Journal of Universal Computer Science, vol. 21, no. 12 (2015), 1536-1562
submitted: 2/3/15, accepted: 11/11/15, appeared: 1/12/15 © J.UCS

On Real-valued Visual Cryptographic Basis Matrices

Neil Buckley
(Department of Mathematics and Computer Science, Liverpool Hope University
United Kingdom
08008783@hope.ac.uk)

Atulya K. Nagar
(Department of Mathematics and Computer Science, Liverpool Hope University
United Kingdom
nagara@hope.ac.uk)

Subramanian Arumugam
(National Centre for Advanced Research in Discrete Mathematics, Kalasalingam University
Anand Nagar, Krishnankoil-626 126, Tamil Nadu, India
s.arumugam.klu@gmail.com)

Abstract. Visual cryptography (VC) encodes an image into noise-like shares, which can be stacked to reveal a reduced quality version of the original. The problem with encrypting colour images is that they must undergo heavy pre-processing to reduce them to binary, entailing significant quality loss. This paper proposes VC that works directly on intermediate grayscale values per colour channel and demonstrates real-valued basis matrices for this purpose. The resulting stacked shares produce a clearer reconstruction than in binary VC, and to the best of the authors' knowledge, is the first method posing no restrictions on colour values while maintaining the ability to decrypt with human vision. Grayscale and colour images of differing entropies are encrypted using fuzzy *OR* and *XOR*, and their PSNR and structural similarities are compared with binary VC to demonstrate improved quality. It is compared with previous research and its advantages highlighted, notably in high quality reconstructions with minimal processing.

Keywords: visual cryptography; secret sharing; cryptanalysis; colour images; image processing
Categories: E.3, H.3.5, I.4

1 Introduction

Conventional cryptography relies on key distribution to valid participants, but two versions of a new cryptographic paradigm were created in 1987 by Kafri and Keren [Kafri, 87] and 1994 by Naor and Shamir [Naor, 94], that do not rely on such public and private keys. These methods, respectively random grids and visual cryptography (VC), are types of visual secret sharing (VSS), which encodes a secret binary image into a set of shadow images, or shares, such that stacking a qualified subset of them reveals a reduced quality version of the original secret. These shares form an access structure, containing qualified and forbidden combinations.

Shadow images comprise random sequences of black and white (or transparent) pixels. As such, they are indistiguishable from random noise, making it impossible for forbidden subsets to decode the secret. Therefore, not only does VSS eliminate key distribution, but it exhibits perfect information theoretic security. The work of Ateniese, et al. [Ateniese, 96:2] proposed methods, such as cumulative array, for constructing schemes for any access structure. However, they yielded far from optimal reconstructed image quality (i.e. *contrast*), leading to years of research into this problem, such as that of [Arumugam, 12], who improved the contrast of a particular type of general access structure, and [Liu, 10], who built large access structures from smaller ones.

The major drawback is that only black and white images can be directly encoded. Grayscale and colour images must first undergo pre-processing (in the form of error-diffusion and colour decomposion, as in [Duraisamy, 13]), to reduce them to binary form. Even the best error-diffusion algorithms, such as Floyd-Steinberg [Floyd, 76] produce lossy images, leaking information from the image. When the shares of a VC scheme are combined in the decryption phase, the result is a highly lossy version of the original secret image, therefore conventional (binary) visual cryptography entails two stages of quality loss. Although much prior work exists in colour VC, the range of colours is limited either by definition or by large resulting pixel expansions.

In this paper, a novel VC method, real-valued visual cryptography (RVC), is presentated that encodes grayscale and colour values directly into the VC scheme, eliminating quality loss emerging from error-diffusion and bringing about clearer reconstructions than previously achieved using methods that permit decryption using both human vision or computation. The latter entails low computational cost fuzzy, as opposed to binary, operations for decrypting the secret, and to the best of the authors' knowledge, is the first VSS proposal that removes all colour restrictions and error-diffusion requirements in such a cryptographic method. Furthermore, pixel exansions are improved versus prior methods. For example, a pixel expansion of four is demonstrated in Section 5 for a 10-colour scheme. Existing VSS methodologies are explored and redefined to relax requirements to real-valued (i.e. grayscale) input and output. The contributions presented herein are:

- A new version of VC – RVC – that eliminates image pre-processing requirements while maintaining the theoretical ability to physically stack shares for visual decryption.
- Redefinition of contrast calculation.
- Fuzzy decryption operations.
- A simulated annealing method or the construction of RVC schemes.

The remainder of the paper is structured as follows: Section 3 provides the background to Naor-Shamir VC. Section 4.1 formulates the mapping of RGB values onto basis matrix values. In Section 4.2, contrast calculation is redefined for real-valued schemes, and Section 4.3 likewise redefines the operations required for decryption. In Section 4.4, RVC scheme construction is described, and Section 5 gives the results of encryption and decryption for grayscale and colour images of various entropies and colour distributions. There is also a comparative analysis with various prior works in the field. Finally, Section 6 concludes the study and proposes further research directions, notably the new mathematical challenges that this innovation presents.

2 Related Work

2.1 Binary Visual Secret Sharing

Although Naor and Shamir's original work on VC remains a popular VSS methodology, they were mindful of the contrast loss problem using the *OR* operation for reconstruction and wrote a follow-up paper proposing a "Cover semi-group" [Naor, 96]. However, they conceded the impossibility of building any (*k, n*) access structures for $k \geq n \geq 3$, thus limiting its applicability. They also expressed concern over its inability to conceal colour images.

In the same year, [Ateniese, 96:2] proposed the cumulative array method for constructing VC basis matrices (which are used to encode secret pixels into shares) from $(|Z_M|, |Z_M|)$ access structures, where Z_M is the set of maximal forbidden subsets of participants. Although it is easily applicable to any required access structure, the resulting pixel expansion, hence contrast (given their mutual inverse proportionality) is far from optimal. For instance, a (3, 6)-VC scheme (VCS) produces pixel expansion 25. However, it is shown in [Hofmeister, 00] that the optimal value is 10.

In 2004, [Yang, 04] devised the first VC method without pixel expansion applicable to any access structure. The method relies on probabilistically setting share pixels to black or white, such that the mean count of black-representative pixels in the reconstruction is higher than for white-representative pixels. This *probabilistic VC* (PVC) is distinguished from conventional, or *deterministic VC* (DVC), in its use of random numbers to select individual share pixel values. Whereas the reconstruction contrasts of DVC can be easily and precisely calculated (see equation (2)), those in PVC are not. Moreover, constrasts resulting from the original PVC work were lower than for DVC. However, more recent work, such as that of [Wang, 11], has considerably improved it. Indeed, their work introduces the concept of a user-defined pixel expansion.

2.2 Colour Visual Secret Sharing

The first to study colour VSS was [Verheul, 97], who considered a secret pixel of colour c_i, where the permitted number of colours is *c*, and $0 \leq i \leq c-1$. They considered the result of stacking coloured share pixels as a "generalized colour", where the presence of a black pixel results in a black reconstruction, and all concurrent pixels of the same colour produce that colour in the reconstruction. Additionally, they provide a construction method resulting in *c* basis matrices, each element of which is an integer in $\{0, \ldots, c-1\}$.

A method with improved contrast was devised by [Blundo, 01], who used *c*-colour (2, *n*) and (*n, n*) schemes, with the requirement (proof given in their paper) that $c > n$. They provide two algorithms, one yielding optimal pixel expansion with maximal contrast, the other yielding a trade-off between these metrics. Pixel expansions in those works are still high, and image preprocessing is still required. Interestingly, an earlier work [Koga, 98] devised a simple method, again using *c* colours, using a colour lattice comprising additive and subtractive models. They construct basis matrices with pixel expansion $m = ck^{n-1}$ for (*n, n*)-VCS, i.e. the conventional optimal pixel expansion multiplied by the number of colours. They achieve this by concatenating combinations of black- and white-representative conventional basis matrices and substituting white share pixels (i.e., zeros) with the required colours. However, their method is limited to (*n, n*)-VCS.

In [Adhikari, 05], the ratio of correctly reconstructed pixels was increased compared to [Koga, 98]. The method entailed either one of two colour sets, $\{C, M, Y\}$ or $\{C, M, Y, R, G, B\}$, each colour requiring its own basis matrix. A further *c*-colour methods are provided by [Cimato, 05], followed up by [Cimato, 07], who demonstrate 3-colour schemes and compute a lower pixel expansion bound of $2^{n-1}c-1$.

Although such methods use mutli-value basis matrices, the preferred technique amongst researchers, such as [Chen, 11] and [Wu, 13] is a combination of halftoning and colour decomposition. Here, a colour image is decomposed into its three base colours (either additive or subtractive), and each of them is treated as an independent grayscale image. The gray levels are diffused into binary form using an error-diffusion algorithm such as Floyd-Steinberg. For example, [Hou, 03] decomposed colour pixels to their *C*, *M* and *Y* components before encoding using 2×2 colour blocks comprising the subtractive colours, white and black.

Colour decomposition is applicable to any type of VSS. The shares are constructed by recomposing the colours, and [Wu, 13] produced arguably the best colour image reconstruction using this and void-and-cluster in their recent work. However, information is inevitably lost in the necessary reduction to binary values. This is particularly deleterious when sharing high-entropy images, as further information is lost in the VSS encoding.

More recently, [Christy, 15] used a stochastic method, in the form of a feed-forward artificial neuron network, to generate extended VC schemes. In such schemes, the shares themselves take on the appearance of cover images, detracting attention from hackers who might be curious about the meaning and purpose of purely noise-like shares. Although they achieved a little higher than 41 peak-to-signal noise-ratio, they do preprocess the image to reduce each channel to binary form.

Interestingly, [Sugawari, 15] also recently developed a new colour VC method without the need for preprocessing, but it does not involve simple transparent shares or computational stacking, but high-order retarder films. Conventional retarder films, commonly used to bring about three-dimensional cinematography, rotate polarized light by fixed angles. However [Sugawari, 15] propose arbitrary angle rotation to create colour VC shares.

All of the aforementioned research falls under the category of visual secret sharing, as decryption can be carried out visually and computationally. Many purely computational methods have also been developed, such as [Lukac, 05], who encypted colour pixel values as numeric data, requiring conversion to binary and simple binary operations to combine shares. Although this and similar methods produce lossless reconstruction, they are examples of visual secret sharing, due to lack of ability to visually decode from stacking shares..

3 Naor-Shamir Visual Cryptography

A secret binary image, I , is encrypted into set H of n shadow images, only qualified subsets of which can be printed onto transparencies and stacked to reveal I', a lossy reconstruction of the secret. Subsets comprise superset $\Gamma = (\Gamma_{qual}, \Gamma_{forb})$, where Γ is the access structure defining qualified and forbidden subsets, and $\Gamma_{qual} \cap \Gamma_{forb} = \varnothing$. For simplicity, $\Gamma_{forb} = \{X : X \in \Gamma, X \notin \Gamma_{qual}\}$. Access structures can be threshold or general. The latter explicity defines each $X \in \Gamma$, whereas the former assumes $\Gamma_{qual} = \{\forall X : |X| \geq k\}$, producing threshold (*k, n*)-VCS.

The procedure, given below, takes two collections of binary matrices, C_0, C_1, forming the basis of a scheme. These are therefore referred to as *basis matrices*. If $S_i \in C_i, i \in \{0,1\}$, then all possible columns permutations of S_i comprise the respective collection. In this paper, the collections are ignored in favour of referring directly to the basis matrices S_i, with $S_i \sim X$ denoting that matrix X is equal to S_i up to column permutation.

Each such matrix has dimensions $n \times m$, where m is the *pixel expansion*. To encode a secret pixel, one member of C_0 or C_1 if selected if it is a white, resp. black, pixel. To encode shadow H_j, the *j*th row of S_i is selected and converted into a rectangular matrix of "subpixels" whose dimensions are as close to each other as possible (ideally a square matrix, if *m* is a square number, thus retaining aspect ratio). Each secret pixel is replaced by these subpixel blocks to form shadow images, as in Figure 1. Given that stacking is equivalent to binary *OR*, $\mathrm{I}' = \mathrm{H}_1 \otimes \ldots \otimes \mathrm{H}_n$. Figure 1 illustrates a simple (2, 2)-VCS.

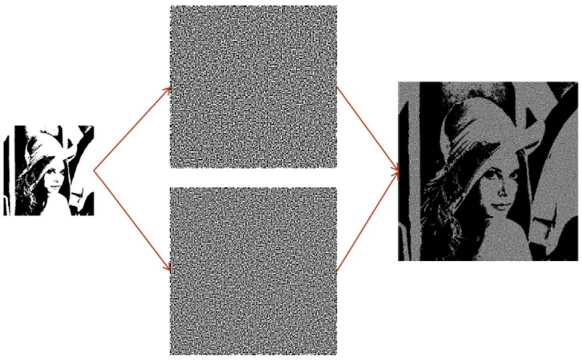


Figure 1: Encryption and decryption of a (2, 2)-VCS

Given contrast reduction and pixel expansion, a major challenge is the design of basis matrices that optimize these metrics. A classical way to contruct them is via accumulative arrays [Ateniese, 96:2]. Although it is efficient and accurate, the CA method rarely results in optimal schemes, because impracticle pixel expansions arise with large k and n.

4 Real-valued Visual Secret Sharing

In this section, RVC scheme construction is demonstrated, along with the pre-requisitives for encryption, decryption and calculation of contrast. In 4.1, a simple equation is presented for converting grayscale or RGB values to real values. In 4.2, the standard contrast calculation of binary VC is generalised for RVC. As binary image decryption relies on binary operations for decryption, these are generalised to fuzzy operations in 4.3. Finally, in 4.4, the RVC methodology is introduced.

4.1 Mapping Real Values to Grayscale Colours

For I to be a valid input into a VSS scheme, it must be binary. Error-diffusion methods can be used on grayscale (and colour) images, but signifant information is lost, as evidenced in Figure 2, below. Furthermore, the result is a set of binary shares (i.e., each colour channel is binary), making it diffult to conceal additional "subliminal" messages in them using steganography.

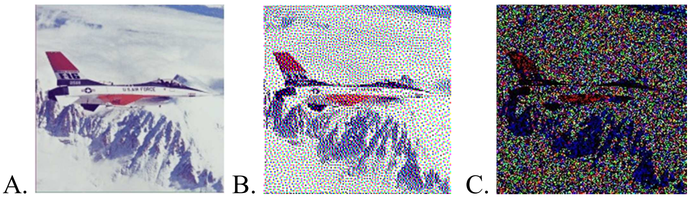


Figure 2: A: Secret image, B: Halftoned image (Floyd-Steinberg), C: Final VSS decryption

The proposed real-valued VC (RVC), avoids this information and quality loss by taking fractional colour values directly as input. Here, the grayscale secret image is a matrix of real values from 0 to 1 inclusive. If an RGB model is used with values ranging from 0 to 255, the pixel at position (*i, j*) of colour channel h is represented as,

$$\mathrm{I}_{ijh} = 1 - \frac{\text{pixel grayscale value}}{255} \qquad (1)$$

where the pixel grayscale value is in $0,\ldots,255$, i.e. the 8-bit colour value of the pixel in the respective channel. Given that a colour image comprises three combined grayscale images (of resp. red, green and blue hues), each channel can be taken as a separate grayscale image.

4.2 Contrast

The formulae given (in, for example [Ateniese, 96:1]; [Chen, 11]) to calculate reconstructed image contrast in VSS schemes assume binary pixel values, but these must be generalizes to real values. Relative contrast, α, in binary VC is defined as the mean difference between black- and white-representative pixels (or subpixel matrices) in the reconstruction. In VC, if $X \in \Gamma_{qual}$ and S_0^X (resp. S_1^X) is the vector resulting from the *OR* of rows X_1, X_2, etc. of s_0 (resp. s_1), then,

$$\alpha = \frac{\left|H(S_0^X) - H(S_1^X)\right|}{m}, \tag{2}$$

where *H(.)* denotes Hamming weight.

An RVC scheme requires basis matrices $s_i, i = \{1, ..., c\}$, where *c* is the number of colours. (2) can thus be generalizes as,

$$\alpha = \frac{\sum_{i=1}^{c-1}\left|H(S_i^X) - H(S_{i+1}^X)\right|}{m} \tag{3}$$

4.3 Operations for Decryption

4.3.1 Fuzzy *OR*

RVC needs fuzzy equivalents of *OR* and *XOR*, respectively denoted $\otimes^{fuz}$ and $\oplus^{fuz}$. The algorithms given in [Zadeh, 65] and other works on fuzzy logic are insuffifient. For example, the classical fuzzy-OR definition simply takes the maximum value. However, Figure 3 illustrates the result of stacking two regions of colour value 0.5. The result of classical fuzzy-*OR* is the maximum, which is 0.5, but the stack is clearly darker than *A* or *B*. In fact, its grayscale value is 0.75 according to equation (1).

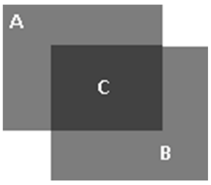


Figure 3: Stack of colours A and B (here A=B=0.5) to produce colour C

Based on this, equation (4) is given for the fuzzy *OR* of shadow image subpixels $\mathrm{H}_1[x,y]$ and $\mathrm{H}_2[x,y]$:

$$b_1 \otimes^{fuz} b_2 = 1-(1-b_1)(1-b_2), b_i = \mathrm{H}_i[x,y] \tag{4}$$

Generalised for *k* shares is the iterative (5):

$$\begin{aligned} &b_1 \otimes^{fuz} ... \otimes^{fuz} b_k = f\left(1-(1-b_i)(1-b_{i+1})\right),\\ &f(b_j) = 1-(1-b_j)(1-b_{j+1}),\\ &i, j = 1, ..., k-1, b_i = \mathrm{H}_i[x,y] \end{aligned} \tag{5}$$

To prove the correctness of equations (4) and (5), let us regard a pixel of colour value φ (e.g. 0.3) containing a ratio of $(\varphi : 1-\varphi)$ black to white pigments. Clearly, stacking pigment values is equivalent to the binary *OR* operation on those values, since individual pigments have values 0 or 1 only (that is, a printer ink of a given colour is either applied, or it is not). Superimposing a second pixel of value λ (of pigment ratio $(\lambda : 1-\lambda)$) onto the first, there are four possibilities, summarised in Table 1.

Pigment of φ	Pigment of λ	Probability
0	0	$(1-\varphi)(1-\lambda)$
0	1	$(1-\varphi)\lambda$
1	0	$\varphi(1-\lambda)$
1	1	$\varphi\lambda$

Table 1: Pixel pigment superimposiiton probabilities

The latter three are relevant, since they are the stacks resulting in an existant pigment. Summing these gives $(1-\varphi)\lambda + \varphi(1-\lambda) + \varphi\lambda$ = $\varphi + \lambda - \varphi\lambda$ = $1-(1-\varphi)(1-\lambda)$, and equation (4) clearly follows from this. (5) is simply an iterative version of (4), taking each successive superimposition as the new φ pigment, therefore its correctness follows.

4.3.2 Fuzzy *XOR*

An advantage of VSS is the ability to decrypt without computation, by physically stacking shares. However, the information theoretic security of VSS is also available while using computation. There has been much research into applications of computational decryption, such as [Wang, 13], who use *XOR* to losslessly decrypt multiple images from one random grid scheme.

A definition of $b_1 \oplus^{fuz} b_2$ is thus needed. [Hernandez, 11] describes a "least sensitive" interpretation, based on the need for a definition that, given $b_3 = b_1 \oplus^{fuz} b_2$, $(b_1 + \Delta) \oplus^{fuz} (b_2 + \Delta) \approx b_3$ for small Δ. Here, this definition is denoted subscript *LS*. They prove that such a definition can be expressed as:

$$b_1 \oplus_{LS}^{fuz} b_2 = \min\left(\max(b_1, b_2), \max(1-b_1)(1-b_2)\right), b_i = \mathrm{H}_i[x, y] \qquad (6)$$

4.4 RVC Schemes

Two additional parameters are introduced in an RVC scheme (RVCS), basis matrix count (μ) and colour value count (κ). These are part of the definition of a threshold access $(k, n, \mu, \kappa) - RVCS$ based on $n \times m$ matrices $S_i, i = \{1, 2, ..., \mu\}$ with pixel expansion *m*. A Naor-Shamir VCS is seen as a special case in which $\mu = \kappa = 2$. General access structures (see [Ateniese, 96:2]), are denoted $(\Gamma, \mu, \kappa) - RVCS$, where $\Gamma = \{\Gamma_{qual}, \Gamma_{forb}\}$ is the access structure comprising all explicity defined qualified and forbidden share subsets.

An RVCS therefore has μ basis matrices, where S_i encodes mapped grayscale value $\frac{i-1}{\mu-1}$. For instance, in a 3-coloured scheme, the mapped values would be 0, 0.5 and 1, each encoded by a basis matrix. Permitted matrix element values are $\frac{i-1}{\kappa-1}, i \in \{1,2,...,\kappa\}$. An example of a valid a (3, 3, 3, 3)-RVCS basis is,

$$S_1 \sim \begin{bmatrix} .5 & 1 & .5 & 0 \\ 0 & 0 & 1 & 0 \\ .5 & 1 & 1 & 0 \end{bmatrix}, S_2 \sim \begin{bmatrix} 1 & 0 & .5 & .5 \\ 0 & 1 & 0 & 0 \\ 1 & .5 & 0 & 1 \end{bmatrix}, S_3 \sim \begin{bmatrix} 0 & 1 & .5 & .5 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & .5 & 1 \end{bmatrix}$$

According to [Kafri, 87], the mean contrast for minimal qualified subsets (i.e. *k=2*) is 0.25 in this example. This is also the contrast resulting from stacking all three shares. In this case, $\mu = \kappa$, i.e. the number of possible colours is equal to the number of basis matrices, but this need not be true. κ is chosen based on the number of grayscale values each basis matrix element can potentially take, whereas higher values of μ are chosen to create a finer grain encoding of colour ranges. In practice, its maximum value is 256, wherein each RGB colour value in 0,…,255 has its own respective matrix, but experiments have shown that such schemes are difficult to derive.

The deterministic construction of these matrices remains a challenge in RVC. Note in [Ateniese, 96:2] the need to construct basis $\{S_0, S_1\}$ for an (*n, n*)-VCS. This is, as discussed in Section 4, simple for binary encoding, but not clear for real-valued encoding.

Simulated annealing is therefore proposed to stochastically derive RVC basis matrices for required parameters. However, for small *k*, *n* and Γ, a brute-force method is applicable, whereby basis matrices of decreasing *cost* are reported.

A candidate solution, *CS*, is generated according to,

$$CS \sim \left\{ \frac{(\kappa-1)rand(0,1)]}{\kappa-1} \right\}^{\mu mn} \qquad (7)$$

where ~ indicates vector permutation and each sequential *m*-sized sub-vector forms the next respective row of $S_i, i = \{1,...,\mu\}$. Important to note is that row $j, j = 1,...,n$ of all basis matrices in the candidate are random column permutations of each other, which is crucial to maintain security, as discussed in Section 5.4.

The objective function is given in Algorithm 1. Although it is efficient, having at most $O(n^2)$ complexity, when executed in a heuristic, the run time depends hugely on $\binom{n}{k}$ and $\min|\Gamma_{qual}|$. If the latter is 2, a valid basis is normally found within 30 seconds on a personal computer. This is sufficient for all graph access structures.

Algorithm 1: Objective Function for Stochastic RVC Basis Generator

Input: *CS*, Γ_{qual} and preferred participant subset, $\Gamma_{pref} \in \Gamma_{qual}$

```
Output: $S_i, i = \{1,...,\mu\}$, costValue
Procedure:
Extract $S_i, i = \{1,...,\mu\}$ from CS
$rewardConst \leftarrow$ reward constant, $punishConst \leftarrow$ punishment constant
$\beta_i \leftarrow H(S_i^{\Gamma_{pref}}), i = \{1,...,\mu\}$
Put $\beta_i$ into ascending order and re-order the basis matrices according to $\beta_i$.
For each $Y \in \Gamma_{qual}$, do,
    $\alpha \leftarrow$ contrast of stack Y according to (5)
    $costValue \leftarrow costValue + rewardConst * \alpha$
End For
For each $Y \in \Gamma_{forb}$, do,
    $\alpha \leftarrow$ contrast of stack Y according to equation (5)
    $costValue \leftarrow costValue - rewardConst * \alpha$
End For
```

In conventional VC, the *i*th row of both basis matrices must have the same Hamming weight. In RVC, they are must be equal up to column permutation (as detailed in Section 6.4). That is,

$$S_i[j] \sim S_{i+1}[j], i = \{1,...,\eta\}, j = \{1,...,n\}. \qquad (8)$$

Any basis not exhibiting this property is rejected in the evaluation phase, as it is not secure. Indeed, this guarantees that individual shares cannot leak the secret and guarantees perfect security in (2, *n*) schemes, as the only forbidden subsets are the individual shares themselves.

The annealer additionally requires a nearest neighbor and acceptance probability algorithm, as used in the SA architecture in Figure 5. Both algorithms follow:

Algorithm 2: Nearest Neighbour

```
Input: $CS_{curr} = CS$, m, n, $\mu, \kappa$
Output: $CS_{new}$
Procedure:
Extract $S_1,...,S_\mu$ from $CS_{curr}$
$r_1 \leftarrow rand(\{1,...,\mu\})$ /RANDOM MATRIX
$r_2 \leftarrow rand(\{1,...,n\})$ //RANDOM ROW OF MATRIX
$r_3 \leftarrow rand(\{1,...,m\})$ //RANDOM SUBPIXEL IN ROW
$r_4 \leftarrow rand(\{-1,+1\})$ //RANDOM DIRECTION
$method \leftarrow rand(\{"alter","shift","permutate"\})$

If method = "alter", then,
    $p \leftarrow S_1[r_2][r_3]$, $q \leftarrow p + \frac{r_4}{\kappa - 1}$
    If q > 1 or q < 0, then $r_4 \leftarrow -r_4$,
    End If
    $p' \leftarrow p + \frac{r_4}{\kappa - 1}$ and replace first instance of p with $p'$ in all $S_i[r_2], i = 1,...,\mu$
Else If method = "shift", then,
    If $r_4 = -1$, then, $S_{r_1}[r_2] \leftarrow S_{r_1}[r_2]$ □ 1  //LEFT SHIFT OF ELEMENTS
    Else If $r_4 = +1$, then, $S_{r_1}[r_2] \leftarrow S_{r_1}[r_2]$ □ 1  //RIGHT SHIFT OF ELEMENTS
    End If
Else If method = "permutate", then,
    Randomly permutate $S_{r_1}[r_2]$
End If
```

Hence a neighbor is selected by either altering the same pixel value in the same row of all matrices, shifting the elements of one matrix row by one position, or randomly permutating one matrix row. Crucially, each method maintains the relation between all matrices that all rows $j, j = 1,...,n$ are equal up to permutation.

Algorithm 3: Acceptance Probability

Input: E_{curr}, E_{new}, T

Output: P

Procedure:

```
If $E_{new} > E_{curr}$, then, $P \leftarrow 1$
Else, $P \leftarrow e^{\frac{E_{curr} - E_{new}}{T}}$
End If
```

Finally, Algorithm 4 describes the use of a derived set of RVC basis matrices to encode a secret image into shares.

Algorithm 4: Encoding a Secret Image

Input: $I, S_i, i = 1,...,\mu$

Output: $H_i, i = 1,...,n$

Procedure:

```
$\Lambda \leftarrow \left\{ \frac{i-1}{\mu - 1} \right\}, i = 1,...,\mu$
For each pixel at position x,y in $I$, do,
    $s \leftarrow I[x, y]$
    $i \leftarrow \{ i : |s - \Lambda_i| = \min(|s - \Lambda_i|) \}$
    $B \leftarrow S_i$
    For $i \leftarrow 1,...,n$, do,
        $V \leftarrow B[i]$
        $M \leftarrow V$ reshaped to a $p \times q$ matrix, where $p \approx q$
        Place subpixel block M at position x,y in $H_i$
    End For
End For
```

Crucially, a common problem in binary VC is the reversal of colours in the reconstruction (producing a negative image). This is exacerbated in RVC, resulting in (seemingly) unpredictable permuations of reconstructed grayscale values. It is for this reason that the collection of RVC basis matrices must be reordered to optimize for the visual quality of a "preferred" share stack, Γ_{pref}. The simulated annealing algorithm is illustrated in Figure 5. Given the superior quality using *XOR*, $\oplus_{LS}^{fuz}$ is defined using equation (6) to improve the contrast of the decoded secret.

Algorithm 1 was applied to evolve the (2, 3, 10, 255)-RVCS below (each element rounded to two decimal places), with *m=4*, producing the shadow images in Figure 4.

$$S_1 \sim \begin{bmatrix} 0.99 & 0.59 & 0.01 & 0.76 \\ 0.99 & 0.03 & 0.06 & 0.39 \\ 0.85 & 0.45 & 0.03 & 0.67 \end{bmatrix}, S_2 \sim \begin{bmatrix} 0.59 & 0.01 & 0.99 & 0.76 \\ 0.39 & 0.03 & 0.99 & 0.06 \\ 0.85 & 0.03 & 0.45 & 0.67 \end{bmatrix}, S_3 \sim \begin{bmatrix} 0.59 & 0.76 & 0.01 & 0.99 \\ 0.39 & 0.06 & 0.03 & 0.99 \\ 0.03 & 0.67 & 0.45 & 0.85 \end{bmatrix},$$

$$S_4 \sim \begin{bmatrix} 0.01 & 0.59 & 0.76 & 0.99 \\ 0.06 & 0.39 & 0.03 & 0.99 \\ 0.45 & 0.67 & 0.03 & 0.85 \end{bmatrix}, S_5 \sim \begin{bmatrix} 0.76 & 0.99 & 0.59 & 0.01 \\ 0.06 & 0.99 & 0.03 & 0.39 \\ 0.45 & 0.67 & 0.03 & 0.85 \end{bmatrix}, S_6 \sim \begin{bmatrix} 0.59 & 0.01 & 0.76 & 0.99 \\ 0.99 & 0.39 & 0.06 & 0.03 \\ 0.85 & 0.45 & 0.67 & 0.03 \end{bmatrix},$$

$$S_7 \sim \begin{bmatrix} 0.59 & 0.01 & 0.99 & 0.76 \\ 0.06 & 0.99 & 0.03 & 0.39 \\ 0.45 & 0.85 & 0.67 & 0.03 \end{bmatrix}, S_8 \sim \begin{bmatrix} 0.59 & 0.01 & 0.99 & 0.76 \\ 0.06 & 0.03 & 0.39 & 0.99 \\ 0.67 & 0.85 & 0.45 & 0.03 \end{bmatrix}, S_9 \sim \begin{bmatrix} 0.76 & 0.01 & 0.59 & 0.99 \\ 0.03 & 0.39 & 0.99 & 0.06 \\ 0.67 & 0.85 & 0.45 & 0.03 \end{bmatrix},$$

$$S_{10} \sim \begin{bmatrix} 0.76 & 0.01 & 0.99 & 0.59 \\ 0.03 & 0.99 & 0.39 & 0.06 \\ 0.85 & 0.03 & 0.45 & 0.67 \end{bmatrix}$$

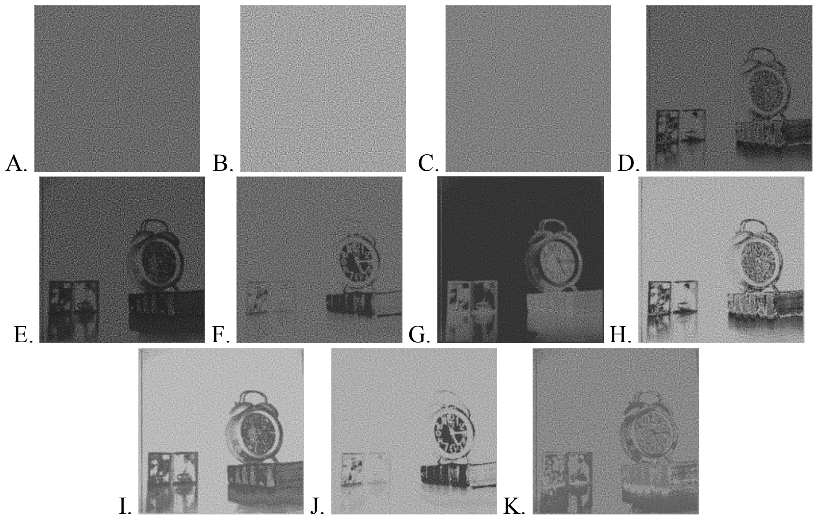


Figure 4: A-C: Share 1 to 3. D-G: Fuzzy-OR stacks of share subsets (1,2), (1,3), (2,3), (1,2,3). H-K: Fuzzy-XOR stacks of the same share subsets.

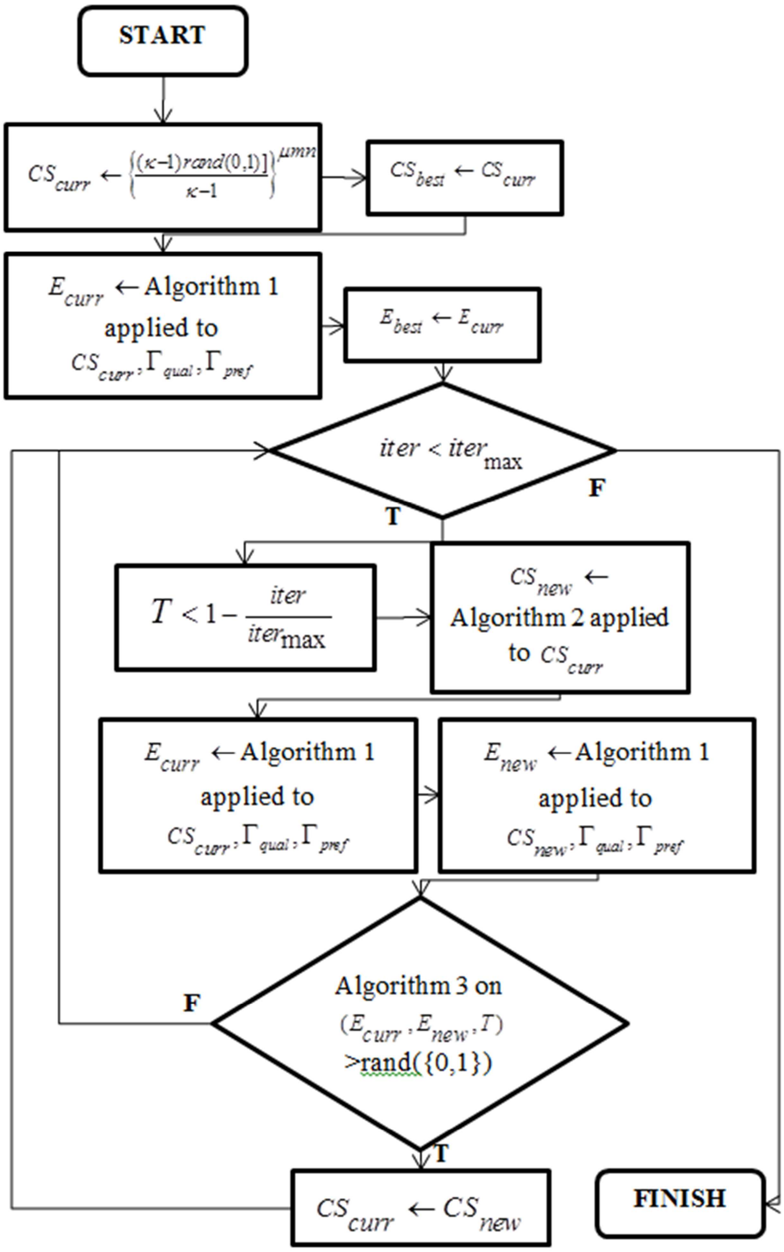

START
$CS_{curr} \leftarrow \left\{\frac{[(\kappa-1)rand(0,1)]}{\kappa-1}\right\}^{\mu mn}$
$CS_{best} \leftarrow CS_{curr}$
$E_{curr} \leftarrow$ Algorithm 1 applied to $CS_{curr}, \Gamma_{qual}, \Gamma_{pref}$
$E_{best} \leftarrow E_{curr}$
$iter < iter_{\max}$
T
F
$T < 1 - \frac{iter}{iter_{\max}}$
$CS_{new} \leftarrow$ Algorithm 2 applied to CS_{curr}
$E_{curr} \leftarrow$ Algorithm 1 applied to $CS_{curr}, \Gamma_{qual}, \Gamma_{pref}$
$E_{new} \leftarrow$ Algorithm 1 applied to $CS_{new}, \Gamma_{qual}, \Gamma_{pref}$
Algorithm 3 on (E_{curr}, E_{new}, T) $>$rand($\{0,1\}$)
F
T
$CS_{curr} \leftarrow CS_{new}$
FINISH

Figure 5: Procedure the SA construction of RVC schemes.

It is clear here that, as in binary VC, *XOR* is superior to *OR* in decoding, as it eliminates redundant dark pixels. Note the reordering of colours resulting in loss of quality particulary in F and J above, whereas D and H are stacked subsets of shares in Γ_{pref} .

5 Results and Discussion

5.1 Parameter Constraints

Parameters μ and κ can be independently set. Indeed, any number of matrices and any number of shadow pixel values can be selected, ensuring:

$$H(S_i^X) \neq H(S_{i+1}^X), \text{ where } i = \{1, \ldots, \eta - 1\}, X \in \Gamma_{qual}. \tag{9}$$

However, it turns out that not all combinations of μ , κ and m yield valid schemes. Obviously, it is preferable to have pixel expansion m as small as possible, resulting in smaller share sizes, but there exists a lower bound. For example, a (2, 2)-RVCS with $\mu = 5, \kappa = 2, m = 4$ has five basis matrices, which are stacked to produce five four-digit binary vectors. Each vector must have a different Hamming weight to satisfy equation (9), but the only possible Hamming weights here are 1, 2 ,3 or 4, so a valid scheme is impossible. (Zero is clearly not a valid Hamming weight, as it implies a matrix comprising only zeros.)

A valid scheme therefore clearly requires that $m \geq \mu$, but ensuring security through equation (8) places further restrictions on the possible combinations and minimal *m*. Experiments with the above parameters indicate that, in this case, $m \geq 8$. An example is the (2, 2, 5, 2)-RVCS basis,

$$S_1 \sim \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}, S_2 \sim \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \end{bmatrix},$$

$$S_3 \sim \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}, S_4 \sim \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix},$$

$$S_5 \sim \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

According to equation (3), this yields relative contrast $\alpha \approx 0.267$. Interestingly and ironically, $m = 10$ produced the better quality $\alpha = \frac{1}{3}$.

However, if, κ increases to to 3 or 4, schemes exist with $m \geq 4$ and $m \geq 3$, respectively. An example of the latter is,

$$S_1 \sim \begin{bmatrix} 0.67 & 1 & 0 \\ 0.33 & 1 & 0 \end{bmatrix}, S_2 \sim \begin{bmatrix} 0.67 & 1 & 0 \\ 0 & 1 & 0.33 \end{bmatrix}, S_3 \sim \begin{bmatrix} 0 & 0.67 & 1 \\ 0.33 & 0 & 1 \end{bmatrix},$$

$$S_4 \sim \begin{bmatrix} 0 & 1 & 0.67 \\ 0.33 & 0 & 1 \end{bmatrix}, S_5 \sim \begin{bmatrix} 0.67 & 0 & 1 \\ 0 & 1 & 0.33 \end{bmatrix}$$

with $\alpha \approx 0.296$ according to equation (3).

5.2 RVCS Encryptions and Decryptions

Here, the basis matrices and resulting shadow images and stacks are demonstrated for two full colour images. The first is a (2, 2, 10, 10)-RVCS with *m=4*. The stochastic algorithm produced the following basis with $\alpha \approx 0.272$ under *OR*-based stacking.

$$S_1 \sim \begin{bmatrix} 0 & 0.89 & 1 & 0.33 \\ 0 & 0.89 & 0.56 & 0.11 \end{bmatrix}, S_2 \sim \begin{bmatrix} 0 & 1 & 0.89 & 0.33 \\ 0 & 0.89 & 0.11 & 0.56 \end{bmatrix}, S_3 \sim \begin{bmatrix} 0.33 & 0 & 0.89 & 1 \\ 0.56 & 0.11 & 0 & 0.89 \end{bmatrix},$$
$$S_4 \sim \begin{bmatrix} 0 & 1 & 0.89 & 0.33 \\ 0.11 & 0 & 0.89 & 0.56 \end{bmatrix}, S_5 \sim \begin{bmatrix} 1 & 0.33 & 0 & 0.89 \\ 0.11 & 0 & 0.56 & 0.89 \end{bmatrix}, S_6 \sim \begin{bmatrix} 1 & 0.33 & 0.89 & 0 \\ 0.56 & 0.89 & 0 & 0.11 \end{bmatrix},$$
$$S_7 \sim \begin{bmatrix} 0 & 0.89 & 0.33 & 1 \\ 0.89 & 0.11 & 0 & 0.56 \end{bmatrix}, S_8 \sim \begin{bmatrix} 0 & 1 & 0.33 & 0.89 \\ 0.89 & 0 & 0.11 & 0.56 \end{bmatrix}, S_9 \sim \begin{bmatrix} 0.89 & 0 & 0.33 & 1 \\ 0.11 & 0.56 & 0.89 & 0 \end{bmatrix},$$
$$S_{10} \sim \begin{bmatrix} 0.89 & 0.33 & 0 & 1 \\ 0 & 0.56 & 0.89 & 0.11 \end{bmatrix}$$

Figure 6 shows the resulting shares and stacks using fuzzy *OR* and *XOR*, after encoding the “earth” image using the above basis.

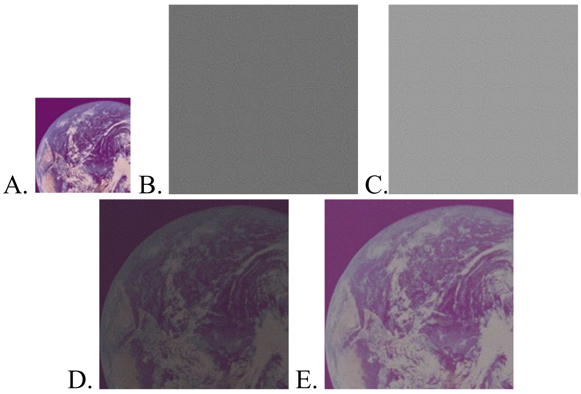


Figure 6: (2, 2, 10, 10)-RVCS, $m=4, \alpha \approx 0.272$. *A: Secret image, B:* H_1*, C:* H_2*, D:* $\mathrm{H}_1 \otimes^{fuz} \mathrm{H}_2$*, E:* $\mathrm{H}_1 \oplus^{fuz}_{LS} \mathrm{H}_2$

Here, $\Gamma_{pref} = \Gamma_{qual}$, since there exists only one stack from two shares. Figure 6D is the result of *OR*-stacking the shares, but with the luxury of computation to decrypt, least sensitive fuzzy *XOR* can be used to reveal the reconstruction in Figure 6E.

Figure 7 shows the encryption of a second full colour test image into a (2, 4, 20, 20)-RVCS, along with fuzzy-*OR* decryptions of minimal qualified subsets. The basis used is,

$$S_1 \sim \begin{bmatrix} 0.95 & 0.16 & 1 & 0.11 \\ 0.79 & 0.63 & 0.84 & 0.05 \\ 0.47 & 0.79 & 0 & 0.89 \\ 0.95 & 0.21 & 1 & 0.21 \end{bmatrix}, S_2 \sim \begin{bmatrix} 0.16 & 1 & 0.95 & 0.11 \\ 0.05 & 0.79 & 0.84 & 0.63 \\ 0 & 0.47 & 0.89 & 0.79 \\ 0.21 & 0.95 & 1 & 0.21 \end{bmatrix}, S_3 \sim \begin{bmatrix} 1 & 0.95 & 0.11 & 0.16 \\ 0.63 & 0.84 & 0.05 & 0.79 \\ 0.47 & 0.79 & 0 & 0.89 \\ 0.95 & 0.21 & 1 & 0.21 \end{bmatrix},$$

$$S_4 \sim \begin{bmatrix} 0.95 & 0.16 & 1 & 0.11 \\ 0.63 & 0.84 & 0.79 & 0.05 \\ 0.47 & 0.79 & 0.89 & 0 \\ 1 & 0.21 & 0.21 & 0.95 \end{bmatrix}, S_5 \sim \begin{bmatrix} 0.95 & 0.11 & 1 & 0.16 \\ 0.63 & 0.05 & 0.79 & 0.84 \\ 0.79 & 0.89 & 0.47 & 0 \\ 0.21 & 0.21 & 0.95 & 1 \end{bmatrix}, S_6 \sim \begin{bmatrix} 0.95 & 0.16 & 0.11 & 1 \\ 0.84 & 0.05 & 0.79 & 0.63 \\ 0 & 0.89 & 0.79 & 0.47 \\ 0.95 & 1 & 0.21 & 0.21 \end{bmatrix},$$

$$S_7 \sim \begin{bmatrix} 1 & 0.95 & 0.11 & 0.16 \\ 0.63 & 0.84 & 0.79 & 0.05 \\ 0.79 & 0.89 & 0 & 0.47 \\ 1 & 0.95 & 0.21 & 0.21 \end{bmatrix}, S_8 \sim \begin{bmatrix} 0.16 & 0.11 & 0.95 & 1 \\ 0.05 & 0.79 & 0.84 & 0.63 \\ 0.89 & 0 & 0.47 & 0.79 \\ 0.21 & 0.21 & 0.95 & 1 \end{bmatrix}, S_9 \sim \begin{bmatrix} 1 & 0.11 & 0.95 & 0.16 \\ 0.63 & 0.05 & 0.79 & 0.84 \\ 0 & 0.47 & 0.79 & 0.89 \\ 0.21 & 0.95 & 0.21 & 1 \end{bmatrix},$$

$$S_{10} \sim \begin{bmatrix} 1 & 0.16 & 0.95 & 0.11 \\ 0.63 & 0.05 & 0.79 & 0.84 \\ 0.47 & 0.89 & 0.79 & 0 \\ 1 & 0.95 & 0.21 & 0.21 \end{bmatrix}, S_{11} \sim \begin{bmatrix} 0.11 & 1 & 0.16 & 0.95 \\ 0.63 & 0.05 & 0.79 & 0.84 \\ 0.89 & 0 & 0.47 & 0.79 \\ 1 & 0.95 & 0.21 & 0.21 \end{bmatrix}, S_{12} \sim \begin{bmatrix} 0.16 & 0.11 & 0.95 & 1 \\ 0.63 & 0.79 & 0.84 & 0.05 \\ 0.79 & 0 & 0.89 & 0.47 \\ 0.21 & 0.21 & 0.95 & 1 \end{bmatrix},$$

$$S_{13} \sim \begin{bmatrix} 0.95 & 0.16 & 0.11 & 1 \\ 0.05 & 0.63 & 0.84 & 0.79 \\ 0.79 & 0.47 & 0.89 & 0 \\ 1 & 0.21 & 0.21 & 0.95 \end{bmatrix}, S_{14} \sim \begin{bmatrix} 0.16 & 0.95 & 1 & 0.11 \\ 0.63 & 0.05 & 0.79 & 0.84 \\ 0.47 & 0.79 & 0 & 0.89 \\ 1 & 0.95 & 0.21 & 0.21 \end{bmatrix}, S_{15} \sim \begin{bmatrix} 0.16 & 0.95 & 1 & 0.11 \\ 0.63 & 0.79 & 0.05 & 0.84 \\ 0 & 0.89 & 0.47 & 0.79 \\ 1 & 0.95 & 0.21 & 0.21 \end{bmatrix},$$

$$S_{16} \sim \begin{bmatrix} 0.95 & 0.16 & 0.11 & 1 \\ 0.79 & 0.63 & 0.84 & 0.05 \\ 0.79 & 0.47 & 0.89 & 0 \\ 0.21 & 0.95 & 0.21 & 1 \end{bmatrix}, S_{17} \sim \begin{bmatrix} 1 & 0.16 & 0.11 & 0.95 \\ 0.63 & 0.84 & 0.79 & 0.05 \\ 0.89 & 0.47 & 0.79 & 0 \\ 0.21 & 0.21 & 0.95 & 1 \end{bmatrix}, S_{18} \sim \begin{bmatrix} 0.95 & 0.11 & 0.16 & 1 \\ 0.63 & 0.84 & 0.79 & 0.05 \\ 0 & 0.47 & 0.89 & 0.79 \\ 0.95 & 0.21 & 1 & 0.21 \end{bmatrix},$$

$$S_{19} \sim \begin{bmatrix} 0.16 & 0.95 & 0.11 & 1 \\ 0.79 & 0.63 & 0.84 & 0.05 \\ 0.47 & 0 & 0.89 & 0.79 \\ 0.21 & 1 & 0.95 & 0.21 \end{bmatrix}, S_{20} \sim \begin{bmatrix} 0.16 & 0.11 & 1 & 0.95 \\ 0.79 & 0.84 & 0.05 & 0.63 \\ 0.89 & 0.47 & 0 & 0.79 \\ 1 & 0.21 & 0.95 & 0.21 \end{bmatrix}$$

.

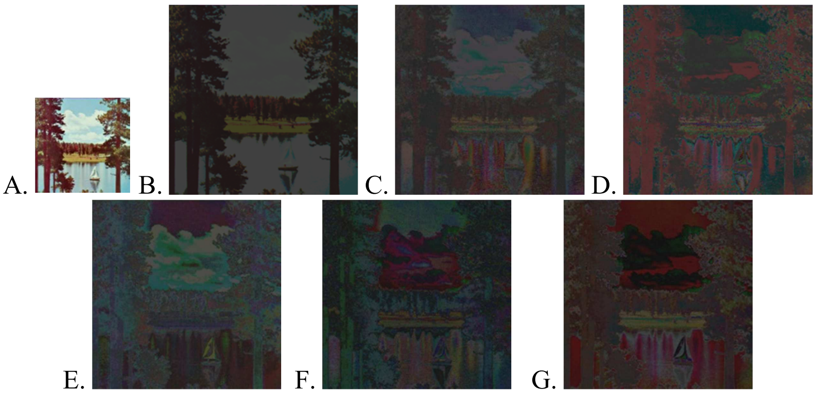


Figure 7: (2, 4, 20, 20)-RVCS, $m = 4, |\alpha| \approx 0.14$. *A: Secret image, B-G: All fuzzy-XOR 2-stacks*

Figure 8 gives the equivalent fuzzy-*XOR* stacks.

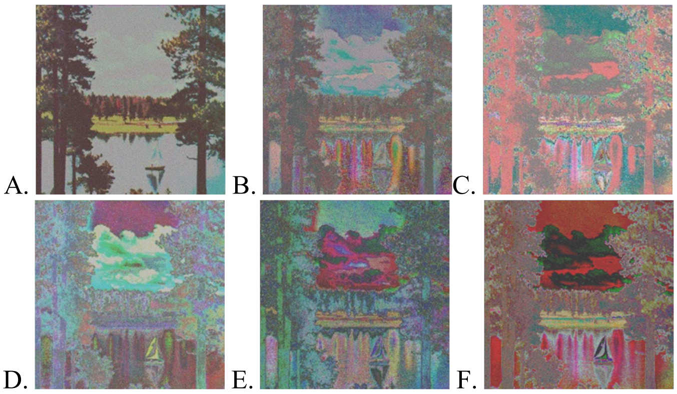


Figure 8: (2, 4, 20, 20)-RVCS, A-F: All possible 2-stacks combined using fuzzy-XOR

In this case, $\Gamma_{pref} = \{1,2\}$, rendering the correct colour order for $H_1\{\otimes,\oplus\}^{fuz}H_2$ but the aforementioned problem of colour reordering is blatant in the above example.

So far, only graph access structure basis matrices have been obtained. In our paper, [Buckley, 13], genetic algorithms were used to evolve binary basis matrices and limitations were conceded due to combinatorial complexity. The problem is exacerbated for real-valued matrices, as the possible number of combinations vastly increases.

5.3 Comparative Analysis

The findings of this study are compared to other key studies in terms of non-binary encryption capability (i.e. pixel values and basis matrix count), ability to decrypt using both *OR* and *XOR*, pre-processing, access structures, perfect information theoretic security and computational cost.

Stacking results (of shares 1 and 2) are compared with those of conventional VC. Results are compared based on peak-to-signal noise ratio (PSNR) calculations, beginning with mean squared error, which is used to derive PSNR, i.e.,

$$MSE_{channel} = \frac{1}{wh}\sum_{x=1}^{w}\sum_{y=1}^{h}(I_{xy} - I'_{xy})^2 \text{ , } PSNR = \frac{\sum_{channel=1}^{3} 20\log_{10}255 - 10\log_{10}MSE_{channel}}{3} \qquad (10)$$

where w and h are the image dimensions, I_{xy} is the respective colour value (on a respective channel) for the decoded image, and likewise I'_{xy} for the decoded image.

Peak-to-signal noise ratio, however, is only a "flat" measure based on average distortion across the altered image. As described in [Wang, 04], human vision is highly adapted to extracting structural information from an image. To take advantage of this, they proposed a Structural Simiarlarity (SSIM) index. Here, a map of similarity metrics is calculated on square-sized blocks $x \subset \mathrm{I}, y \subset \mathrm{I}'$. A block size of 10 is used in this analysis. The map is calculated as follows, and the map average to arrive at the final index value.

$$SSIM_{map} = \frac{\left(2\overline{x}\overline{y} + c_1\right)\left(2\operatorname{cov}(x,y) + c_2\right)}{\left(\left(\overline{x}\right)^2 + \left(\overline{y}\right)^2 + \operatorname{std}(x) + \operatorname{std}(y) + c_2\right)}, SSIM = 100.\overline{SSIM_{map}} \tag{11}$$

where $c_1 = (0.01.2^7)^2, c_2 = (0.03.2^7)^2$ and overbar denotes mean.

	Unrestricted non-binary values	**No. of basis matrices**	***OR* and *XOR* decryption**	**Halftoning not required**	**Threshold access**	**General access**
[Naor, 94]	No	2	Yes	No	Yes	No
[Naor, 96]	No	2	No	No	No	No
[Ateniese, 96:2]	No	2	Yes	No	Yes	Yes
[Verheul, 97]	No	*c*	Yes	No	Yes	No
[Lin, 04]	Yes	2	No	Yes	Yes	Yes
[Blundo, 01]	No	Up to 5	Yes	No	Yes	Yes
[Liu, 10]	No	2	Yes	No	Yes	No
[Wu, 13]	No	n/a	No	No	Yes	No
[Christy, 15]	No	n/a	Yes	No	Yes	No
[Sugawari, 15]	Yes	n/a	n/a	n/a	Yes	Yes
RVC	**Yes**	**2 or more**	**Yes**	**Yes**	**Yes**	**Yes**

Table 2: Comparative analysis 1

	Perfect inf. sec. security	**Comp. Cost based on $\binom{n}{k}$ or $\min(\lvert\Gamma_{qual}\rvert)$**	***XOR* Decoded PSNR**	***XOR* Decoded SSIM**
[Naor, 94]	Yes	$O(n^2)$	4-9	0-13%
[Naor, 96]	Yes	$O(n^2)$	5-10	0-16%
[Ateniese, 96:2]	Yes	$O(n^2)$	4-9	0-13%
[Verheul, 97]	Yes	$O(n^2)$	6-11	0-13%
[Lin, 04]	Yes	$O(n^2)$	n/a	n/a
[Blundo, 01]	Yes	$O(n^2)$	<10	10-20%
[Liu, 10]	Yes	$O(n^2)$	<12	5-20%
[Wu, 13]	Yes	$O(n^2)$	n/a	n/a
[Christy, 15]	No	n/a (stochastic)	n/a	n/a
[Sugawari, 15]	Yes	Unknown	n/a (only simple images)	n/a (only simple images)
RVC	**Yes**	$O(n^2)$ **per SA iteration**	**9-13**	**5-50%**

Table 3: Comparative analysis 2

The *OR* and *XOR* stacking results of binary and real-valued VC are compared for four grayscale images (A-D) of differing entropy values and colour distributions, as well as three full-colour images (E-G). The test images are given in Table 4, where entropy is given by,

$$E = \frac{-100 \sum_{0 \le i \le 255, \xi_i > 0} \xi_i \log_2 \xi_i}{8}, \tag{12}$$

where ξ_i is the number of pixels of the ith grayscale colour.

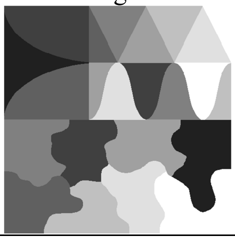
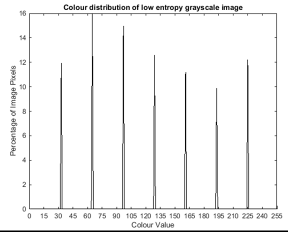
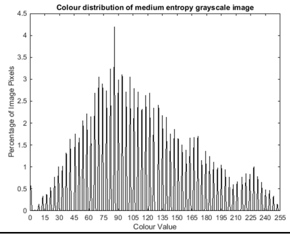
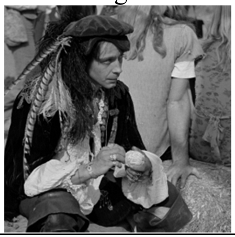
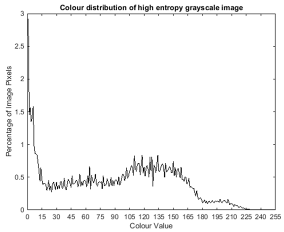
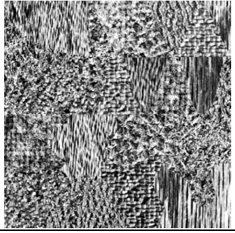
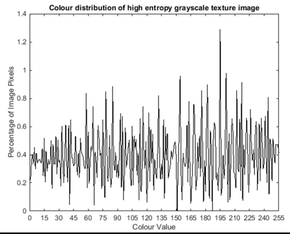
Test Image	**Colour Distribution**	**Percentage Entropy**
Image A 	 	37.3
Image B 	 	71.3
Image C 	 	94
Image D 	 	97.5

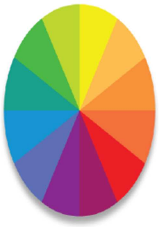
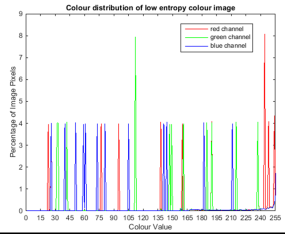
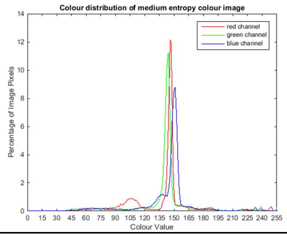
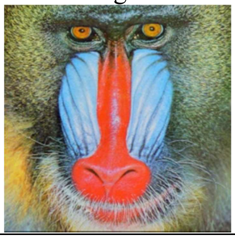
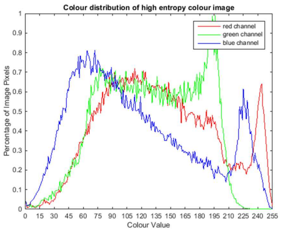
Image E 	 	50.8
Image F 	 	74.6
Image G 	 	97.1

Table 4: Test images and colour analyses

Each image was encrypted into a pair of binary, as well as real-valued shadow images using a (2, 2, 10, 30)-RVC with $m = 4$. Table 5 compares the resulting binary versus fuzzy decryptions on, respectively, the binary and real-valued shares for both binary operations.

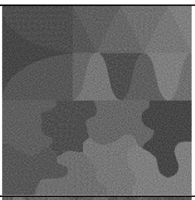
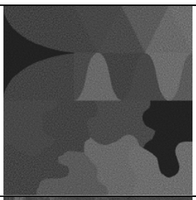
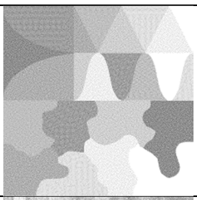
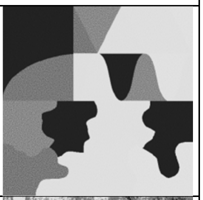
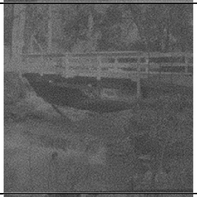
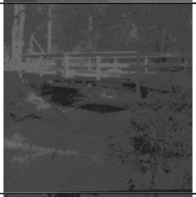
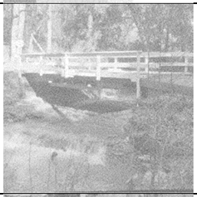
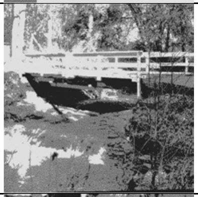
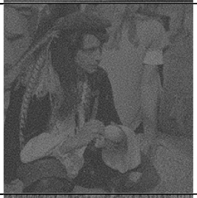
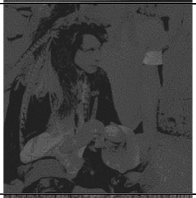
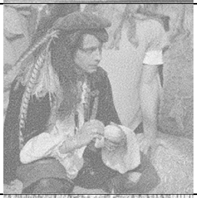
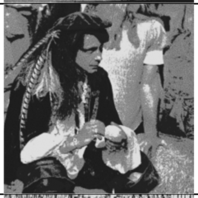
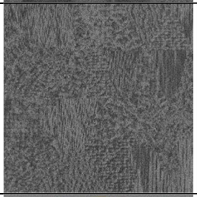
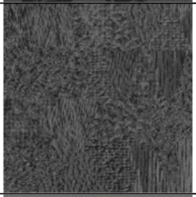
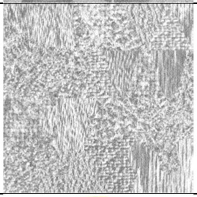
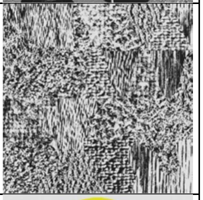
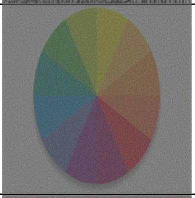
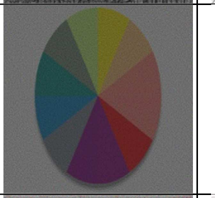
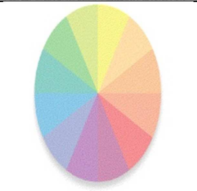
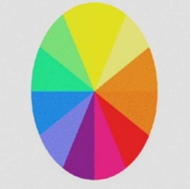
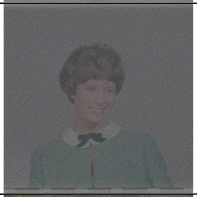
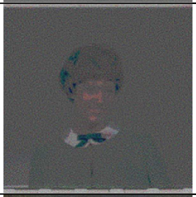
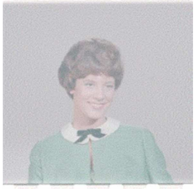
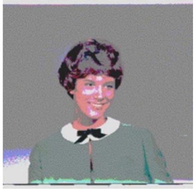
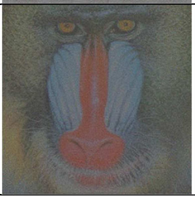
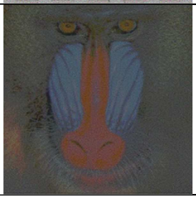
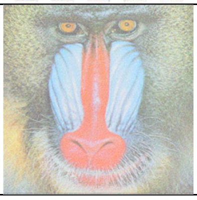
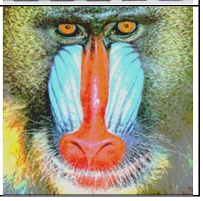
Img	VC Binary-OR Stacks	RVC Fuzzy-OR Stacks	VC Binary-XOR Stacks	RVC Fuzzy-XOR Stacks
A				
B				
C				
D				
E				
F				
G				

Table 5: Binary vs Real-valued VC decrypted images using the OR operations

In each case, similarity of the decrypted image to the secret image was calculated in terms of PSNR and SSIM, the results of which follow in Table 6, and in Table 7 are the percentage increases in reconstructed image similarity of RVC over conventional VC. The results there clearly incidicate improved quality of reconstructed images using the RVC approach, as opposed to conventional VC. Grayscale images with few colours or a smooth colour value distribution produced the most significantly improved PSNR and SSIM metrics, although the chaotically distributed Image D also showed lesser improvements in real-valued versus binary VC decryption.

	(Peak to Signal Noise Ratio %, Structural Similarity %)			
Image	**binary VC using *OR***	**binary VC using *XOR***	**RVC using *OR***	**RVC using *XOR***
A	(5.2, 1.0)	(6.4, 11.2)	(6.1, 5.1)	(11, 25)
B	(5.9, 3.1)	(5.6, 6.6)	(7.3, 6.1)	(11, 23.3)
C	(6.0, 2.7)	(4.9, 4.0)	(8.7, 8.3)	(11.8, 20.8)
D	(5.4, 13.2)	(6.5, 31.3)	(6.2, 15.2)	(10.4, 49.0)
E	(4.0, 0.7)	(9.1, 45.9)	(4.3, 3.2)	(12.3, 18.3)
F	(5.7, 0.8)	(6.5, 1.5)	(6.4, 1.4)	(9.5, 5.2)
G	(5.7, 4.0)	(6.1, 9.2)	(6.6, 5.4)	(10.4, 22.7)

Table 6: PSNR of binary vs real-valued VC with parameters (2, 2, 10, 255), m=4

Interestingly, there is a broad increase in quality improvement with higher entropy values, with some obvious exceptions. Consider the low entropy colour Image E. Here, there is a huge improvement in *OR*-based stacking, but XOR-based stacking exhibits a decrease in quality versus conventional VC. However, in this case, the above metric is misleading, as it is clear in Table 5 that fuzzy-*XOR* decrypts the white background as an average gray colour, as opposed to binary-*XOR*, which retains the pure white. Since there are so many white pixels in the secret image, this change produces the significant error value.

Images A to G were then encrypted into a (2, 4, 30, 30)-RVC, producing the *OR* and *XOR* reconstructions in Figure 6 for $\Gamma_{pref} = \{1, 2\}$. The overall increases in PSNR and SSIM are summarized in Table 7.

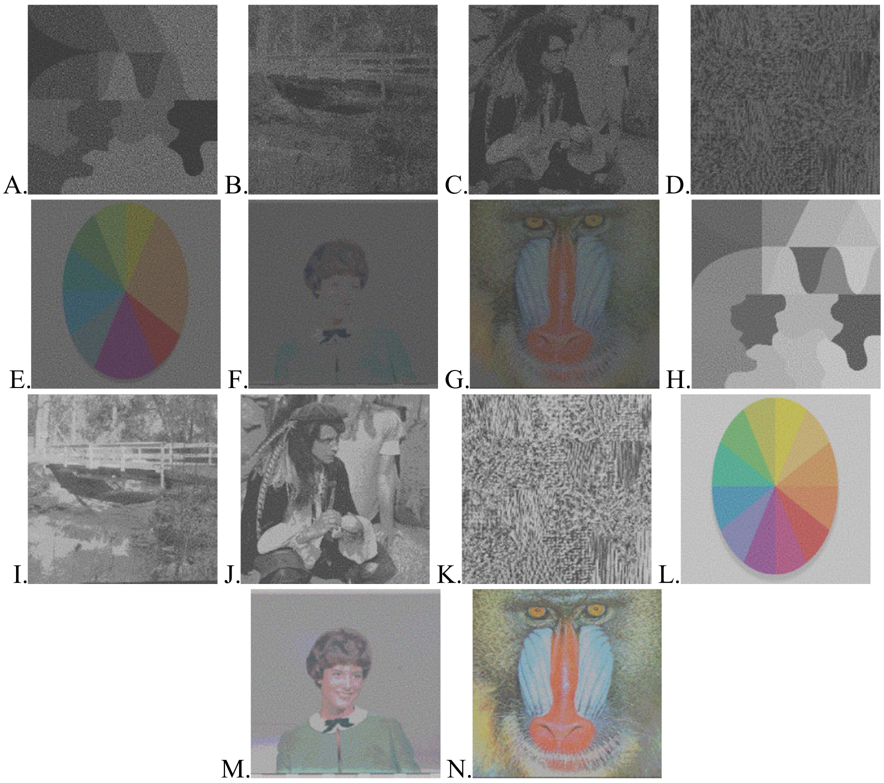


Figure 9: Fuzzy-OR and –XOR reconstructions of Images A-G from a (2, 4, 30, 255)-RVC

	Mean Percentage Improvement in PSNR and SSIM			
Scheme	***OR*** **PSNR**	***XOR*** **PSNR**	***OR*** **SSIM**	***XOR*** **SSIM**
(2, 2, 10, 255)	19	74	180	164
(2, 4, 30, 255)	59	73	52	269

Table 7: Improvements in similarity metrics using binary vs real-valued VC with parameters (2, 2, 10, 255), m=4

The improved decrypted image quality from using real-valued VC is clear here, particularly for computational *XOR* decryption. It is surprising, in fact, that the improvements gained from RVC for this access structure are, on average, 80% higher than for the previous one.

5.4 Security

Information theoretic security of conventional VC relies on the requirement, when calculating a scheme's basis matrices, that $H(S_0[j]) = H(S_1[j]) \forall j = \{1, \ldots, n\}$. Equivalently, $\sum S_0[j] = \sum S_1[j]$, because the sum and Hamming weight of a binary vector are equal. For the construction of the subpixel matrix at coordinate (x, y) in the shadow image held by the *j*th participant, the following procedure is followed:

1. Select basis matrix $S_i[j], i = \mathrm{I}_{xy}$, 2. Randomly permutate $S_i[j]$, 3. Reshape $S_i[j]$ into rectangular matrix *M*, and 4. Place *M* at coordinate (x, y) in H_j .

An adversary possessing a share might want to retrieve the secret from that share. He must therefore retrieve I_{xy} from *M*. However, step 2 clearly renders this impossible due to Hamming weight equality. For example, if $S_0[1] = \begin{bmatrix}1 & 0 & 1 & 0\end{bmatrix}$ and $S_1[1] = \begin{bmatrix}0 & 0 & 1 & 1\end{bmatrix}$, both could be permutated to become $S_0[1] \square S_1[1] \square \begin{bmatrix}0 & 1 & 1 & 0\end{bmatrix} \therefore M = \begin{bmatrix}0 & 1 \\ 1 & 0\end{bmatrix}$. Hence the procedure is irreversible.

A set of RVC matrices is non-binary, meaning Hamming weight is inapplicable. Instead, equality of row sums might be considered preferable. For example, $S_1[1] = \begin{bmatrix}0 & 1 & 0.5 & 0.5\end{bmatrix}, S_2[1] = \begin{bmatrix}1 & 0 & 1 & 0\end{bmatrix}, S_3[1] = \begin{bmatrix}0.5 & 0.5 & 0.5 & 0.5\end{bmatrix}$. However, steps 2 and 3 do not conceal the mapping from $I_{x,y}$ to *M*, because the distribution of colour values in the resulting subpixel matrices differ.

To maintain information theoretic security in graph access structures, it is necessary that, given *j*, all $S_i[j]$ are equal up to permutation, as in equation (8). In non-graph access structures, ie. (*k*, *n*)-VC, $k \geq 3$, one must also ensure that no subset of shares $X \in \Gamma_{forb}$ can access any part of the secret when they are stacked according to equations (4) and (5). Here, it is necessary that $S_1^X \sim S_2^X \sim \ldots \sim S_\kappa^X$, i.e. all (*OR*'d or *XOR*'d) stacks of all subsets are equal up to column permutation.

As noted in Section 4.4, the candidate solution is initialised such that row $i, i = 1, \ldots, n$ in all matrices are equal up to permutation and the only change to the solution comes about using Algorithm 1. Furthermore, this algorithm guarantees to maintain the quality up to permutation, hence information theoretic security in the final constructed RVC scheme is guaranteed for graph access structures. (As discussed, non-graph access is not considered in this paper.)

6 Conclusion and Ongoing Research

There is little prior research into the direct encoding of colours into visual secret sharing schemes, as opposed to preprocessing the secret image to reduce it to binary or using purely computational methods. This is the first study that removes all restrictions from colour values that can be fed directly into a visual cryptographic scheme construction algorithm for schemes permitted decryption using human vision. In this paper, an objective function has been given to stochastically derive valid real-valued basis matrices, in which fractions between 0 and 1 represent grayscale values from 0 to 255. In effect, this is the first true colour VC methodology

No halftoning is required, retaining more of the information from the original grayscale or colour image. This producing a better quality reconstruction evidenced here in the fuzzy *OR* and *XOR* decryptions of various images of differing entropies and colour distributions, in different access structures. Real-valued VC has been compared with conventional binary VC in terms of peak-to-signal noise ratio and structural similarity index, and in the vast majority of cases, there is a significant (indeed, in many cases, a dramatic) improvement in image reconstruction quality. The mean improvement in quality for *OR*-based stacking in the examples in Section 5 (taking both PSNR and SSIM into account) is 86%, and that for computational (*XOR*-based) stacking is 145%!

This proposal is an entirely new type of VSS presententing new computer scientific and mathematical challenges. The problem of colour reordering in RVC has been demonstrated in Section 5. The equivalent problem in conventional VC is production of a negative image, but this is only a problem when using pixel expansion reduced basis matrices. Matrices with maximal expansion, produced using the cumulative array method [Ateniese, 96:2], although extremely space-inefficient, do not suffer from this effect. It is therefore a pressing challenge to devise an equivalent technique for generating (k, n, μ, κ) schemes.

A further unknown is the precise relationship between the bounds on parameters k, n, μ, κ and m. A governing system of formulae would help to calculate the feasibility of a given scheme, without having to infer infeasibility from the lack of successful experimental results. For example, if a high-entropy image needs to be encoded into a scheme with a low pixel expansion, it would be useful to derive theoretical bounds on the number of colour values.

References

[Adhikari, 05] Adhikari, A, Sikdar, A.: A new (2,n)-visual threshold scheme for color images, In Proc. Indocrypt 2003, volume 2904, 148-161

[Arumugam, 12] Arumugam, S., Lakshmanan, R., Nagar, A.K.: On (k, n)*-visual cryptography scheme, Designs, Codes and Cryptography, July 2012, 1-10

[Ateniese, 96:1] Ateniese, G., Blundo, C., De Santis, A.: Constructions and Bounds for Visual Cryptography, Automata, Languages and Programming, volume 1099, 416-428

[Ateniese, 96:2] Ateniese, G., Blundo, C., De Santis, A.: Visual Cryptography for General Access Structure, Information and Computation, 129(2), 86-106

[Blundo, 01] Blundo C., De Bonis A., De Santis, A.,, De Santis, A.,: Improved Schemes for Visual Cryptography, Designs, Codes and Cryptography, 24(3), 255-278

[Buckley, 13] Buckley, N., Nagar, A.K., Arumugam, S.: Evolution of Visual Cryptography Basis Matrices with Binary Chromosomes, In Proc. 8th EUROSIM Conf. on Modelling and Simulation, 7-12

[Chen, 11] Chen, T., Tsao, K.: Threshold visual secret sharing by random grid, The Journal of Systems and Software, 84(7), 1197-1208

[Christy, 15] Christy, J.I., Seenivasagam, V.: Feed Forward Networks in Color Extended Visual Cryptography to Generate Meaningful Shares, Internerational Journal of Security and Its Applications, 9(1), 165-178

[Cimato, 05] Cimato, S., De Prisco, R., De Santis, A.: Optimal colored threshold visual cryptography schemes, Designs, Codes and Cryptography, 35(3), 311-335

[Cimato, 07] Cimato, S., De Prisco, R., De Santis, A.: Colored visual cryptography without color darkening, Theoretical Computer Science, 374(1-3), 261-276

[Duraisamy, 13] Duraisamy, A., Sathiyamoorthy, M., Chandrasekar, S.: Protection of Privacy in Visual Crytography Scheme Using Error Diffusing Technique, Int. Journal of Computer Science and Network, n/a, 60-66

[Floyd, 76] Floyd, R.W., Steinberg, L.: An Adaptive Algorithm for Spatial Greyscale, Proceedings of the Society for Information Display (Second Quarter 1976), 17(2), 75-77

[Hernandez, 11] Hernandez, J.E., Nava, J.: Least Sensitive (most robust) fuzzy “exclusive or” operations, In Proc. 2011 Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS), 1-6

[Hofmeister, 00] Hofmeister, T., Krause, M., Simon, H.U.: Contrast-optimal k out of n secret sharing schemes in visual cryptography, Theoretical Computer Science, 240(2), 471-485

[Hou, 03] Hou, Y.: Visual cryptography for color images, Pattern Recognition, volume 36, 1619-1629

[Kafri, 87] Kafri, O., Keren, E.: Image encryption by multiple random grids, Optical Letters, 12(6), 377-379

[Koga, 98] Koga, H., Yamamoto, H.: Proposal of a lattice-based visual secret sharing scheme for color and gray-scale images, IEICE Transactions on Fundamentals of Electronics, Communication and Computer Sciences, 81-A(6), 1262-1269

[Lin, 04] Lin, C., Tsai, W., Tsai, W.: Secret image sharing with steganography and authentication, The Journal of Systems and Software, 73(3), 405-414

[Liu, 10] Liu, F, Wu, C., Lin, X.: Step Construction of Visual Cryptography Scheme, IEEE Transactions on Information Forensics and Security, 5(1), 27-38

[Lukac, 05] Lukac, R., Plataniotis, K.: Bit-level based secret sharing for image encryption, Pattern Recognition, 38(5), 767-772

[Naor, 94] Naor, M., Shamir A.: Visual Cryptography, In Proc. EUROCRYPT 1994, 1-12

[Naor, 96] Naor, M., Shamir A.: Visual Cryptography II : Improving the Contrast Via the Cover Base, Security in Communication Networks, n/a, 197-202

[Sugawari, 15] Sugawari, S., Harada, K., Sakai, D.: High-chroma visual cryptography using interference colour of high-order retarder films, Optical Review, 22(4), 544-552

[Ulichney, 93] Ulichney, R.: The void-and-cluster method for dither array generation, In Proc. SPIE 1913, Human Vision, Visual Processing and Digital Display, 332-343

[Verheul, 97] Verheul, E.R., van Tilborg, H.C.A.: Constructions and Properties of k out of n visual secret sharing schemes, Designs, Codes and Cryptography, 11(2), 179-196

[Wang, 11] Wang, D., Yi, F., Li, X.: Probabilistic visual secret sharing schemes for grey-scale images and color image, Information Sciences, 181(11), 2189-2208

[Wang, 04] Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image Quality Assessment: From Error Visibility to Structural Similarity, IEEE Transactions on Image Processing, 13(4), 600-612

[Wang, 13] Wang, Z., Pizzolatti, M.S., Chang, C.: Reversible Visual Secret Sharing Based on Random-grids for Two-image Encryption, Int. Journal of Innovative Computing, Information and Control, 9(4), 1691-1701

[Wu, 13] Wu, X., Sun, W.: Improving the visual quality of random grid-based visual secret sharin, Signal Processing, 93(5), 977-995

[Yang, 04] Yang, C.: New visual secret sharing schemes using probabilistic method, Pattern Recognition Letters, 25(4), 481-494

[Zadeh, 65] Zadeh, L.A.: Fuzzy Sets, Information and Control, 8(3), 338-353.