $[x_1, ..., x_p]$ would be randomly chosen from matrix $C_n^{even}(1 : 2^{n-1}, [x_1, ..., x_p])$ with equal probability $1/2^{n-1}$, and assigned to V_r. We know that, number of row vectors with even hamming weight is equal to the row vector with odd hamming weight in the matrix $C_n^{even}(1 : 2^{n-1}, [x_1, ..., x_p])$. It is known that, XOR-Ed result of the row vector with even hamming weight is 0 while it is 1 for row vectors with odd hamming weight. Hence, row vector from matrix $C_n^{even}(1 : 2^{n-1}[x_1, ..., x_p])$ with p elements will produce a random number 0 or 1 with probability 1/2. Thus, the probability of row vector V_r leading to white pixel is 1/2, given as $\text{prob}(A^{\{\oplus, x_1, ..., x_p\}}[V_k = 0]) = 1/2$.

On the other hand, when $V_k = 1$, a row vector would be randomly chosen from matrix $C_n^{odd}(1 : n, [x_1, ..., x_p])$ with equal probability $1/2^{n-1}$, and assigned to V_r. Similarly, in the matrix $C_n^{odd}(1 : 2^{n-1}, [x_1, ..., x_p])$ we have an equal number of row vectors with even and odd hamming weight. As a consequence, probability of row vector V_r leading to black pixel is 1/2, such that $\text{prob}(A^{\{\oplus, x_1, ..., x_p\}}[V_k = 1]) = 1/2$. According to *Definitions* 1 and 2, we have $T(A^{\{\oplus, x_1, ..., x_p\}}[V_k = 0]) = (A^{\{\oplus, x_1, ..., x_p\}}[V_k = 1]) = 1/2$.

Now, according to Algorithm 3, $A^1, ..., A^p$ are computed using (1), assigned to $S_{x_1}(i, j), ..., S_{x_p}(i, j)$ such that $S_{x_1}(i, j) \leftarrow A^1, ..., S_{x_p}(i, j) \leftarrow A^p$. Here, probability of vector bits $A_k^1, ..., A_k^p$ is 1/2 therefore values $A^1, ..., A^p$ are random. Therefore, $T(S_{\{\oplus, x_1, ..., x_p\}}[S(1)]) = T(S_{\{\oplus, x_1, ..., x_p\}}[S(0)]) = 1/2$. Hence, an XOR-Ed result by any p, $(p < n)$ shares $S_{\{\oplus, x_1, ..., x_p\}} = S_{x_1} \oplus .... \oplus S_{x_p}$ have no clue about the secret image. □

Lemma 3 *Given n shares $S_1, ..., S_n$, each of which is random noise like image, generated using Algorithm 3, the XOR-Ed result by n shares $S_{\{\oplus, 1, ..., n\}} = S_1 \oplus ... \oplus S_n$ can visually reconstruct the secret image S: $T(S_{\{\oplus, 1, ..., n\}}[S(1)]) > T(S_{\{\oplus, 1, ..., n\}}[S(0)])$.*

Proof For a K bit secret image $S(i, j)$(K=1 for binary and K=8 for grayscale image). Let n be the number of participants. According to Algorithm 3, $S(i, j)$ assigned to variable d, which is converted to vector V such that $V = [V_8V_7V_6V_5V_4V_3V_2V_1]$. Let V_r be a row vector with n elements $A_k^1, ..., A_k^n$ such that $V_r = [A_k^1, ..., A_k^n]$, where $A^n = \sum_{k=1}^{8}(2^{k-1} \times A_k^n)$. When $V_k = 0$, a row vector would be chosen from matrix C_n^{even} with equal probability $1/2^{n-1}$, and the chosen vector is then assigned to V_r, its hamming weight is always an even number because any row vector in the matrix C_n^{even} always have an even number of 1. Thus XOR-Ed result by all bits $A_k^1, ..., A_k^n$ is always 0. As a result, we obtain $T(A^{\{\oplus, 1..., n\}}V[0]) = 0$.

On the other hand, when $V_k = 1$, a row vector would be randomly chosen from matrix C_n^{odd} with equal probability $1/2^{n-1}$ and assigned to V_r, no matter which row vector is selected from the matrix C_n^{odd}, its hamming weight is always an odd number because any row vector in the matrix C_n^{odd} always has odd number of 1. Thus XOR-Ed results of all the elements are always 1. Hence we obtain $T(A^{\{\oplus, 1..., n\}}V[1]) = 1$.

Since, $S_1(i, j) \leftarrow A^1, ..., S_n(i, j) \leftarrow A^n$. Therefore $T(A^{\{\oplus, 1..., n\}}[V_k = 0]) = 0$ and $T(A^{\{\oplus, 1..., n\}}[V_k = 1]) = 1$ implies that $T(S_{\{\oplus, 1, ..., n\}}|[S(0)]) = 0$ as well as $T(S_{\{\oplus, 1, ..., n\}}|[S(1)]) = 1$. It is clearly obvious that, $T(S_{\{\oplus, 1, ..., n\}}[S(1)]) - T(S_{\{\oplus, 1, ..., n\}}|[S(0)]) = 1 -$

$0 = 1$. Hence, $T(S_{\{\oplus,1,\ldots,n\}}[S(1)]) > T(S_{\{\oplus,1,\ldots,n\}}[S(0)]))$. An XOR-Ed results visually reconstructs the secret image. □

Theorem 1 *Let $S_1, \ldots, S_n$ be the n shares generated using Algorithm 3. Algorithm 3 is said to be a valid construction for XOR-based n out n scheme $(n, n) - VCS_{XOR}$, if the following conditions are satisfied.*

- *Every share is a random-noise like share and have no clue about the secret image S : $T(S_k[S(0)]) = T(S_k[S(1)]) = 1/2$, where $k = 1, \ldots, n$.*
- *The XOR-Ed result by any p, $(p < n)$ shares $S_{\{\oplus,x_1,\ldots,S_{x_p}\}} = S_{x_1} \oplus \ldots \oplus S_{x_p}$ is a random noise like image and have no clue about the secret image S: $T(S_{\{\oplus,x_1,\ldots,x_p\}}[S(1)]) = T(S_{\{\oplus,x_1,\ldots,x_p\}}[S(0)]) = 1/2$.*
- *XOR-Ed result by n shares visually reconstruct secret image S: $T(S_{\{\oplus,1,\ldots,n\}}[S(1)]) > T(S_{\{\oplus,1,\ldots,n\}}[S(0)]))$.*

Theorem 2 *Given n shares $S_1, \ldots, S_n$, generated by Algorithm 3, will reveal a secret image loss-less if the contrast of revealed secret image is 1, given by $\alpha_{xor} = 1$.*

Proof From the proof of lemma 3, we have $T(S_{\{\oplus,1,\ldots,n\}}|[S(1)]) = 1$, and $T(S_{\{\oplus,1,\ldots,n\}}|[S(0)]) = 0$. By Definition 3, the contrast of XOR-Ed result can be obtained by

$$\alpha_{xor} = \frac{T(S_{\{\oplus,1,\ldots,n\}}[S(1)]) - T(S_{\{\oplus,1,\ldots,n\}}[S(0)])}{1 + T(S_{\{\oplus,1,\ldots,n\}}[S(0)])} = \frac{1-0}{1+0} = 1$$

□

4 Experimental results and analysis

Feasibility and histogram analysis of the experimental results are described in Sections 4.1 and 4.2 respectively. Performance analysis is described in Section 4.3. Validity analysis through numerical illustrations is described in Section 4.4. Security against attacks is described in Section 4.5 followed computational complexity in Section 4.6.

4.1 Feasibility

Feasibility of the proposed scheme is evaluated by experiments on binary and grayscale images. Herein Figs. 3 and 5 binary secret images and in Figs. 4 and 6 grayscale secret image is used. Experimental results of the case (2, 2) by Algorithm 3 unfold in Fig. 3. Image Fig. 3a is a secret image 128 × 128 in size, images Fig. 3b and c are shares and (d) is reconstructed image. Experimental results of the case (2, 2) for the standard grayscale image (Leena image) is unfolding in Fig. 4, here image (a) is a secret image 128 × 128 in size, images (b) and (c) are shares (d) is reconstructed image. Figure 5, show the experimental results of the case (3, 3) by Algorithm 3, images Fig. 5a and e are secret and reconstructed images respectively, while (b), (c) and (d) are shares. Experimental results of the case (3, 3) for gray-scale secret images are depicted in Fig. 6. Image Fig. 6a is the input image, images Fig. 6b–d are shares and (e) is reconstructed secret image. All the share images have the same size as the size of secret image. Experimental results show that proposed scheme is feasible for both the binary and grayscale secret images.

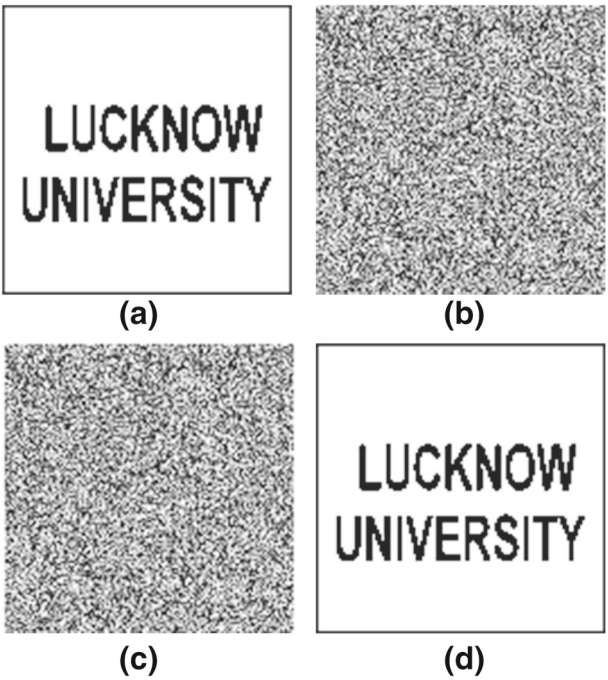

Fig. 3 Experimental results of the (2, 2) case for binary image by *Algorithm* 3. **a** Secret image, **b** Share S_1, **c** Share S_2, **d** Revealed secret image ($S_1 \oplus S_2$)

4.2 Histogram analysis

Histograms are a type of bar plot for numeric data that group the data into bins. The number of bins in the histogram depends on the image type. In histogram for binary there is only 2 bins and gray image uses 256 bins. Figure 7 is showing histograms of secret images and revealed secret images of the Figs. 3, 4, 5 and 6. Here, it is obvious that histograms of the secret image and revealed secret image of concerned figures are similar, concludes that secret image and revealed secret images of concerning figures are identical.

4.3 Performance analysis

To evaluate the performance of the proposed scheme, it is required to carry out a quantitative analysis of the experimental results. In the simulation of the proposed scheme four experiments are carried out, two for the binary secret image and the other two for gray secret image. Therefore, to carry out the performance analysis of the experimental results different quality metrics are required. In the case of binary image the quality metrics such as precision, recall, F-measure, balanced classification rate, balance error rate and negative rate matrix are used. And, to measure the quality of revealed secret image in the case of gray image quality metrics such as maximum absolute error (MAE), mean square error (MSE) and peak signal-to-noise ratio (PSNR) is used. A brief description of these parameters is given below:

Objective evaluation parameters or quality metrics are quantitative error measurement parameters between the respective pixels of the images. Let us suppose that S and S' are

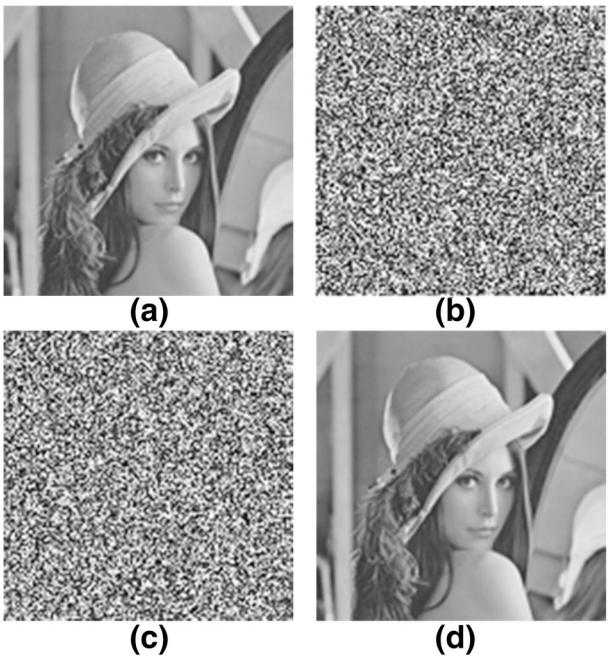

Fig. 4 Experimental results of the (2, 2) case for grayscale image by *Algorithm* 3. **a** Secret image, **b** Share S_1, **c** Share S_2, **d** Revealed secret image ($S_1 \oplus S_2$)

binary secret image and revealed (output) binary image respectively. There are some test results N_{TP}, N_{FP}, N_{TN} and N_{TP} which are represented as true positive, false positive, true negative and false negative respectively. Here, N_{TP}, N_{FP}, N_{TN} and N_{TP} are defined as follows:

True Positive (N_{TP}): When pixel value at corresponding location in both the S and S' is same and equal to 1 then this case is called true positive, and total number of location count of such type of pixels is denoted by N_{TP}.

False Positive (N_{FP}): When binary pixel value of particular location in S which is 1 altered 0 in corresponding location of the S' this case is called true negative and total number of such type of pixels is denoted by N_{FP}.

True Negative (N_{TN}): When pixel value at corresponding location in both the S and S' is same and equal to 0 then this case is called true negative, and total number of location count of such type of pixels is denoted by N_{TN}.

False Negative (N_{FN}): When binary pixel value of particular location in S which is 0 altered to 1 in corresponding location of the S' this case is called false negative and total number of such type of pixels is denoted by N_{FN}.

On the basis of these test results various metrics are calculated to evaluate similarity between binary input image and binary output image. Some of the most important metrics are defined below.

Negative Rate Matrix (NRM): NRM depends on pixel-wise inequality between input image S and output image S', defined as

$$NRM = \frac{NR_{fu} + NR_{fp}}{2} \tag{4}$$

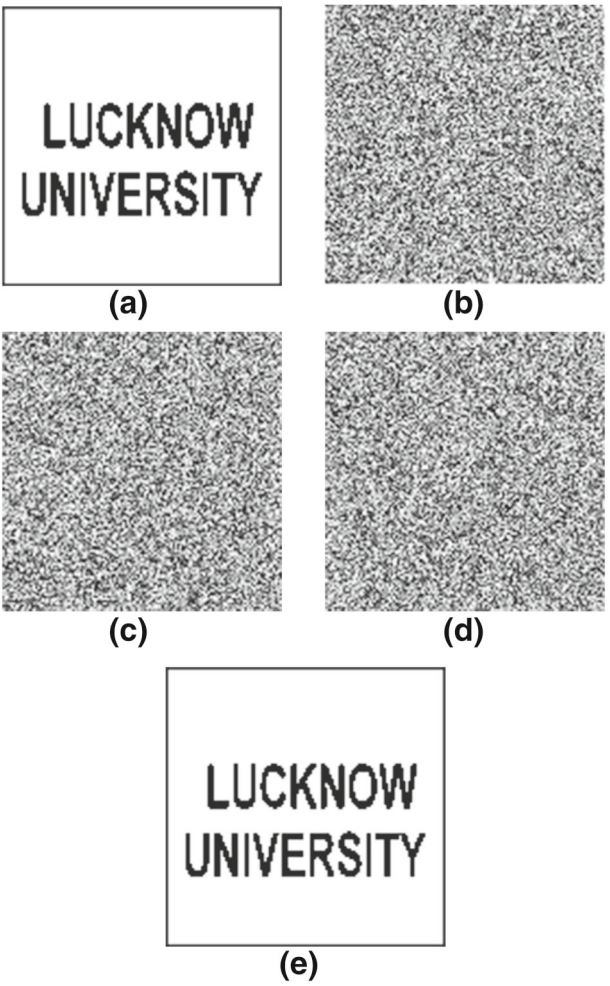

Fig. 5 Experimental results of the (3,3) case for binary image by *Algorithm* 3. **a** Secret image, **b** Share S_1, **c** Share S_2, **d** Share S_3, **e** Revealed secret image ($S_1 \oplus S_2 \oplus S_3$)

where

$$NR_{fn} = \frac{N_{FN}}{N_{FN} + N_{TP}} \tag{5}$$

and

$$NR_{fp} = \frac{N_{FP}}{N_{FP} + N_{TN}} \tag{6}$$

Between two identical images the value of NRM is always 0.

Recall/Sensitivity:

$$Recall = \frac{N_{TP}}{N_{TP} + N_{FN}} \tag{7}$$

Between two identical images the value of Recall will be 1.

Precision:

$$Precision = \frac{N_{TP}}{N_{TP} + N_{FP}} \tag{8}$$

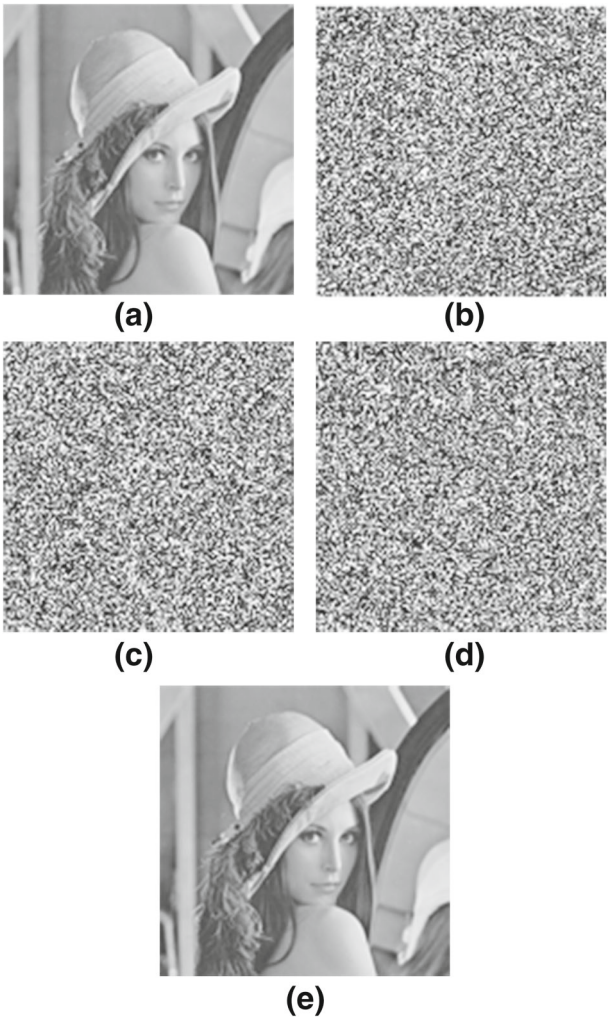

Fig. 6 Experimental results of the (3,3) case for gray-scale image by *Algorithm* 3. **a** Secret image, **b** Share S_1, **c** Share S_2, **d** Share S_3, **e** Revealed secret image ($S_1 \oplus S_2 \oplus S_3$)

Between two identical images the value of Precision will be 1.

F-Measure:

$$FM = \frac{2 \times Recall \times Precision}{Recall + Precision} \tag{9}$$

Between two identical images the value of F-Measure will be 1.

Specificity:

$$Specificity = \frac{N_{TN}}{N_{TN} + N_{FP}} \tag{10}$$

Between two identical images the value of Specificity will be 1.

Balanced Classification Rate(BCR)/Area Under the Curve(AUC):

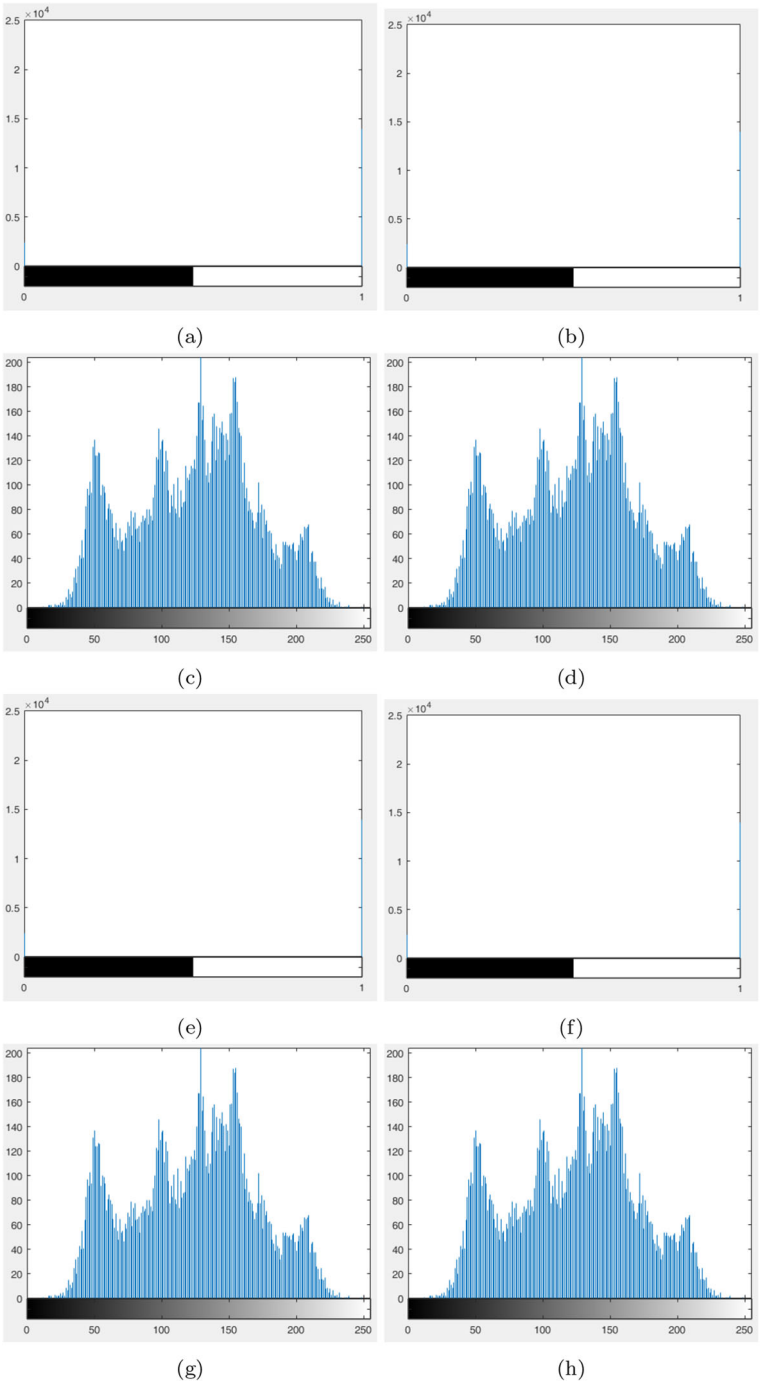

Fig. 7 Histograms of the secret image and revealed secret image: **a** Secret image **b** revealed secret image of Fig. 3, **c** Secret image **d** revealed secret image of Fig. 4, **e** Secret image **f** revealed secret image of Fig. 5, **g** Secret image **h** revealed secret image of Fig. 6

$$BCR = 0.5 \times (Specificity + Sensitivity) \tag{11}$$

Between two identical images the value of BCR/AUC will be 1.

Balanced Error Rate (BER):

$$BER = 100 \times (1 - BCR) \tag{12}$$

Between two identical images the value of the BER is 0.

Structural Similarity Index (SSIM):

$$SSIM(x, y) = \frac{(2 \cdot \mu_x \cdot \mu_y + c_1)(2 \cdot \sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{13}$$

where μ_x, μ_y, σ_x^2, σ_y^2 and σ_{xy} refers to average value of x, average value of y, variance of x, variance of y, covariance of x and y respectively. The value of $SSIM$ is 1 for two identical images and vary from -1 to +1.

Effectiveness of visual cryptography scheme judged on these objective evaluation parameters. The visual cryptography scheme is said to be effective, if the values of these parameters reach towards their ideal values on comparison between input and output image.

Definition 4 Maximum Absolute Error (MAE): It is defined as the maximum absolute difference between the respective pixels of the original image and the reconstructed or obtained by any other modification. Let S(M,N) and S'(M,N) are input and output images, then MAE is defined as

$$MAE = MAX(abs(S(i, j) - S'(i, j))) \tag{14}$$

where, $1 \leq i \leq M$ and $1 \leq i \leq M$

Definition 5 Mean Square Error (MSE) : It represents the cumulative squared error between original and obtained image. It is given by following equation

$$MSE = \frac{\sum_{i=1,j=1}^{M,N}[S(i, j) - S'(i, j)]^2}{M \times N} \tag{15}$$

When original and obtained images are same, value of MSE will be 0. It means lower the value of MSE, lower the difference between original and obtained image

Definition 6 Peak Signal to Noise Ratio (PSNR): This ratio measures the quality of the reconstructed image with respect to the original image. Higher the PSNR, better the quality of the reconstructed image. It is calculated by the following equation

$$PSNR = 10 \times \log_{10}\left(\frac{R^2}{MSE}\right)dB \tag{16}$$

Where, R is the maximum fluctuation in the input image data type. It is 1 for double-precision floating-point data type, and 255 for 8 bit unsigned data type. PSNR value will be ∞ for two identical images.

Experiments and ideal values of the quality metrics for the binary image are listed in Table 6 and for gray image in Table 7. Here, it is clear that experimental and ideal values

of the various quality metrics are same, which means revealed secret image is same as an original secret image in all the cases.

4.4 Validity and numerical analysis

By using the concept of the proposed scheme two examples are illustrated to ensure effectiveness and validity of the proposed scheme. In example 1, a single pixel grayscale secret image is shared among three participants. In example 2, a single pixel grayscale secret image is shared between two participants. The proposed scheme is valid construction if the results of XOR-Ed all the shares in both the examples reconstructs the original secret image.

Example 1 This example demonstrates secret sharing of the proposed scheme for $n = 3$ participants, where the integer pixel value of the single pixel grayscale secret image is 204.

Encryption

Since number of participants are $n = 3$. Therefore, we have to generate construction matrix C_3 using Algorithm 1. After that divide C_3, into C_3^{even} and C_3^{odd} using Algorithm 2.

Now apply the instructions given in Algorithm 3 to encrypt the secret image.

Given $d = 204$, convert d into 8 bit binary representation and assign it to vector V as, $V = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$

Here, $V_8 = 1$, $V_7 = 1$, $V_6 = 0$, $V_5 = 0$, $V_4 = 1$, $V_3 = 1$, $V_2 = 0$ and $V_1 = 0$. Here, Tables 2 and 3 are prepared with respect to the matrices C_3^{even} and C_3^{odd} generated, by Algorithm 2. The next step is table selection for encryption purpose and selection of a particular row of the selected table based on random number t. Now consider $V_8 = 1 \neq 0$, hence select Table 3, next compute a random number t as, $t^o = randi(2^{3-1})$, let us assume $t^0 = randi(2^{3-1}) = 2$ is returned, then select second row of the Table 3 and assign respective values to A_8^1, A_8^2 and A_8^3 as given below $A_8^1 = 0$, $A_8^2 = 1$ and $A_8^3 = 0$. Similarly, for $V_7 = 1 \neq 0$, hence select Table 3, next execute the function $t^0 = randi(2^{3-1})$, let it be $t^0 = 3$, it means take the third row of Table 3 and compute A_7^1, A_7^2 and A_7^3 as, $A_7^1 = 1$, $A_7^2 = 0$ and $A_7^3 = 0$, in the same way select every V_k, where $k = 1$ to 8. The resultant values of A_k^1, A_k^2 and A_k^3 for every V_k are given in Table 4. Now compute integer values of the vectors A^1, A^2 and A^3, according to (1).

Computation of A^1

$$\begin{aligned} A^1 &= A_8^1 \times 2^7 + A_7^1 \times 2^6 + A_6^1 \times 2^5 + A_5^1 \times 2^4 + A_4^1 \times 2^3 + A_3^1 \times 2^2 + A_2^1 \times 2^1 \\ &\quad + A_1^1 \times 2^0 \\ \Rightarrow A^1 &= 0 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ \Rightarrow A^1 &= 93 \end{aligned}$$

Table 2 Table with respect to random number t for C_3^{even}

t^e	A_k^1	A_k^2	A_k^3
1	0	0	0
2	0	1	1
3	1	0	1
4	1	1	0

Table 3 Table with respect to random number t for C_3^{odd}

t^o	A_k^1	A_k^2	A_k^3
1	0	0	1
2	0	1	0
3	1	0	0
4	1	1	1

Computation of A^2

$$\begin{aligned} A^2 &= A_8^2 \times 2^7 + A_7^2 \times 2^6 + A_6^2 \times 2^5 + A_5^2 \times 2^4 + A_4^2 \times 2^3 + A_3^2 \times 2^2 + A_2^2 \times 2^1 \\ &\quad + A_1^2 \times 2^0 \\ \Rightarrow A^2 &= 1 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\ \Rightarrow A^2 &= 150 \end{aligned}$$

Computation of A^3

$$\begin{aligned} A^3 &= A_8^3 \times 2^7 + A_7^3 \times 2^6 + A_6^3 \times 2^5 + A_5^3 \times 2^4 + A_4^3 \times 2^3 + A_3^3 \times 2^2 + A_2^3 \times 2^1 \\ &\quad + A_1^3 \times 2^0 \\ \Rightarrow A^3 &= 0 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\ \Rightarrow A^3 &= 7 \end{aligned}$$

Now, assign values of A^1, A^2 and A^3 to respective location of the shares S_1, S_2 and S_3. Hence, integer values of generating shares are as follows, $S_1 = 93$, $S_2 = 150$ and $S_3 = 7$.

Decryption

To reveal the secret pixel, apply Algorithm 4 to the shares S_1, S_2 and S_3. To do so, perform an XOR-operation between S_1 and S_2 as $temp = S_1 \oplus S_2$, $\Rightarrow temp = 93 \oplus 150 \Rightarrow temp = 203$, now, perform an XOR operation between temp and S_3 as, $R^d = temp \oplus S_3 \Rightarrow R^d = 203 \oplus 7 \Rightarrow R^d = 204$, value of R^d is same as input secret image d. It means Proposed scheme recovers secret image without any loss.

Example 2 This example demonstrates the secret sharing of the proposed scheme for the $n = 2$ participants, where the integer pixel value of the single pixel grayscale secret image is 160.

Table 4 Table for computation of A^1, A^2 and A^3

k	V_k	t	A_k^1	A_k^2	A_k^3
8	1	$t^o = 2$	0	1	0
7	1	$t^o = 3$	1	0	0
6	0	$t^e = 1$	0	0	0
5	0	$t^e = 4$	1	1	0
4	1	$t^o = 3$	1	0	0
3	1	$t^o = 4$	1	1	1
2	0	$t^e = 2$	0	1	1
1	0	$t^e = 3$	1	0	1

Encryption

Here number of participants $n = 2$, therefore, create construction matrix C_2 for $n = 2$, by executing Algorithm 1, and C_2^{even}, C_2^{odd} by executing Algorithm 2.

$$C_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}_{4\times 2}, C_2^{even} = \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}_{2\times 2}, C_2^{odd} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}_{2\times 2}$$

Now, use Algorithm 3 to generate secret shares. To do so first convert d into 8 bit binary numbers and assign it to vector V, hence $V = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$ for further processing follows Table 5, which is created using matrices C_2^{even} and C_2^{odd}. From Table 5,

$A^1 = A_8^1 \times 2^7 + A_7^1 \times 2^6 + A_6^1 \times 2^5 + A_5^1 \times 2^4 + A_4^1 \times 2^3 + A_3^1 \times 2^2 + A_2^1 \times 2^1 + A_1^1 \times 2^0 \Rightarrow A^1 = 197$ $A^2 = A_8^2 \times 2^7 + A_7^2 \times 2^6 + A_6^2 \times 2^5 + A_5^2 \times 2^4 + A_4^2 \times 2^3 + A_3^2 \times 2^2 + A_2^2 \times 2^1 + A_1^2 \times 2^0 \Rightarrow A^2 = 0 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \Rightarrow A^2 = 101$ assign values of A^1 and A^2 to respective locations of the shares S_1 and S_2. Hence, $S_1 = A^1 = 197$, $S_2 = A^2 = 101$.

Decryption

To decrypt the secret image use Algorithm 4,hence $R^d = S_1 \oplus S_2 \Rightarrow R^d = 197 \oplus 101 \Rightarrow R^d = 160$. It is same as the original secret image (Table 6 and 7).

4.5 Security against attacks

The proposed scheme generates meaningless shares due to which subject of suspect for the assailants. Therefore, participants can tamper the content of the shares. Now, a question arises, how shares resist disclosing of information over various attacks which an attacker can do intentionally or unintentionally. A brief description of possible attacks and resistivity of the shares to overcome the effect of the attacks and its impact on revealed secret image is given below:

– **Passive Attacks:** A passive attack attempts to divulge or use information without affecting to resource. Possible passive attacks are release of message contents and traffic analysis. Since the proposed scheme generates meaningless shares which have no clue about the secret image [28]. It means attacker or individual participants are not

Table 5 Table for computation of A^1 and A^2

k	V_k	t	A_k^1	A_k^2
8	1	$t^o = 2$	1	0
7	0	$t^e = 2$	1	1
6	1	$t^o = 1$	0	1
5	0	$t^e = 1$	0	0
4	0	$t^e = 1$	0	0
3	0	$t^e = 2$	1	1
2	0	$t^e = 1$	0	0
1	0	$t^e = 2$	1	1

Table 6 Values of various objective evaluation parameters for revealed secret images of Figs. 3 and 5

Quality Metrics	Simulation cases			
	(2, 2)		(3, 3)	
	Experimental value	Ideal Value	Experimental value	Ideal Value
Precision	1	1	1	1
Recall	1	1	1	1
F-measure(%)	100	100	100	100
SSIM	1	1	1	1
Specificity	1	1	1	1
BCR	1	1	1	1
BER(%)	0	0	0	0
NRM	0	0	0	0

able to disclose the contents of the shares even if they have high computational power. Traffic analysis occurs when an attacker monitors source of information during transmission. When shares are transmitted from participants to intended authority for the purpose of decryption, then the attacker can determine the location and identity but information remains unrevealed.

– **Active Attacks:** Active attacks involve tampering with the contents of the shares. The contents of the shares could be altered by the participants or attackers during transmission with the intention of harm or denial of service to the system. This alteration will reflect in the revealed secret image.

4.6 Computation complexity

For the assessment of the performance of a particular visual secret sharing scheme, it is required to compute the computation complexity of that scheme. The performance of the visual secret sharing assessed on the complexity of decryption. Hence, it is desired to calculate the computational complexity of the proposed scheme. When XOR decryption is applied computational complexity of decryption is proportional to the number of XOR-Ed shares. Let n be the number of shares, computational complexity of decryption is $\mathcal{O}(n)$. The computation complexity assessment of the proposed scheme in comparison to some state-of-the-art approaches is tabulated in Table 8. Here, it is obvious that in comparison to

Table 7 Values of various evaluation parameters for revealed secret images of Figs. 4 and 6

Quality Metrics	Simulation cases			
	(2, 2)		(3, 3)	
	Experimental value	Ideal Value	Experimental value	Ideal Value
MAE	0	0	0	0
MSE	0	0	0	0
PSNR	∞	∞	∞	∞

Table 8 Comparison with other schemes on the basis of computational complexity for the decryption

Schemes	Computational complexity
Ours	$\mathcal{O}(n)$
Tuyls et al. [40]	$\mathcal{O}(n)$
Wu et al. [45]	$\mathcal{O}(n)$
Liu et al. [26]	$\mathcal{O}(n)$
Ou et al. [29]	$\mathcal{O}(n)$
Singh et al. [35]	$\mathcal{O}(n)$
Shyu et al. [33]	$\mathcal{O}(1)$
Chen et al. [6]	$\mathcal{O}(1)$
Guo et al. [14]	$\mathcal{O}(1)$
Lin et al. [22]	$\mathcal{O}(nlog^2n)$
Yang et al. [50]	$\mathcal{O}(nlog^2n)$

schemes [22, 50] based on Shamir's method [28] which requires the evaluation of polynomials and interpolation, proposed scheme requires less time for decryption. The OR-based visual secret sharing schemes [6, 14, 33] shown in table is $\mathcal{O}(1)$requires no computation for decryption. The computational complexity for decryption of the proposed and other XOR-based schemes [26, 29, 35, 40, 45] is relatively higher than that of OR-based schemes. However, the advantageous feature of XOR-based visual secret sharing schemes are that the quality of revealed secret image is superior to OR-based schemes and solves pixel alignment problem.

5 Comparison with the state-of-the-art approaches

The proposed scheme is compared with some state-of-the-art approaches on the basis of subjective evaluation parameters such as the type of the scheme, secret image type, the content of the shares, pixel expansion and contrast-loss. To understand it clearly, brief descriptions about these subjective parameters are as follows:

1. **Type:** Visual secret sharing schemes are categorized into three types $(2, n)$, (k, n) and (n, n) which can reveal secret image by using any 2 or more, any k or more and only n out of n shares respectively.
2. **Decryption:** There are three ways, namely OR operation, XOR operation and compute to reveal the secret. The visual secret sharing schemes which reveal secret on the basis of compute, requires additional information.
3. **Secret image type:** There is four types such as binary, grayscale, grayscale converted to halftone (G-half) and color image converted to halftone (C-half) of secret images are taken into interest by the researchers.
4. **Pixel expansion:** Visual secret sharing schemes either generate expanded (larger than the secret image) or non-expansible shares (same size as secret image).
5. **Contrast-loss:** Visual secret sharing schemes suffers with contrast-loss or not and is responsible for quality of shares as well as revealed secret image.
6. **Content of the shares:** Visual secret sharing schemes either generates meaningless or meaningful shares. Here, meaningless shares are just noise-like images. The meaningful

Table 9 Feature comparison with the state-of-the-art approaches

Schemes	Type	Decryption	Secret Image type	Meaningful Share	Pixel Expansion	Revealed Image Quality
Shyu et al. [33]	(n, n)	OR	Binary	No	No	Low
Chen et al. [6]	(k, n)	OR	Binary	No	No	Low
Guo et al. [14]	(n, n)	OR	Binary	Yes	No	Low
Tuyls et al. [40]	(k, n)	XOR	Binary	No	Yes	High
Wu et al. [45]	(n, n)	XOR	Binary	Yes	No	High
Liu et al. [26]	$(2, n)$	XOR	Binary	No	Yes	High
Ou et al. [29]	(n, n)	XOR	Binary, Halftone	Yes	No	High
Singh et al. [35]	(n,n)	XOR	Binary	Yes	No	High
Proposed	(n, n)	XOR	Binary, **Gray**	No	No	**High(Same)**

shares carry some additional information which is useful to ease the management, but still filled with some noisy pixels.

7. **Revealed image quality:** The quality of the revealed secret image (RIQ) is measured in terms of Low, High and Varying. Here, Low means quality of revealed secret image is poorer than the secret image. High means quality of revealed secret image is same as that of the secret image. Varying means quality varies depending upon the underlying operation to reveal the secret image.

Comparable values for these parameters of the proposed scheme and state-of-the-art approaches are given in Table 9. The schemes proposed by Shyu et al. [33], Chen et al. [6], and Guo et al. [14] are OR-based visual cryptography schemes used to share the binary secret, reconstructs secrets with low contrast due to which quality of the revealed secret image is poor. Although the scheme proposed by Gue et al. generates meaningful shares which facilitates to manage shares easily. The schemes proposed by Tuyls et al. [40], Wu et al. [45], Liu et al. [26], Ou et al. [29] and Singh et al. [35] are XOR-based visual cryptography schemes, limited to share binary or halftone secret image but reveals secret with high visual quality. Proposed scheme based on XOR operation applies to the both binary and grayscale secrets and reconstructs the secret image perfectly with high visual quality. From the Table 9, it is clear that unlike the other schemes the proposed scheme is the only scheme which is directly applicable to binary and grayscale images and reconstructs the secret image perfectly without clarity-loss(same to input secret image)

6 Conclusion and future scope

To deal with the situation where information associated with high quality image (gray) is at risk of disclosing in a single user system (participant, channel etc.). In this paper, an XOR-based $(n, n) - VCS_{XOR}$ visual secret sharing scheme using pixel vectorization is proposed. Which diffuses and disguise secret image into a number of random looking shares distributed to participants in such a way that each participant receive only one share and have no clue about the secret. The proposed scheme applies directly to grayscale secret image without converting to halftone image. The proposed scheme imposes no overhead of storing codebook, generates implicitly. Proposed scheme also resolves the problem of pixel expansion, restriction with the number of participants and lossy recovery of the secret image. It is

also applied to binary image. To achieve the objective three algorithms, construction matrix generation (Algorithm 1), division of construction matrix (Algorithm 2) into basis matrices and $(n, n)-VCS_{XOR}$ visual secret sharing scheme (2) are developed. Numerical illustration and theoretical analysis are provided to validate the correctness of the scheme. Experimental results and analysis show that it is feasible for both binary and grayscale secret images. Results also compared with the state-of-the approaches on the basis of subjective evaluation parameters and computational complexity, found that proposed scheme performs better.

Acknowledgments This work was supported by University Grant Commission, India (RGNF award No : F1-17.1/2014-15/RGNF-2014-15-SC-UTT-87345/(SA-III/Website)) and University of Lucknow, Lucknow.

References

1. Blundo C, De Bonis A, De Santis A (2001) Improved schemes for visual cryptography. Designs Codes Crypto 24:255–278
2. Blundo C, D'Arco P, De Santis A, Stinson DR (2003) Contrast optimal threshold visual cryptography schemes. SIAM J Discret Math 16:224–261
3. Chao HC, Fan TY (2017) XOR-Based progressive visual secret sharing using generalized random grids. Displays 49:6–15
4. Chattopadhyay AK, Nag A, Singh JP, Singh AK (2020) A verifiable multi-secret image sharing scheme using XOR operation and hash function. Multimedia Tools Appl 1–30
5. Chen TH, Li KC (2012) Multi-image encryption by circular random grids. Inf Sci 189:255–265
6. Chen TH, Tsao KH (2011) Threshold visual secret sharing by random grids. J Sys Softw 84:1197–1208
7. Chen TH, Tsao KH (2011) User-friendly random-grid-based visual secret sharing. IEEE Trans Circ Sys Video Technol 21:1693–1703
8. Chen TH, Tsao KH, Lee YS (2012) Yet another multiple-image encryption by rotating random grids. Signal Process 92:2229–2237
9. Chen YC, Horng G, Tsai DS (2012) Comment on "cheating prevention in visual cryptography" IEEE Trans Image Process 21:3319–3323
10. Chen YC, Tsai DS, Horng G (2013) Visual secret sharing with cheating prevention revisited. Digit Sig Process 23:1496–1504
11. Cimato S, De Prisco R, De Santis (2005) Probabilistic visual cryptography schemes. Comput J 49:97–107
12. Deshmukh M, Nain N, Ahmed M (2018) Efficient and secure multi secret sharing schemes based on boolean XOR and arithmetic modulo. Multimedia Tools Appl 77:89–107
13. Fu Z, Cheng Y, Yu B (2019) Perfect recovery of XOR-based visual cryptography scheme. Multimedia Tools Appl 78:2367–2384
14. Guo T, Liu F, Wu C (2014) K out of k extended visual cryptography scheme by random grids. Signal Process 94:90–101
15. Hofmeister T, Krause M, Simon HU (2000) Contrast-optimal k out of n secret sharing schemes in visual cryptography. Theor Comput Sci 240:471–485
16. Hu CM, Tzeng WG (2006) Cheating prevention in visual cryptography. IEEE Trans Image Process 16:36–45
17. Jia X, Wang D, Chu Q, Chen Z (2019) An efficient XOR-based verifiable visual cryptographic scheme. Multimedia Tools Appl 78:8207–8223
18. Kafri O, Keren E (1987) Encryption of pictures and shapes by random grids. Opt Lett 12:377–379
19. Koga H, Ueda E (2006) Basic properties of the (t, n)-threshold visual secret sharing scheme with perfect reconstruction of black pixels. Designs Codes Crypto 40:81–102
20. Lee YS, Chen TH (2012) Insight into collusion attacks in random-grid-based visual secret sharing. Signal Process 92:727–736
21. Lian C, Pang L, Liang J (2014) Generalized random grid-based visual secret sharing for general access structures. Comput J 58:2426–2442
22. Lin CC, Tsai WH (2004) Secret image sharing with steganography and authentication. J Sys Softw 73:405–414
23. Lin CH, Chen TH, Wu YT, Tsao KH, Lin KS (2014) Multi-factor cheating prevention in visual secret sharing by hybrid codebooks. J Vis Commun Image Represent 25:1543–1557

24. Lin CH, Lee YS, Chen TH (2015) Friendly progressive random-grid-based visual secret sharing with adaptive contrast. J Vis Commun Image Represent 33:31–41
25. Lin SJ, Chung WH (2011) A probabilistic model of (t, n) visual cryptography scheme with dynamic group. IEEE Trans Inf Foren Sec 7:197–207
26. Liu F, Wu C (2014) Optimal XOR based (2, n)-visual cryptography schemes. In: International workshop on digital watermarking, pp 333–349
27. Liu F, Wu CK, Lin XJ (2009) The alignment problem of visual cryptography schemes. Designs Codes Crypto 50:215–227
28. Naor M, Shamir A (1994) Visual cryptography. In: Workshop on the theory and application of of cryptographic techniques, pp 1–12
29. Ou D, Sun W, Wu X (2015) Non-expansible XOR-based visual cryptography scheme with meaningful shares. Signal Process 108:604–621
30. Pang L, Miao D, Lian C (2016) User-friendly random-grid-based visual secret sharing for general access structures, vol 9, pp 966–976
31. Praveen K, Sethumadhavan M (2018) Cheating immune visual cryptographic scheme with reduced pixel expansion. In: Progress in advanced computing and intelligent engineering, pp 257–265
32. Salehi S, Balafar MA (2014) Visual multi secret sharing by cylindrical random grid. J Inf Sec Appl 19:245–255
33. Shyu SJ (2009) Image encryption by multiple random grids. Patt Recogn 42:1582–1596
34. Singh N (2018) XOR Encryption techniques of video steganography: a comparative analysis. In: International conference on intelligent systems design and applications, pp 203–214
35. Singh P, Raman B, Misra M (2018) A (n, n) threshold non-expansible XOR based visual cryptography with unique meaningful shares. Signal Process 142:301–319
36. Tan L, Lu Y, Yan X, Liu L, Zhou X (2020) XOR-Ed visual secret sharing scheme with robust and meaningful shadows based on QR codes. Multimedia Tools Appl 79:5719–5741
37. Thien CC, Lin JC (2003) An image-sharing method with user-friendly shadow images. IEEE Trans Circ Sys Video Technol 13:1161–1169
38. Tsai DS, Chen TH, Horng G (2007) A cheating prevention scheme for binary visual cryptography with homogeneous secret images. Patt Recogn 40:2356–2366
39. Tuyls P et al (2002) A polarisation based visual crypto system and its secret sharing schemes. https://eprint.iacr.org/2002/194
40. Tuyls P et al (2005) XOR-Based visual cryptography schemes. Designs Codes Crypto 37:169–186
41. Wu X, Liu T, Sun W (2013) Improving the visual quality of random grid-based visual secret sharing via error diffusion. J Vis Commun Image Represent 24:552–566
42. Wu X, Ou D, Dai L, Sun W (2013) XOR-Based meaningful visual secret sharing by generalized random grids. In: Proceedings of the first ACM workshop on information hiding and multimedia security, pp 181–190
43. Wu X, Ou D, Liang Q, Sun W (2012) A user-friendly secret image sharing scheme with reversible steganography based on cellular automata. J Syst Softw 85:1852–1863
44. Wu X, Sun W (2012) Random grid-based visual secret sharing for general access structures with cheat-preventing ability. J Sys Softw 85:1119–1134
45. Wu X, Sun W (2013) Generalized random grid and its applications in visual cryptography. IEEE Trans Inf Foren Sec 8:1541–1553
46. Wu X, Sun W (2013) Improving the visual quality of random grid-based visual secret sharing. Signal Process 93:977–995
47. Wu X, Sun W (2013) Random grid-based visual secret sharing with abilities of OR and XOR decryptions. J Vis Commun Image Represent 24:48–62
48. Wu X, Sun W (2014) Improved tagged visual cryptography by random grids. Signal Process 97:64–82
49. Yan X, Wang S, El-Latif AAA, Niu X (2015) Random grids-based visual secret sharing with improved visual quality via error diffusion. Multimedia Tools Appl 74:9279–9296
50. Yang CN, Chen TS, Yu KH, Wang CC (2007) Improvements of image sharing with steganography and authentication. J Sys Softw 80:1070–1076
51. Zhou Z, Arce GR, Di Crescenzo G (2006) Halftone visual cryptography. IEEE Trans Image Process 15:2441–2453