PAPER

Proposal of a Lattice-Based Visual Secret Sharing Scheme for Color and Gray-Scale Images

Hiroki KOGA† *and* **Hirosuke YAMAMOTO**†, *Members*

SUMMARY The visual secret sharing scheme (VSSS) proposed by Naor and Shamir[1] provides a way to encrypt a secret black-white image into shares and decrypt the shares without using any cryptographic computation. This paper proposes an extension of VSSS to sharing of color or gray-scale images. In this paper (k, n) VSSS for images with J different colors is defined as a collection of J disjoint subsets in n-th product of a finite lattice. The subsets can be sequentially constructed as a solution of a certain simultaneous linear equation. In particular, the subsets are simply expressed in $(n, n), (n-1, n)$ and $(2, n)$ cases. Any collections of $k-1$ shares reveal no information on a secret image while stacking of k arbitrary shares reproduces the secret image.

key words: *secret sharing, visual secret sharing, visual cryptography*

1. Introduction

The visual secret sharing scheme (VSSS) proposed by Naor and Shamir[1] unveils a new realization of the (k, n) threshold method[2] for black-white images. In the (k, n) VSSS a black-white image is encrypted into n black-white images called *shares*. If each share is printed on a material such as a transparency, the original black-white image is reproduced only by stacking up arbitrary k shares. This means that no cryptographic computation is required in the decryption of the shares.

In (k, n) VSSS a pair of two sets C_0 and C_1 composed by n-tuples of m *subpixels* plays an important role. White and black pixels of a black-white image are encrypted as elements randomly chosen from C_0 and C_1, respectively. All n shares become m times larger than the original black-white image. Two sets C_0 and C_1 enable parallel decryption of the shares by using a property of human visual system. They cause difference of brightness between black and white pixels when arbitrary k shares are stacked up. They also guarantee that no information on the original image is revealed from any collection of $k-1$ shares. For given k and n Katoh and Imai[3] and Droste[4] attempt to find C_0 and C_1 with small m.

It is quite natural to consider an extension of (k, n) VSSS applicable to color or gray-scale images. In the case of gray-scale images, Naor and Shamir[1] refers to two ideas that realize VSSS. However, it is suspicious that reproduced images are actually recognized as gray-scale images. On the other hand, the visual secret sharing of color images is treated by Naor and Shamir[5]. Though[5] proposes a new scheme based on “cover semi-group,” the scheme only enables an efficient $(2, 2)$ VSSS for color images including two colors.

In this paper a new method to realize (k, n) VSSS for color images is proposed. At least in principle, the method enables to encrypt color images with arbitrary numbers of colors into shares and decrypt arbitrary k shares simply by stacking up them. The method includes a technique to realize (k, n) VSSS for gray-scale images with arbitrary numbers of levels. Given a set of colors, this method guarantees that any collection of $k-1$ shares reveals no information on the color of each pixel as well as the original images themselves.

Mathematically, proposed (k, n) VSSS for color images with J colors is defined as a collection of J subsets in n-th Cartesian product of a finite lattice. Such VSSS is called the *lattice-based VSSS*. In the lattice-based VSSS, not only pixels are treated as elements of the finite lattice but also stacking up two pixels is described as an operation defined on the finite lattice. A certain class of (k, n) VSSS for black-white images can be explained as the lattice-based (k, n) VSSS.

This paper is organized as follows. Section 2 is devoted to the definition of the lattice-based (k, n) VSSS. The definition can be regarded as a modification of the definition given by Naor and Shamir. A simple construction of the lattice-based (n, n) VSSS for $n \geqq 2$ is proposed in Sect. 3. It is shown that in $(2, 2)$ case color images with 2^p colors are encrypted into two shares with 2^{p-1} subpixels if there exists an isomorphism between the set of colors and the Boolean algebra with 2^p elements, where p is an arbitrary positive integer. Section 4 develops a construction of the lattice-based (k, n) VSSS. The construction is valid for arbitrarily given $n \geqq 2$ and $2 \leqq k \leqq n-1$. Remarks on the lattice-based VSSS are given in Sect. 5.

2. Definition of the Lattice-Based (k, n) VSSS

Let L be an arbitrary finite lattice. From the definition of the finite lattice, for arbitrary elements $a, b \in L$ both the least upper bound and the greatest lower bound of $\{a, b\}$ belong to L. The least upper bound and the greatest lower bound of $\{a, b\}$ are denoted by $a \cup b$ and

Manuscript received June 13, 1997.

Manuscript revised January 16, 1998.

†The authors are with the Graduate School of Engineering, The University of Tokyo, Tokyo, 113–8656 Japan.

$a \cap b$, respectively. It is well-known that the idempotent law, the commutative law, the associative law and the absorption law hold with respect to $\cup$ and $\cap$.

Figure 1 shows the Hasse diagram of the binary lattice. The lattice is denoted by L_{bin}. It has only two elements. The least upper bound and the greatest lower bound of two elements are defined as follows: $0 \cup 0 = 0, 1 \cup 0 = 0 \cup 1 = 1 \cup 1 = 1, 0 \cap 0 = 1 \cap 0 = 0 \cap 1 = 0$ and $1 \cap 1 = 1$. Note that $\cup$ means the "or" of the two elements.

Another finite lattice L_{col} is given in Fig. 2. It is well-known that the eight colors, white (0), cyan (C), magenta (M), yellow (Y), red (R), green (G), blue (B) and black (1), have the finite lattice structure given in Fig. 2. Here, we use a convention to denote the greatest element and the least element by 1 and 0, respectively. Mixture of arbitrary two colors means finding the least upper bound of the two colors. For example, the mixtures of two colors are computed as follows: $0 \cup 0 = 0, 0 \cup C = C, C \cup C = C, C \cup Y = G, G \cup 1 = 1$ and $1 \cup 1 = 1$. With respect to the mixtures, it is convenient to interpret 1 as black that turns all colors into black and 0 as white that does not affect the other colors. The mixtures of more than two colors are easily defined since the operator $\cup$ satisfies the commutative law and the associative law.

It is important to note that the m-th Cartesian product of an arbitrary finite lattice L is also a finite lattice. For any $(a_1, a_2, \ldots, a_m)$ and $(b_1, b_2, \ldots, b_m) \in L^m$, two operators $\cup_{L^m}$ and $\cap_{L^m}$ are induced in L^m as follows:

$$\begin{aligned}(a_1, a_2, \ldots, a_m) \cup_{L^m} (b_1, b_2, \ldots, b_m) \\ = (a_1 \cup_L b_1, a_2 \cup_L b_2, \ldots, a_m \cup_L b_m), \\ (a_1, a_2, \ldots, a_m) \cap_{L^m} (b_1, b_2, \ldots, b_m) \\ = (a_1 \cap_L b_1, a_2 \cap_L b_2, \ldots, a_m \cap_L b_m),\end{aligned}$$

where $\cup_L$ and $\cap_L$ denote the operators defined in L. When there is little confusion, both $\cup_L$ and $\cup_{L^m}$ are denoted by $\cup$.

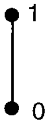


Fig. 1 Hasse diagram of L_{bin}.

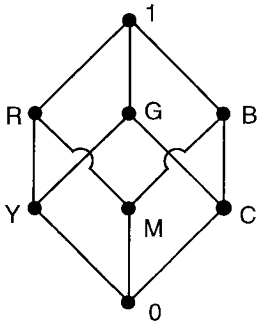


Fig. 2 Hasse diagram of L_{col}.

Now, we are ready to define the lattice-based (k, n) VSSS.

Definition 1: Let $m > 0$ be given. Denote by L a finite lattice of a finite number of colors that can be physically realized. Suppose that $\mathcal{C} = \{c_1, c_2, \ldots, c_J\}$ is a subset of elements in L, which is not necessarily a sublattice of L. For all q satisfying $1 \leqq q \leqq k$ and distinct $\{i_1, i_2, \ldots, i_q\} \subseteqq \{1, 2, \ldots, n\}$ define a mapping $h^{(i_1, i_2, \ldots, i_q)} : (L^m)^n \to L^m$ by

$$h^{(i_1, i_2, \ldots, i_q)}(x) = x_{i_1} \cup x_{i_2} \cup \cdots \cup x_{i_q}, \tag{1}$$

where $x = (x_1, x_2, \ldots, x_n) \in (L^m)^n$ and for any $a, b \in L^m$ $a \cup b$ means the least upper bound of $\{a, b\}$. If there exists $\{(\mathcal{X}_{c_j}, \mathcal{Y}_{c_j})\}_{j=1}^{J} \subseteqq (L^m)^n \times L^m$ with the following three properties, $\{(\mathcal{X}_{c_j}, \mathcal{Y}_{c_j})\}_{j=1}^{J}$ is called the *lattice-based* (k, n) *VSSS with colors* $\mathcal{C}$.

1. For all $j = 1, 2, \ldots, J$ and distinct $\{i_1, i_2, \ldots, i_k\} \subseteqq \{1, 2, \ldots, n\}$, all $x \in \mathcal{X}_{c_j}$ satisfy
$$h^{(i_1, i_2, \ldots, i_k)}(x) \in \mathcal{Y}_{c_j}.$$
2. For all $q < k$ and $\{i_1, i_2, \ldots, i_q\} \subset \{1, 2, \ldots, n\}$ define $\mathcal{X}_{c_j}^{(i_1, i_2, \ldots, i_q)}$ by
$$\mathcal{X}_{c_j}^{(i_1, i_2, \ldots, i_q)} = \{(x_{i_1}, x_{i_2}, \ldots, x_{i_q}) : (x_1, x_2, \ldots, x_n) \in \mathcal{X}_{c_j}\}. \tag{2}$$
Then, $\mathcal{X}_{c_j}^{(i_1, i_2, \ldots, i_q)}, j = 1, 2, \ldots, J$ are indistinguishable in the sense that they contain the same elements with the same frequencies.
3. For all $c_j \in \mathcal{C}$ satisfying $c_j \neq 1 \in L$, all the elements in $\mathcal{Y}_{c_j}$ are composed by 1s and at least one c_j. In case that $c_j = 1$, $\mathcal{Y}_{c_j}$ has only one element composed by m 1s. □

For example, in case that $m = 2$ and $L = L_{\text{bin}}$, the lattice-based (2, 2) VSSS with colors $\mathcal{C} = \{0, 1\}$ is constructed as follows:

$$\mathcal{X}_0 = \left\{ \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \right\}, \mathcal{Y}_0 = \{01, 10\}, \tag{3}$$

$$\mathcal{X}_1 = \left\{ \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right\}, \mathcal{Y}_1 = \{11\}, \tag{4}$$

where we use the notation that an element in $(L^m)^n$ is expressed in the form of $n \times m$ matrix when L is the m-th Cartesian product of another finite lattice such as L_{bin} or L_{col}. Then, $h^{(i_1, i_2, \ldots, i_q)}(x)$ defined in (1) means to compute the least upper bound of the i_1-th, i_2-th, ..., and i_q-th rows of the matrix describing x. In the example above, elements in $\mathcal{Y}_0$ can be recognized as white since they contain one 0.

The lattice-based VSSS includes a certain class of VSSS for black-white images if it is defined over a Cartesian product of $L_{\rm bin}$. In (2,2) VSSS for black-white images white pixels and black pixels are encrypted as elements of C_0 and C_1, respectively, where C_0 and C_1 are sets of 2×2 matrices that are obtained by permuting all the columns of the following S_0 and S_1:

$$S_0=\begin{bmatrix}1&0\\1&0\end{bmatrix},\quad S_1=\begin{bmatrix}0&1\\1&0\end{bmatrix}$$

[1]. It is clear that $C_0=\mathcal{X}_0$ and $C_1=\mathcal{X}_1$, where $\mathcal{X}_0$ and $\mathcal{X}_1$ are the sets defined in (3) and (4), respectively. On the other hand, $(2,3)$ VSSS for black-white images cannot be treated in the framework of the lattice-based VSSS. In $(2,3)$ case S_0 and S_1 are written as

$$S_0=\begin{bmatrix}1&0&0\\1&0&0\\1&0&0\end{bmatrix},\quad S_1=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix},$$

respectively [1], which may lead to $\mathcal{Y}_0=\{100,010,001\}$ and $\mathcal{Y}_1=\{110,101,011\}$. This $\mathcal{Y}_1$ violates the definitions of the lattice-based VSSS. To be treated as the lattice-based $(2,3)$ VSSS $\mathcal{Y}_1$ should be equal to $\{111\}$.

Notice that there is a practical but an essential reason why the finite lattice is introduced in the framework of VSSS. Let S be a set with an internal operation. Suppose that all subpixels take values in S and the internal operation describes stacking up two subpixels. It is natural to require the operation to satisfy the commutative law and the associative law. These two laws enable (k,n) VSSS to decrypt k shares by stacking up them in an arbitrary order. In addition, almost all elements in S cannot have their inverses with respect to the internal operation. Permitting the existence of inverses for all $s\in S$ leads to pathologic VSSS. For example, stacking a black subpixel with another subpixel can yield a white or transparent subpixel. Finite lattice is one of the simplest algebraic structures that meets these requirements.

3. Simple Construction of (n,n) VSSS

In order to construct the lattice-based VSSS with colors $\mathcal{C}=\{c_1,c_2,\ldots,c_J\}$, it is necessary to choose a finite lattice L, $m>0$ and $\{(\mathcal{X}_{c_j},\mathcal{Y}_{c_j})\}_{j=1}^{J}\in(L^m)^n\times L^m$ adequately. Simple realizations of the lattice-based (n,n) VSSS are given in this section.

3.1 Simple $(2,2)$ VSSS

Denote by $L_{\sf YCG}$ the sublattice of $L_{\rm col}$ composed by $0,\sf Y,C,G$ and 1. In case that $L=L_{\sf YCG}$ and $m=4$, the lattice-based $(2,2)$ VSSS with colors $\mathcal{C}=\{\sf Y,C,G\}$ can be constructed. Elements of $\mathcal{X}_{\sf Y},\mathcal{X}_{\sf C}$ and $\mathcal{X}_{\sf G}$ are expressed in the following form, respectively:

$$\mathcal{X}_{\sf Y}:\begin{bmatrix}\sf Y&0&1&\sf C\\0&\sf Y&\sf C&1\end{bmatrix},\quad \mathcal{X}_{\sf C}:\begin{bmatrix}\sf C&0&1&\sf Y\\0&\sf C&\sf Y&1\end{bmatrix},$$

$$\mathcal{X}_{\sf G}:\begin{bmatrix}\sf Y&\sf C&0&1\\\sf C&\sf Y&1&0\end{bmatrix}.$$

All the elements of $\mathcal{X}_{\sf Y},\mathcal{X}_{\sf C}$ and $\mathcal{X}_{\sf G}$ are obtained by permuting the columns of the corresponding elements. Hereafter, writing $\mathcal{Y}_c, c\in\mathcal{C}$ is omitted since they are easily obtained. Note that each element of $\mathcal{Y}_{\sf Y},\mathcal{Y}_{\sf C}$ and $\mathcal{Y}_{\sf G}$ has two 1s. This means that two out of four subpixels are recognized as a color belonging to $\mathcal{C}$, while the others become black.

In case that $L=L_{\rm col}$ and $m=8$, the lattice-based $(2,2)$ VSSS with colors $\mathcal{C}=\{0,\sf Y,M,C,R,G,B,1\}$ can be constructed by using $0,\sf Y,M,C$ and 1. Eight elements belonging to $\mathcal{X}_0,\mathcal{X}_{\sf Y},\mathcal{X}_{\sf M},\mathcal{X}_{\sf C},\mathcal{X}_{\sf R},\mathcal{X}_{\sf G},\mathcal{X}_{\sf B}$ and $\mathcal{X}_1$ are written as follows, respectively:

$$\mathcal{X}_0:\begin{bmatrix}0&\sf Y&\sf M&\sf C&1&1&1&1\\0&1&1&1&\sf Y&\sf M&\sf C&1\end{bmatrix},$$

$$\mathcal{X}_{\sf Y}:\begin{bmatrix}\sf Y&0&\sf M&\sf C&1&1&1&1\\0&\sf Y&1&1&\sf M&\sf C&1&1\end{bmatrix},$$

$$\mathcal{X}_{\sf M}:\begin{bmatrix}\sf M&0&\sf C&\sf Y&1&1&1&1\\0&\sf M&1&1&\sf C&\sf Y&1&1\end{bmatrix},$$

$$\mathcal{X}_{\sf C}:\begin{bmatrix}\sf C&0&\sf Y&\sf M&1&1&1&1\\0&\sf C&1&1&\sf Y&\sf M&1&1\end{bmatrix},$$

$$\mathcal{X}_{\sf R}:\begin{bmatrix}\sf Y&\sf M&\sf C&0&1&1&1&1\\\sf M&\sf Y&1&1&\sf C&0&1&1\end{bmatrix},$$

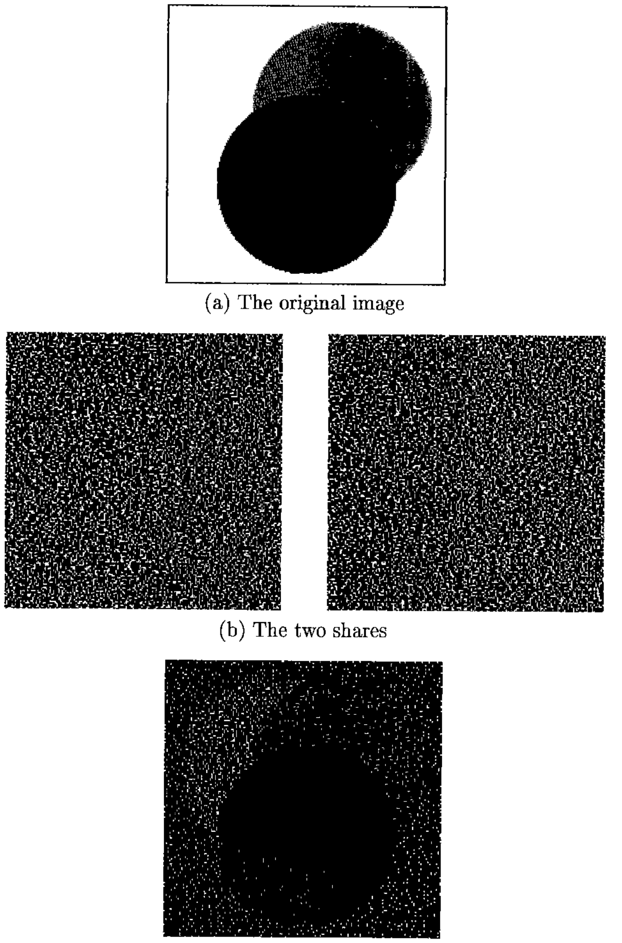
$$\mathcal{X}_{\sf G}:\begin{bmatrix}\sf C&\sf Y&\sf M&0&1&1&1&1\\\sf Y&\sf C&1&1&\sf M&0&1&1\end{bmatrix},$$

$$\mathcal{X}_{\sf B}:\begin{bmatrix}\sf M&\sf C&\sf Y&0&1&1&1&1\\\sf C&\sf M&1&1&\sf Y&0&1&1\end{bmatrix},$$

$$\mathcal{X}_1:\begin{bmatrix}\sf Y&\sf M&\sf C&0&1&1&1&1\\1&1&1&1&\sf Y&\sf M&\sf C&0\end{bmatrix},$$

Figure 3 shows the original image, the two shares and the reproduced image. Since there are $8=4\times 2$ subpixels, either the height or the width of the original image should be enlarged twice before the encryption. Each pixel in the original image is encrypted according to its color $c\in\mathcal{C}$, i.e., it is encrypted into an element randomly chosen from $\mathcal{X}_c$. Though under such encryption the two shares and the reproduced image becomes $16=4\times 4$ times larger than the original image, the four images in Fig. 3 are drawn in the same size. As is understood from Fig. 3, each share does not reveal any information on the original image. Figure 3 shows that the original image, which gives an intuitive view of $L_{\rm col}$, can be recognized from the reproduced image in the presence of black subpixels. The reproduced image in Fig. 3 is obtained by computer simulation. However, we have checked that the original image is actually reproduced fairly well by stacking up the two shares printed on two transparencies.

In case that $L=L_{\rm col}$ and $m=5$, the lattice-based $(2,2)$ VSSS with colors $\mathcal{C}=\{0,\sf Y,M,C,R,G,B,1\}$ are realized in a different way. Eight elements belonging to



(a) The original image

(b) The two shares

(c) The reproduced image

Fig. 3 Encryption and decryption of the lattice-based VSSS with eight colors.

$\mathcal{X}_0, \mathcal{X}_{\mathrm{Y}}, \mathcal{X}_{\mathrm{M}}, \mathcal{X}_{\mathrm{C}}, \mathcal{X}_{\mathrm{R}}, \mathcal{X}_{\mathrm{G}}, \mathcal{X}_{\mathrm{B}}$ and $\mathcal{X}_1$ are written as

$$\mathcal{X}_0 : \begin{bmatrix} 0 & \mathrm{Y} & \mathrm{M} & \mathrm{C} & 1 \\ 0 & \mathrm{B} & \mathrm{G} & \mathrm{R} & 1 \end{bmatrix}, \mathcal{X}_{\mathrm{Y}} : \begin{bmatrix} \mathrm{Y} & \mathrm{M} & \mathrm{C} & 1 & 0 \\ 0 & \mathrm{G} & \mathrm{R} & \mathrm{B} & 1 \end{bmatrix},$$
$$\mathcal{X}_{\mathrm{M}} : \begin{bmatrix} \mathrm{M} & \mathrm{C} & \mathrm{Y} & 1 & 0 \\ 0 & \mathrm{R} & \mathrm{B} & \mathrm{G} & 1 \end{bmatrix}, \mathcal{X}_{\mathrm{C}} : \begin{bmatrix} \mathrm{C} & \mathrm{Y} & \mathrm{M} & 1 & 0 \\ 0 & \mathrm{B} & \mathrm{G} & \mathrm{R} & 1 \end{bmatrix},$$
$$\mathcal{X}_{\mathrm{R}} : \begin{bmatrix} 0 & \mathrm{Y} & \mathrm{M} & \mathrm{C} & 1 \\ \mathrm{R} & \mathrm{B} & \mathrm{G} & 1 & 0 \end{bmatrix}, \mathcal{X}_{\mathrm{G}} : \begin{bmatrix} 0 & \mathrm{C} & \mathrm{Y} & \mathrm{G} & 1 \\ \mathrm{G} & \mathrm{R} & \mathrm{B} & 1 & 0 \end{bmatrix},$$
$$\mathcal{X}_{\mathrm{B}} : \begin{bmatrix} 0 & \mathrm{M} & \mathrm{C} & \mathrm{Y} & 1 \\ \mathrm{B} & \mathrm{G} & \mathrm{R} & 1 & 0 \end{bmatrix}, \mathcal{X}_1 : \begin{bmatrix} 1 & 0 & \mathrm{Y} & \mathrm{M} & \mathrm{C} \\ 0 & 1 & \mathrm{B} & \mathrm{G} & \mathrm{R} \end{bmatrix},$$

respectively. Notice that any permutation of not only the columns but also the rows of the eight elements belong to the corresponding eight sets. This construction is easily extended into the case that there exists an isomorphism between L and a Boolean algebra. Such situation will happen when, for instance, intermediate colors of Y, M and C are treated. If L and a Boolean algebra are isomorphic, it is well-known that there exists an integer $p \geqq 1$ satisfying $L \cong L_{\mathrm{bin}}^{p}$. In fact, the construction above is based on the fact that $L_{\mathrm{col}} \cong L_{\mathrm{bin}}^{3}$. Therefore, the lattice-based (2,2) VSSS with 2^p colors can be constructed in the same manner. Such construction requires $2^{p-1}+1$ subpixels.

3.2 Simple (n, n) VSSS

The lattice-based (n, n) VSSS can be constructed by using (n, n) VSSS for black-white images. Naor and Shamir[1] proposes the following construction of C_0 and C_1. Let $W = \{e_1, e_2, \ldots, e_n\}$ be a finite set. Let $\pi_1, \pi_2, \ldots, \pi_{2^{n-1}}$ and $\sigma_1, \sigma_2, \ldots, \sigma_{2^{n-1}}$ be lists of all subsets of W whose cardinalities are even and odd, respectively. Define two $n \times 2^{n-1}$ matrices S_0 and S_1 whose ij-components are determined by

$$S_0[i,j] = \begin{cases} 1, & \text{if } e_i \in \pi_j, \\ 0, & \text{otherwise,} \end{cases}$$

and

$$S_1[i,j] = \begin{cases} 1, & \text{if } e_i \in \sigma_j, \\ 0, & \text{otherwise,} \end{cases}$$

for all $i \leqq i \leqq n$ and $1 \leqq j \leqq 2^{n-1}$. All matrices obtained by permuting the columns of S_0 and S_1 yield C_0 and C_1, respectively. In case that $n = 3$, S_0 and S_1 can be written as follows:

$$S_0 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \quad S_1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}. \qquad (5)$$

How do we use S_0 and S_1 in the construction of the lattice-based (n, n) VSSS? A property of the two matrices is a key to the construction. We describe the construction via an example in the case of $n = 3$. Define $S_0(\mathrm{x})$ and $S_1(\mathrm{x})$ by

$$S_0(\mathrm{x}) = \begin{bmatrix} \mathrm{x} & \mathrm{x} & 1 & 1 \\ \mathrm{x} & 1 & \mathrm{x} & 1 \\ \mathrm{x} & 1 & 1 & \mathrm{x} \end{bmatrix}, \; S_1(\mathrm{x}) = \begin{bmatrix} 1 & \mathrm{x} & \mathrm{x} \\ \mathrm{x} & 1 & \mathrm{x} \\ \mathrm{x} & \mathrm{x} & 1 \end{bmatrix}.$$

In fact, $S_0(\mathrm{x})$ and $S_1(\mathrm{x})$ are obtained by replacing 0s in S_0 and S_1 with xs and deleting the first column in S_1. Suppose that x is either Y or C and consider each row of $S_0(\mathrm{x})$ and $S_1(\mathrm{x})$ as elements in L_{YCG}^{4} and L_{YCG}^{3}, respectively. It is important to note that the least upper bound of the three rows of $S_0(\mathrm{x})$ contains one x while the least upper bound of the three rows of $S_1(\mathrm{x})$ no longer contains x. Therefore, concatenations of $S_0(\mathrm{Y})$ with $S_1(\mathrm{C})$ and $S_0(\mathrm{C})$ with $S_1(\mathrm{Y})$ lead to the following elements of the lattice-based (3,3) VSSS with colors $\mathcal{C} = \{\mathrm{Y}, \mathrm{C}\}$:

$$\mathcal{X}_{\mathrm{Y}} : \begin{bmatrix} \mathrm{Y} & \mathrm{Y} & 1 & 1 & 1 & \mathrm{C} & \mathrm{C} \\ \mathrm{Y} & 1 & \mathrm{Y} & 1 & \mathrm{C} & 1 & \mathrm{C} \\ \mathrm{Y} & 1 & 1 & \mathrm{Y} & \mathrm{C} & \mathrm{C} & 1 \end{bmatrix},$$
$$\mathcal{X}_{\mathrm{C}} : \begin{bmatrix} \mathrm{C} & \mathrm{C} & 1 & 1 & 1 & \mathrm{Y} & \mathrm{Y} \\ \mathrm{C} & 1 & \mathrm{C} & 1 & \mathrm{Y} & 1 & \mathrm{Y} \\ \mathrm{C} & 1 & 1 & \mathrm{C} & \mathrm{Y} & \mathrm{Y} & 1 \end{bmatrix}.$$

Increasing the number of colors is easy. For example, the lattice-based (3,3) VSSS with colors $\mathcal{C} =$

$\{\mathrm{Y},\mathrm{C},\mathrm{G}\}$ is obtained by introducing $S_0(\mathrm{G})$ and $S_1(\mathrm{G})$. One of elements in $\mathcal{X}_\mathrm{Y}, \mathcal{X}_\mathrm{C}$ and $\mathcal{X}_\mathrm{G}$ can be written as

$$\begin{aligned}
\mathcal{X}_\mathrm{Y} &: S_0(\mathrm{Y}) \odot S_1(\mathrm{C}) \odot S_1(\mathrm{G}),\\
\mathcal{X}_\mathrm{C} &: S_0(\mathrm{C}) \odot S_1(\mathrm{G}) \odot S_1(\mathrm{Y}),\\
\mathcal{X}_\mathrm{G} &: S_0(\mathrm{G}) \odot S_1(\mathrm{Y}) \odot S_1(\mathrm{C}),
\end{aligned}$$

respectively, where $\odot$ denotes the concatenation of the matrices. When $\mathcal{C}$ is a collection of J colors, the number of subpixels m required by the scheme becomes

$$m = \begin{cases} J \cdot 2^{n-1} - 1, & \text{if } n \text{ is even,}\\ J \cdot 2^{n-1} - (J-1), & \text{if } n \text{ is odd.}\end{cases}$$

4. Construction of (k, n) VSSS

In this section a useful method to construct the lattice-based (k,n) VSSS for arbitrarily given n and k is proposed. Two matrices with n rows, which are denoted by $A(\mathrm{x})$ and $D(\mathrm{x})$, play the same roles as $S_0(\mathrm{x})$ and $S_1(\mathrm{x})$, respectively, where $S_0(\mathrm{x})$ and $S_1(\mathrm{x})$ are matrices introduced in Sect. 3.2. In case that $\mathrm{x} = \mathrm{Y}$ or C, while xs and 1s consist of the least upper bound of any k rows of $A(\mathrm{x})$, only 1s consist of the least upper bound of any k rows of $D(\mathrm{x})$. In such construction of the lattice-based (k,n) VSSS with colors $\mathcal{C} = \{\mathrm{Y},\mathrm{C}\}$, one of elements of $\mathcal{X}_\mathrm{Y}$ and $\mathcal{X}_\mathrm{C}$ can be expressed as $A(\mathrm{Y}) \odot D(\mathrm{C})$ and $A(\mathrm{C}) \odot D(\mathrm{Y})$, respectively. The two matrix are designed so that any collection of $k-1$ rows of $A(\mathrm{Y}) \odot D(\mathrm{C})$ or $A(\mathrm{C}) \odot D(\mathrm{Y})$ reveals no information on Y and C.

The construction using $A(\mathrm{x})$ and $D(\mathrm{x})$ includes two advantages. First, the number of colors is easily increased. The lattice-based (k,n) VSSS with colors $\mathcal{C} = \{\mathrm{Y},\mathrm{C},\mathrm{G}\}$ is realized by $A(\mathrm{Y}) \odot D(\mathrm{C}) \odot D(\mathrm{G})$, $A(\mathrm{C}) \odot D(\mathrm{G}) \odot D(\mathrm{Y})$ and $A(\mathrm{G}) \odot D(\mathrm{Y}) \odot D(\mathrm{C})$. Secondly, the construction can be applied to gray-scale images. It does not use $\mathrm{x} \cup \mathrm{x} = \mathrm{x}$ for $\mathrm{x} \in L$ neither 0 nor 1 in decryption of the shares. All the operations required for the decryption are $0 \cup 0 = 0$, $0 \cup \mathrm{x} = \mathrm{x} \cup 0 = \mathrm{x}$, $1 \cup \mathrm{x} = \mathrm{x} \cup 1 = 1$ and $1 \cup 1 = 1$. This indicates that L should not necessarily be a finite lattice but a set with the least and the greatest elements.

Throughout this section, we define L as $L = L_{\mathrm{YCG}}$ and construct the lattice-based (k,n) VSSS with colors $\mathcal{C} = \{\mathrm{Y},\mathrm{C}\}$. Though the construction is mainly explained via examples, it is easily extended to general cases.

4.1 (n,n) VSSS

Suppose that $\mathrm{x} = \mathrm{Y}$ or C. For each $j = 1, 2, \ldots, n$ define $M_{n,n-j}(\mathrm{x})$ as the matrix that is obtained by all the permutations of the column containing one x, $n-j$ 0s and $j-1$ 1s. In case that $n = 4$, $M_{4,3}(\mathrm{x})$, $M_{4,2}(\mathrm{x})$, $M_{4,1}(\mathrm{x})$ and $M_{4,0}(\mathrm{x})$ can be expressed as follows:

$$M_{4,3}(\mathrm{x}) = \begin{bmatrix} \mathrm{x} & 0 & 0 & 0\\ 0 & \mathrm{x} & 0 & 0\\ 0 & 0 & \mathrm{x} & 0\\ 0 & 0 & 0 & \mathrm{x} \end{bmatrix}, \tag{6}$$

$$M_{4,2}(\mathrm{x}) = \begin{bmatrix} \mathrm{x} & 0 & 0 & \mathrm{x} & 0 & 1 & 1 & 0 & \mathrm{x} & 1 & 0 & 0\\ 0 & \mathrm{x} & 0 & 0 & \mathrm{x} & \mathrm{x} & 0 & 1 & 1 & 0 & 1 & 0\\ 0 & 0 & \mathrm{x} & 1 & 1 & 0 & \mathrm{x} & \mathrm{x} & 0 & 0 & 0 & 1\\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \mathrm{x} & \mathrm{x} & \mathrm{x} \end{bmatrix}, \tag{7}$$

$$M_{4,1}(\mathrm{x}) = \begin{bmatrix} \mathrm{x} & 0 & 1 & 1 & 0 & \mathrm{x} & \mathrm{x} & 1 & 1 & 0 & 1 & 1\\ 0 & \mathrm{x} & \mathrm{x} & 0 & 1 & 1 & 1 & \mathrm{x} & 1 & 1 & 0 & 1\\ 1 & 1 & 0 & \mathrm{x} & \mathrm{x} & 0 & 1 & 1 & \mathrm{x} & 1 & 1 & 0\\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & \mathrm{x} & \mathrm{x} & \mathrm{x} \end{bmatrix}, \tag{8}$$

$$M_{4,0}(\mathrm{x}) = \begin{bmatrix} \mathrm{x} & 1 & 1 & 1\\ 1 & \mathrm{x} & 1 & 1\\ 1 & 1 & \mathrm{x} & 1\\ 1 & 1 & 1 & \mathrm{x} \end{bmatrix}. \tag{9}$$

Notice that $M_{4,3}(\mathrm{x})$ is the only one matrix that the least upper bound of the its four rows contains xs.

Consider $A(\mathrm{x})$ and $D(\mathrm{x})$ expressed in the following forms:

$$\begin{aligned}
A(\mathrm{x}) &= M_{4,3}^{[\alpha_1]}(\mathrm{x}) \odot M_{4,1}^{[\alpha_3]}(\mathrm{x}),\\
D(\mathrm{x}) &= M_{4,2}^{[\alpha_2]}(\mathrm{x}) \odot M_{4,0}^{[\alpha_4]}(\mathrm{x}),
\end{aligned}$$

where for matrix S and $\alpha > 0$ $S^{[\alpha]}$ is defined as

$$S^{[\alpha]} = \overbrace{S \odot S \odot \cdots \odot S}^{\alpha \text{ times}}$$

and $\alpha_1, \alpha_2, \alpha_3$ and α_4 are positive integers specified afterwards. Define $M_{4,2}^{(1,2,3)}(\mathrm{x})$ and $M_{4,2}^{\langle\langle 1,2,3\rangle\rangle}(\mathrm{x})$ as follows:

$$M_{4,2}^{(1,2,3)}(\mathrm{x}) = \begin{bmatrix} \mathrm{x} & 0 & 0 & \mathrm{x} & 0 & 1 & 1 & 0 & \mathrm{x} & 1 & 0 & 0\\ 0 & \mathrm{x} & 0 & 0 & \mathrm{x} & \mathrm{x} & 0 & 1 & 1 & 0 & 1 & 0\\ 0 & 0 & \mathrm{x} & 1 & 1 & 0 & \mathrm{x} & \mathrm{x} & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$M_{4,2}^{\langle\langle 1,2,3\rangle\rangle}(\mathrm{x}) = \begin{bmatrix} \mathrm{x} & 0 & 0 & \mathrm{x} & 0 & 1 & 1 & 0 & \mathrm{x}\\ 0 & \mathrm{x} & 0 & 0 & \mathrm{x} & \mathrm{x} & 0 & 1 & 1\\ 0 & 0 & \mathrm{x} & 1 & 1 & 0 & \mathrm{x} & \mathrm{x} & 0 \end{bmatrix}.$$

Here, $M_{4,2}^{(1,2,3)}(\mathrm{x})$ are obtained by removing the fourth row of $M_{4,2}(\mathrm{x})$. Removing all the columns of $M_{4,2}^{(1,2,3)}(\mathrm{x})$ containing no x yields $M_{4,2}^{\langle\langle 1,2,3\rangle\rangle}(\mathrm{x})$. For $j = 1, 3, 4$ define $M_{4,4-j}^{\langle\langle 1,2,3\rangle\rangle}(\mathrm{x})$ in the same manner. It is easy to verify the following four equalities:

$$M_{4,3}^{\langle\langle 1,2,3\rangle\rangle}(\mathrm{x}) \doteq M_{3,2}(\mathrm{x}), \tag{10}$$

$$M_{4,2}^{\langle\langle 1,2,3\rangle\rangle}(\mathrm{x}) \doteq M_{3,2}(\mathrm{x}) \odot M_{3,1}(\mathrm{x}), \tag{11}$$

$$M_{4,1}^{\langle\langle 1,2,3\rangle\rangle}(\mathrm{x}) \doteq M_{3,1}(\mathrm{x}) \odot M_{3,0}(\mathrm{x}), \tag{12}$$

$$M_{4,0}^{\langle\langle 1,2,3\rangle\rangle}(\mathrm{x}) \doteq M_{3,0}(\mathrm{x}), \tag{13}$$

where for two matrices P and Q $P \doteq Q$ means that $P = Q$ actually holds by permuting the columns of

Q adequately. Define $A^{(1,2,3)}(\mathsf{x}), A^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x}), D^{(1,2,3)}(\mathsf{x})$ and $D^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x})$ in the same way.

Now, set $\alpha_1 = 1$ and determine α_2, α_3 and α_4 by solving $A^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x}) = D^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x})$, which can be expressed as

$$\begin{aligned} &M_{4,3}^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x}) \odot M_{4,1}^{\langle\!\langle 1,2,3\rangle\!\rangle[\alpha_3]}(\mathsf{x}) \\ &\qquad \doteq M_{4,2}^{\langle\!\langle 1,2,3\rangle\!\rangle[\alpha_2]}(\mathsf{x}) \odot M_{4,0}^{\langle\!\langle 1,2,3\rangle\!\rangle[\alpha_4]}(\mathsf{x}) \end{aligned} \quad (14)$$

or

$$\begin{aligned} &M_{3,2}(\mathsf{x}) \odot M_{3,1}^{[\alpha_3]}(\mathsf{x}) \odot M_{3,0}^{[\alpha_3]}(\mathsf{x}) \\ &\qquad \doteq M_{3,2}^{[\alpha_2]}(\mathsf{x}) \odot M_{3,1}^{[\alpha_2]}(\mathsf{x}) \odot M_{3,0}^{[\alpha_4]}(\mathsf{x}), \end{aligned} \quad (15)$$

where (10)–(13) are used for obtaining (15). Since there is no column that is contained in any two of $M_{3,2}(\mathsf{x}), M_{3,1}(\mathsf{x})$ and $M_{3,0}(\mathsf{x})$, (15) implies that $\alpha_2 = \alpha_3 = \alpha_4 = 1$. Therefore, $A(\mathsf{x})$ and $D(\mathsf{x})$ can be expressed as

$$\begin{aligned} A(\mathsf{x}) &= M_{4,3}(\mathsf{x}) \odot M_{4,1}(\mathsf{x}), \\ D(\mathsf{x}) &= M_{4,2}(\mathsf{x}) \odot M_{4,0}(\mathsf{x}). \end{aligned}$$

These matrices lead to the lattice-based (4, 4) VSSS with colors $\mathcal{C} = \{\mathsf{Y}, \mathsf{C}\}$. All the permutations of columns of $A(\mathsf{Y}) \odot D(\mathsf{C})$ and $A(\mathsf{C}) \odot D(\mathsf{Y})$ consist of $\mathcal{X}_{\mathsf{Y}}$ and $\mathcal{X}_{\mathsf{C}}$, respectively.

Surprisingly, $\alpha_1, \alpha_2, \alpha_3$ and α_4 chosen in this way yield

$$\mathcal{X}_{\mathsf{Y}}^{(i_1,i_2,i_3)} = \mathcal{X}_{\mathsf{C}}^{(i_1,i_2,i_3)} \quad (16)$$

for any $\{i_1, i_2, i_3\} \subset \{1, 2, 3, 4\}$, where $\mathcal{X}_{\mathsf{Y}}^{(i_1,i_2,i_3)}$ and $\mathcal{X}_{\mathsf{C}}^{(i_1,i_2,i_3)}$ are the sets defined in (2). In order to verify (16), we first notice that $\alpha_1, \alpha_2, \alpha_3$ and α_4 satisfying $A^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x}) \doteq D^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x})$ guarantee $\mathcal{X}_{\mathsf{Y}}^{(1,2,3)} = \mathcal{X}_{\mathsf{C}}^{(1,2,3)}$. Since some of columns of $A^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x})$ and $D^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x})$ are removed from $A^{(1,2,3)}(\mathsf{x})$ and $D^{(1,2,3)}(\mathsf{x})$, respectively, $A^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x}) \doteq D^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x})$ does not mean $A^{(1,2,3)}(\mathsf{x}) \doteq D^{(1,2,3)}(\mathsf{x})$. However, $A^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x}) \doteq D^{\langle\!\langle 1,2,3\rangle\!\rangle}(\mathsf{x})$ means $A^{(1,2,3)}(\mathsf{Y}) \odot D^{(1,2,3)}(\mathsf{C}) \doteq A^{(1,2,3)}(\mathsf{C}) \odot D^{(1,2,3)}(\mathsf{Y})$, which gives rise to $\mathcal{X}_{\mathsf{Y}}^{(1,2,3)} = \mathcal{X}_{\mathsf{C}}^{(1,2,3)}$. From the symmetry on the rows of $M_{4,j}(\mathsf{x}), j = 0, 1, 2, 3$, it is obvious that $\mathcal{X}_{\mathsf{Y}}^{(1,2,3)} = \mathcal{X}_{\mathsf{C}}^{(1,2,3)}$ implies $\mathcal{X}_{\mathsf{Y}}^{(i_1,i_2,i_3)} = \mathcal{X}_{\mathsf{C}}^{(i_1,i_2,i_3)}$ for any distinct $\{i_1, i_2, i_3\} \subset \{1, 2, 3, 4\}$. In the case that $\{i_1, i_2, i_3\}$ is not distinct, (16) is obvious from the distinct case.

Extension of this construction to (n, n) case is easy. Two matrices $A(\mathsf{x})$ and $D(\mathsf{x})$ can be written as

$$\begin{aligned} A(\mathsf{x}) &= M_{n,n-1}(\mathsf{x}) \odot M_{n,n-3}(\mathsf{x}) \odot \cdots \odot M_{n,0}(\mathsf{x}), \\ D(\mathsf{x}) &= M_{n,n-2}(\mathsf{x}) \odot M_{n,n-4}(\mathsf{x}) \odot \cdots \odot M_{n,1}(\mathsf{x}), \end{aligned}$$

in case that n is odd, and

$$\begin{aligned} A(\mathsf{x}) &= M_{n,n-1}(\mathsf{x}) \odot M_{n,n-3}(\mathsf{x}) \odot \cdots \odot M_{n,1}(\mathsf{x}), \\ D(\mathsf{x}) &= M_{n,n-2}(\mathsf{x}) \odot M_{n,n-4}(\mathsf{x}) \odot \cdots \odot M_{n,0}(\mathsf{x}), \end{aligned}$$

in case that n is even. It is easy to check that $A(\mathsf{x}) \odot D(\mathsf{x})$ has the following two properties: (i) the number of columns is $n \times 2^{n-1}$, and (ii) the least upper bound of n rows contains n xs. Note that in the calculation of the least upper bound $\mathsf{x} \cup \mathsf{x} = \mathsf{x}$ is not used, where $\mathsf{x} = \mathsf{Y}$ or C. This property leads to a practical advantage if it is compared with the construction given in Sect. 3.2. Practically, $\mathsf{C} \cup \mathsf{C}$ and $\mathsf{Y} \cup \mathsf{Y}$ may seem different from C and Y, respectively, i.e., they seem darker than Y and C themselves. This construction keeps the reproduced images from getting dark.

4.2 $(n-1, n)$ VSSS

Suppose that $n = 4$. Let $M_{4,2}(\mathsf{x}), M_{4,1}(\mathsf{x})$ and $M_{4,0}(\mathsf{x})$ be matrices defined in (7)–(9). For each $j = 0, 1, 2$ define $M_{4,j}^{(1,2)}(\mathsf{x})$ be the matrix obtained by removing the third and the fourth rows from $M_{4,j}(\mathsf{x})$. Let $M_{4,j}^{\langle\!\langle 1,2\rangle\!\rangle}(\mathsf{x})$ be the matrix obtained by removing all the columns of $M_{4,j}^{(1,2)}(\mathsf{x})$ including no x. It is easy to verify the following three equalities:

$$\begin{aligned} M_{4,2}^{\langle\!\langle 1,2\rangle\!\rangle}(\mathsf{x}) &\doteq M_{2,1}^{[2]}(\mathsf{x}) \odot M_{2,0}(\mathsf{x}), \\ M_{4,1}^{\langle\!\langle 1,2\rangle\!\rangle}(\mathsf{x}) &\doteq M_{2,1}(\mathsf{x}) \odot M_{2,0}^{[2]}(\mathsf{x}), \\ M_{4,0}^{\langle\!\langle 1,2\rangle\!\rangle}(\mathsf{x}) &\doteq M_{2,0}(\mathsf{x}). \end{aligned}$$

Define $A(\mathsf{x})$ and $D(\mathsf{x})$ as

$$\begin{aligned} A(\mathsf{x}) &= M_{4,2}^{[\alpha_1]}(\mathsf{x}) \odot M_{4,0}^{[\alpha_3]}(\mathsf{x}), \\ D(\mathsf{x}) &= M_{4,1}^{[\alpha_2]}(\mathsf{x}), \end{aligned}$$

where α_1, α_2 and α_3 are positive integers specified afterwards. Define $A^{(1,2)}(\mathsf{x}), A^{\langle\!\langle 1,2\rangle\!\rangle}(\mathsf{x})$, $D^{(1,2)}(\mathsf{x})$ and $D^{\langle\!\langle 1,2\rangle\!\rangle}(\mathsf{x})$ in the same way.

Now, set $\alpha_1 = 1$ and choose α_2 and α_3 satisfying

$$M_{2,1}^{[2]}(\mathsf{x}) \odot M_{2,0}^{[1+\alpha_3]}(\mathsf{x}) \doteq M_{2,1}^{[\alpha_2]}(\mathsf{x}) \odot M_{2,0}^{[2\alpha_2]}(\mathsf{x}), \quad (17)$$

which is equivalent to $A^{\langle\!\langle 1,2\rangle\!\rangle}(\mathsf{x}) \doteq D^{\langle\!\langle 1,2\rangle\!\rangle}(\mathsf{x})$. Clearly, $(\alpha_2, \alpha_3) = (2, 3)$ is the solution of (17). Therefore, $A(\mathsf{x})$ and $D(\mathsf{x})$ are expressed as $A(\mathsf{x}) = M_{4,2}(\mathsf{x}) \odot M_{4,0}^{[3]}(\mathsf{x})$ and $D(\mathsf{x}) = M_{4,1}^{[2]}(\mathsf{x})$, respectively.

In $(n-1, n)$ case $A(\mathsf{x})$ and $D(\mathsf{x})$ are written as

$$\begin{aligned} A(\mathsf{x}) &= M_{n,n-2}(\mathsf{x}) \odot M_{n,n-4}^{[3]}(\mathsf{x}) \odot \cdots \odot M_{n,1}^{[n-2]}(\mathsf{x}), \\ D(\mathsf{x}) &= M_{n,n-3}^{[2]}(\mathsf{x}) \odot M_{n,n-5}^{[4]}(\mathsf{x}) \odot \cdots \odot M_{n,0}^{[n-1]}(\mathsf{x}), \end{aligned}$$

in case that n is odd, and

$$\begin{aligned} A(\mathsf{x}) &= M_{n,n-2}(\mathsf{x}) \odot M_{n,n-4}^{[3]}(\mathsf{x}) \odot \cdots \odot M_{n,0}^{[n-1]}(\mathsf{x}), \\ D(\mathsf{x}) &= M_{n,n-3}^{[2]}(\mathsf{x}) \odot M_{n,n-5}^{[4]}(\mathsf{x}) \odot \cdots \odot M_{n,1}^{[n-2]}(\mathsf{x}), \end{aligned}$$

in case that n is even. The number of columns of $A(\mathsf{x}) \odot D(\mathsf{x})$ becomes $n(n-1)2^{n-2}$ while the least upper bound of arbitrary $n-1$ rows contains $(n-1)$ xs.

4.3 $(2,n)$ VSSS

In $(2,n)$ case $A(\mathsf{x})$ and $D(\mathsf{x})$ are also simply expressed. They can be written as

$$\begin{aligned} A(\mathsf{x}) &= M_{n,1}(\mathsf{x}), \\ D(\mathsf{x}) &= M_{n,0}^{[n-1]}(\mathsf{x}), \end{aligned}$$

which are easily obtained in the same manner developed in Sect. 4.1 and Sect. 4.2. The number of columns of $A(\mathsf{x}) \odot D(\mathsf{x})$ is $2n(n-1)$ while the least upper bound of its arbitrary two rows contains two xs.

4.4 (k,n) VSSS

The lattice-based (k,n) VSSS for $3 \leqq k \leqq n-2$ can be constructed by using the same method developed in the preceding subsections. For simplicity, suppose that k is even. Define $A(\mathsf{x})$ and $D(\mathsf{x})$ as follows:

$$\begin{aligned} A(\mathsf{x}) &= M_{n,k-1}^{[\alpha_1]}(\mathsf{x}) \odot M_{n,k-3}^{[\alpha_3]}(\mathsf{x}) \odot \cdots \odot M_{n,1}^{[\alpha_{k-1}]}(\mathsf{x}), \\ D(\mathsf{x}) &= M_{n,k-2}^{[\alpha_2]}(\mathsf{x}) \odot M_{n,k-4}^{[\alpha_4]}(\mathsf{x}) \odot \cdots \odot M_{n,0}^{[\alpha_k]}(\mathsf{x}), \end{aligned}$$

where $\{\alpha_j\}_{j=1}^{k}$ is a sequence of positive integers. Let $\mathcal{X}_{\mathsf{Y}}$ and $\mathcal{X}_{\mathsf{C}}$ be collections of matrices obtained from all the permutations of the columns of $A(\mathsf{Y}) \odot D(\mathsf{C})$ and $A(\mathsf{C}) \odot D(\mathsf{Y})$, respectively. Define $A^{\langle\langle 1,\ldots,k-1\rangle\rangle}(\mathsf{x})$ and $D^{\langle\langle 1,\ldots,k-1\rangle\rangle}(\mathsf{x})$ as before. All should be done is to determine $\{\alpha_j\}_{j=1}^{k}$ satisfying $A^{\langle\langle 1,\ldots,k-1\rangle\rangle}(\mathsf{x}) = D^{\langle\langle 1,\ldots,k-1\rangle\rangle}(\mathsf{x})$. However, it is quite natural to ask if such sequence of positive integers actually exists.

Hereafter, the existence of a sequence of integers $\{\alpha_j\}_{j=1}^{k}$ is proven. Unfortunately, positiveness of the sequence, i.e., $\alpha_j > 0$ for all $j = 1,2,\ldots,k$, cannot be proven. The positiveness, however, is not essentially important. Suppose that $A(\mathsf{x})$ includes $M_{n,k-j}^{[\alpha_j]}(\mathsf{x})$ satisfying $\alpha_j \leqq 0$. In case that $\alpha_j < 0$, $M_{n,k-j}^{[\alpha_j]}(\mathsf{x})$ should be removed from $A(\mathsf{x})$ and concatenated to $D(\mathsf{x})$. In case that $\alpha_j = 0$, $M_{n,k-j}(\mathsf{x})$ need not be concatenated to $A(\mathsf{x})$. The same operation can also be applied to $D(\mathsf{x})$ if it includes $M_{n,k-j}^{[\alpha_j]}(\mathsf{x})$ satisfying $\alpha_j \leqq 0$.

Proof on the existence of a sequence of integers $\{\alpha_j\}_{j=1}^{k}$ is not so difficult. For simplicity, set $\alpha_1 = 1$. From the definition of $M_{n,k-j}(\mathsf{x})$, $M_{n,k-j}^{\langle\langle 1,\ldots,k-1\rangle\rangle}(\mathsf{x})$ can be expressed as follows:

$$\begin{aligned} M_{n,k-j}^{\langle\langle 1,\ldots,k-1\rangle\rangle}(\mathsf{x}) \doteq M_{k-1,k-2}^{[\beta_{j,1}]}(\mathsf{x}) \odot M_{k-1,k-3}^{[\beta_{j,2}]}(\mathsf{x}) \odot \\ \cdots \odot M_{k-1,0}^{[\beta_{j,k-1}]}(\mathsf{x}), \end{aligned} \tag{18}$$

$\beta_{j,i}$, $j = 1,2,\ldots,k$, $i = 1,2,\ldots,k-1$, are integers satisfying

$$\beta_{j,i} = \begin{cases} \begin{pmatrix} n-k+1 \\ i-j+1 \end{pmatrix}, & \text{if } j-1 \leqq i \leqq n-k+j, \\ 0, & \text{otherwise.} \end{cases} \tag{19}$$

Since $\beta_{j,i} = 0$ for all $i < j-1$ and $\beta_{j,j-1} = 1$, α_j, $j = 2,\ldots,k$ is calculated according to

$$\alpha_{2m} = \sum_{i=1}^{m-1} \alpha_{2i+1}\beta_{2i+1,2m-1} - \sum_{i=1}^{m-1} \alpha_{2i}\beta_{2i,2m-1}, \tag{20}$$

$$\alpha_{2m+1} = \sum_{i=1}^{m} \alpha_{2i}\beta_{2i,2m} - \sum_{i=1}^{m-1} \alpha_{2i+1}\beta_{2i+1,2m} \tag{21}$$

for $m = 1,2,\ldots$ until α_k is obtained. Notice that for all $j = 2,\ldots,k$ α_j is sequentially calculated by using $\alpha_1,\ldots,\alpha_{j-1}$. Since no division is included in (20) and (21), $\{\alpha_j\}_{j=1}^{k}$ turns out to be a sequence of integers.

5. Discussions

This section is devoted to comparison between the two kinds of the lattice-based VSSS given in Sect. 3.2 and Sect. 4, respectively. The lattice-based VSSS given in Sect. 4 is intended to keep the reproduced images from getting dark. It does not use the property that $\mathsf{x} \cup \mathsf{x} = \mathsf{x}$ while the one given in Sect. 3.2 uses the property, where x is an arbitrary element in a finite lattice L different from 0 and 1. Though construction of the lattice-based (k,n) VSSS for general k and n is discussed only in Sect. 4, the same can be established in Sect. 3.2. For example, $S_0(\mathsf{x})$ and $S_1(\mathsf{x})$ can be expressed as

$$\begin{aligned} S_0(\mathsf{x}) &= N_{n,n}(\mathsf{x}) \odot N_{n,n-2}(\mathsf{x}) \odot \cdots \odot N_{n,1}(\mathsf{x}), \\ S_1(\mathsf{x}) &= N_{n,n-1}(\mathsf{x}) \odot N_{n,n-3}(\mathsf{x}) \odot \cdots \odot N_{n,0}(\mathsf{x}) \end{aligned}$$

in (n,n) case of odd n and

$$\begin{aligned} S_0(\mathsf{x}) &= N_{n,2}(\mathsf{x}), \\ S_1(\mathsf{x}) &= N_{n,1}^{[n-1]}(\mathsf{x}) \end{aligned}$$

in $(2,n)$ case, where $N_{n,n-j}(\mathsf{x}), j = 0,1,\ldots,n$, are matrices obtained by all the permutations of the column with $n-j$ xs and j 1s. For obtaining $S_0(\mathsf{x})$ and $S_1(\mathsf{x})$ in (k,n) case, we define $S_0(\mathsf{x})$ and $S_1(\mathsf{x})$ as

$$\begin{aligned} S_0(\mathsf{x}) &= N_{n,k}^{[\alpha_1]}(\mathsf{x}) \odot N_{n,k-2}^{[\alpha_3]}(\mathsf{x}) \odot \cdots \odot N_{n,1}^{[\alpha_k]}(\mathsf{x}), \\ S_1(\mathsf{x}) &= N_{n,k-1}^{[\alpha_2]}(\mathsf{x}) \odot N_{n,k-3}^{[\alpha_4]}(\mathsf{x}) \odot \cdots \odot N_{n,0}^{[\alpha_{k+1}]}(\mathsf{x}) \end{aligned}$$

and find a sequence of integers $\{\alpha_j\}_{j=1}^{k+1}$ that meets $S_0^{\langle\langle 1,\ldots,k-1\rangle\rangle}(\mathsf{x}) \doteq S_1^{\langle\langle 1,\ldots,k-1\rangle\rangle}(\mathsf{x})$, where k is assumed to be an odd integer and $S_0^{\langle\langle 1,\ldots,k-1\rangle\rangle}(\mathsf{x})$ and $S_1^{\langle\langle 1,\ldots,k-1\rangle\rangle}(\mathsf{x})$ are defined in the same way as $A^{\langle\langle 1,\ldots,k-1\rangle\rangle}(\mathsf{x})$ in Sect. 4.4. Then, the lattice-based (k,n) VSSS with colors $\mathcal{C} = \{\mathsf{Y},\mathsf{C}\}$ is realized by $S_0(\mathsf{Y}) \odot S_1(\mathsf{C})$ and $S_0(\mathsf{C}) \odot S_1(\mathsf{Y})$.

Generally, the construction of the lattice-based (k,n) VSSS using $A(\mathsf{x})$ and $D(\mathsf{x})$ will make the reproduced images clear compared with the one using $S_0(\mathsf{x})$ and $S_1(\mathsf{x})$, though it requires more subpixels. Quality of a reproduced image is essentially determined by the ratio r of the number of nonblack pixels to the total number of pixels in the reproduced image. In case of the lattice-based (n,n) VSSS with colors $\mathcal{C} = \{\mathsf{Y},\mathsf{C}\}$ constructed by $A(\mathsf{x})$ and $D(\mathsf{x})$, r becomes $1/2^{n-1}$. On the other hand, $r = 1/(2^n - 1)$ if $S_0(\mathsf{x})$ and $S_1(\mathsf{x})$ are used. Even in (k,n) case, $A(\mathsf{x})$ and $D(\mathsf{x})$ make the ratio greater. Practically, r should be less than $\frac{1}{25}$ in order to grasp a concealed image from the reproduced image. If the concealed image is simple, $r \leqq \frac{1}{100}$ is acceptable. However, the reproduced image seems as if it contained only black pixels in case that $r \geqq \frac{1}{400}$.

It is interesting to notice that finding two matrices $S_0(\mathsf{x})$ and $S_1(\mathsf{x})$ that satisfy $S_0(\mathsf{x})^{\langle\langle 1,\dots,k-1\rangle\rangle} \doteq S_1(\mathsf{x})^{\langle\langle 1,\dots,k-1\rangle\rangle}$ and setting $\mathsf{x} = 0$ lead to (k,n) VSSS for black-white images. In $(2,n)$ case such $S_0(\mathsf{x})$ and $S_1(\mathsf{x})$ can be expressed as

$$\begin{aligned} S_0(\mathsf{x}) &= N_{n,2}(\mathsf{x}) \odot N_{n,0}^{[(n-1)(n-2)/2]}(\mathsf{x}), \\ S_1(\mathsf{x}) &= N_{n,1}^{[n-1]}(\mathsf{x}). \end{aligned}$$

Note that both $S_0(\mathsf{x})$ and $S_1(\mathsf{x})$ are $n \times n(n-1)$ matrices. Two sets C_0 and C_1 are collections of all the permutations of the columns of $S_0(0)$ and $S_1(0)$, respectively. However, (k,n) VSSS constructed in this way is not efficient. In fact, the following $n \times n$ matrices $S_0(\mathsf{x})$ and $S_1(\mathsf{x})$ are available in $(2,n)$ case:

$$\begin{aligned} S_0(\mathsf{x}) &= N_{n,n}(\mathsf{x}) \odot N_{n,0}^{[n-1]}(\mathsf{x}), \\ S_1(\mathsf{x}) &= N_{n,1}(\mathsf{x}). \end{aligned}$$

This dissatisfaction arises from the requirement that in the lattice-based (k,n) VSSS the least upper bounds of any k rows of $S_1(\mathsf{x})$ should be composed by all 1s. For constructing efficient (k,n) VSSS for black-white images, algorithms proposed in [3] and [4] should be used.

6. Conclusion

This paper attempts to extend a class of images that the visual secret sharing scheme proposed by Naor and Shamir can be applied to. First, the visual secret sharing scheme is defined as a collection of subsets in a Cartesian product of a finite lattice. Stacking up two pixels is described as computing the least upper bound of the two pixels. Given $n \geqq 2$ and k satisfying $2 \leqq k \leqq n-1$, it is shown that the (k,n) visual secret sharing scheme for color and gray-scale images are realized by concatenating matrices with certain properties.

Acknowledgement

The authors acknowledge to Dr. Taku Katoh who sent them [4]. The authors are grateful to Mr. Yoshito Hirata and Mr. Kazuhiro Kobayashi who helped to make several experiment on the lattice-based (k,n) VSSS.

References

[1] M. Naor and A. Shamir, "Visual cryptography," Advances in Cryptography-EUROCRYPT'94, Perugia, Italy, pp.1–12, May 1994.

[2] A. Shamir, "How to share a secret," Communications of the ACM, vol.22, pp.612–613, 1979.

[3] T. Katoh and H. Imai, "An extended construction method of visual secret sharing scheme," IEICE Trans., vol.J79-A, no.8, pp.1344–1351, Aug. 1996.

[4] S. Droste, "New results on visual cryptography," Advances in Cryptography-CRYPT'96, Santa Barbara, USA, pp.401–415, Aug. 1996.

[5] M. Naor and A. Shamir, "Visual cryptogrphy II: Improving the contrast via the cover base," available from `http://theory.lcs.mit.edu/~tcryptol/1996/96-07.html`.

Hiroki Koga was born in Fukuoka, Japan, on November 2, 1967. He received the B.E., M.E. and Dr.E. degree from University of Tokyo, Japan, in 1990, 1992 and 1995, respectively. He is currently an assistant professor in the Department of Mathematical Engineering and Information Physics, Graduate School of Engineering, University of Tokyo. His research interest includes the Shannon theory, data compression and information security.

Hirosuke Yamamoto was born in Wakayama, Japan, on November 15, 1952. He received the B.E. degree from Shizuoka University, Shizuoka, Japan, in 1975 and the M.E. and Dr.E. degrees from the University of Tokyo, Tokyo, Japan, in 1977 and 1980, respectively, all in electrical engineering. In 1980 he joined Tokushima University, Tokushima, Japan. He was an Associate Professor at Tokushima University from 1983 to 1987, and at University of Electro-Communications, Tokyo, Japan, from 1987 to 1993. Since 1993 he has been an Associate Professor in the Department of Mathematical Engineering and Information Physics, Graduate School of Engineering, University of Tokyo, Japan. In 1989–90, he was a Visiting Scholar at the Information Systems Laboratory, Stanford University. His research interests are in Shannon theory, coding theory, cryptography, and communication theory.