Threshold Visual Cryptography Schemes With Tagged Shares

PEI-LING CHIU[1], AND KAI-HUI LEE [2]

[1]Department of Risk Management and Insurance, Ming Chuan University, Taipei 11103, Taiwan
[2]Department of Computer Science and Information Engineering, Ming Chuan University, Taipei 11103, Taiwan

Corresponding author: Kai-Hui Lee (khlee@mail.mcu.edu.tw)

ABSTRACT The appearance of shares encoded by traditional visual cryptography is not easy to distinguish and meaningless. It means the shares cannot be managed efficiently nor user-friendly. A tagged visual cryptography scheme (TVCS) provides additional tag images for the shares and decodes the tag images by folding up a single tagged share. A TVCS allows users to efficiently manage shares and greatly increases the user-friendliness of the shares. Previous researches have focused on codebook-based and random-grid-based TVCS approaches, but the visual qualities of recovered secret images and tag images can still be improved. Therefore, this study aims to provide better visual quality of recovered images by improving on previous methods. Moreover, the proposed approach supports adjustable visual qualities of recovered secret images and tag images. In this study, we theoretically analyze the performances of the proposed approach and then demonstrate implementation of the approach.

INDEX TERMS Visual cryptography scheme, visual secret sharing scheme, tagged visual cryptography scheme.

I. INTRODUCTION

The threshold visual cryptography scheme, denoted (k, n)-VCS $(2 \leq k \leq n)$, for the sharing of binary images is also known as a threshold visual secret sharing scheme $((k, n)$-VSSS). A secret image is encrypted into n noise-like shares that are printed on transparencies, and the shares are then distributed among n participants. No information about the secret image is revealed in any noise-like share. The encrypted secret image can be accessed by a qualified set that consists of at least k participants, whereas a set with less than k participants will not be able to retrieve the secret information. The qualified set retrieves and visually decrypts a secret image by stacking their transparencies without performing any cryptographic computation and without any knowledge of cryptography. In addition, the visual cryptography scheme (VCS) features one-time-pad encryption so that it can securely protect secret images [1]–[6].

The traditional VCSs protect secret images by noise-like shares; it leads to a management problem. Meaningless shares are difficult to identify, which is one of the main drawbacks

The associate editor coordinating the review of this manuscript and approving it for publication was Ahmed Farouk [ID].

to noise-like shares. It is difficult for participants to recognize each share, when a participant simultaneously holds several noise-like shares for different secret images. In 2001, Ateniese *et al.* pointed out this issue [7] and, to address it, proposed the construction of the extended visual cryptography scheme (EVCS), and added stego-images, or cover images into noise-like shares, to make identifiable meaningful shares. Subsequently, many researchers have focused on approaches to encoding meaningful shares in visual cryptography, which are known as friendly visual cryptography schemes (FVCSs) [8], [9]. Improving the friendliness of share management has become an important issue in VCS research [10]–[19].

One solution to mitigate the share-identification issue is the tagged visual cryptography scheme (TVCS) [20]. Each share of a TVCS is given a tag pattern, but the appearance of the shares still consists of noise-like black and white dots. The tag-pattern image is revealed by folding up a share. Wu and Sun adopted the random-grid technique to construct a threshold TVCS, denoted (k, n)-TVCS. This method avoids pixel expansion and the VCS codebook preparation problem [21].

Both TVCSs and FVCSs solve the problem of meaningful shares of VCSs, each method has different properties and

TABLE 1. Property comparisons between FVCSs and TVCSs.

		FVCSs	TVCSs
1	Appearance of shares	Meaningful shares	Tagged shares by folding up
2	Applications	•Management	•Management •Authentication •Cheating prevention
3	Residual traces interference	One of issues	An open issue

advantages. TABLE 1 summarizes the features comparison of FVCSs and TVCSs in appearance of shares, applications, and residual traces problem. The descriptions are presented as below.

The FVCSs and TVCSs use meaningful shares to address the problem. The share has an identifiable appearance that can greatly reduce the management problem for participants. The appearance of shares of FVCSs has an identifiable image on it, anyone can see the image directly. The TVCSs generate noise-like shares that same as traditional VCSs, but tagged visual cryptography conceals the tag pattern in each share. When the tagged share is folded up, the concealed tag pattern can be revealed and be recognized by naked eyes. Hence, participants can easily manage their shares according to the tag patterns.

From perspective of applications, Chen *et al.* use TVCS to achieve access control in secret sharing [22]. Different qualify set can authorize to decrypt different secret regions, and verified by tagged shares. Furthermore, cheating prevention is another practical issue of VCS, for example, k dishonest participants collude to cheat honest participants with fake shares. To provide cheating prevention, tagged VCS can verify shares by tag patterns [21].

In addition, residual trace problem usually be regarded in FVCSs. When meaningful shares of FVCSs are stacked, the mix cover images will interfere with the stacked images. For example, FIGURE 1 shows the phenomenon of residual traces in a (2, 2)-FVCS. FIGURE 1 (a) and (b) are meaningful shares with "T" and "V" patterns. While superimposing two shares, the secret image with heart pattern is decrypted in FIGURE 1 (c). But mixed "T" and "V" traces still reside and interfere with visual perception of recovered image, which is called the residual traces problem. Hence, FVCSs usually consider residual traces problem in encryption approaches to improve visual quality. On the contrary, at present, there is no clear residual traces problem and discussion in TVCSs researches. As TABLE 1 listed, FVCSs and TVCSs have different advantages and applications.

In addition, observed by research literature, because of the importance of meaningful shares in VCS, the problem of FVCS has attracted a lot of attentions. Now, the cover images used in FVCS researches are more than binary images. Many studies have tried to use more types of images, e.g., halftone images, random dot stereograms, QR code images and so on [14]–[16], which makes the researches more
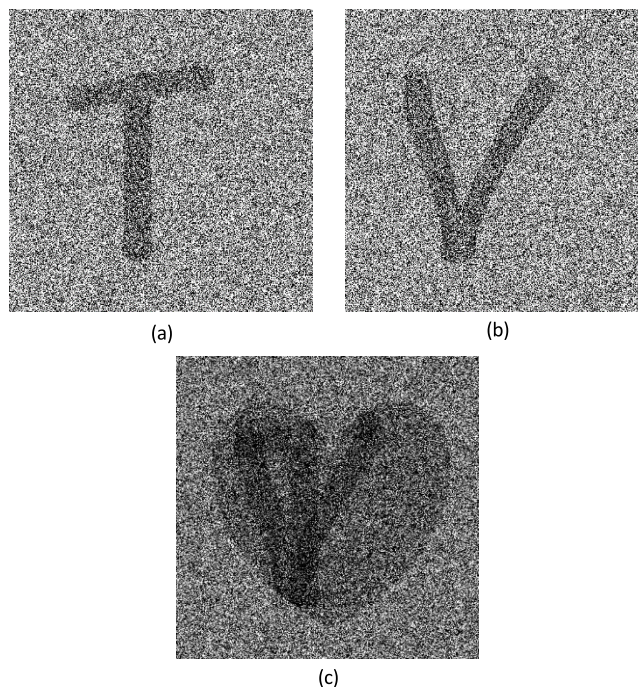

FIGURE 1. The phenomenon of residual traces problem in a (2, 2)-FVCS. (a) Share 1, (b) Share 2, (c) The recovered image.

abundant. In sharing schemes investigation, beside threshold VCSs, recently FVCS studies have explored more diverse sharing schemes, for instance, general access structure with meaningful shares [5], friendly progressive visual cryptography [11], [19] and so on. However, all the above issues are open issues for tag visual cryptography.

Compared with the original VCS, the FVCSs and the TVCSs have lower visual quality of recovered secret images. In the effort to obtain friendly shares, the visual quality of images will be lost. Hence, improving the quality of recovered secret images and tag images are crucial issues in TVCS research.

This study adopts existing optimization probabilistic VCS codebooks [23], [24] as a basis for proposing a systematic encryption method for a (k, n)-TVCS. Our main objectives are to improve the visual qualities of recovered tag images and secret images. Moreover, the proposed approach provides a tuning feature for users. That is, users can adjust visual qualities between recovered secret and tag images according to their requirements.

The remainder of this paper is organized as follows. Section II reviews related works. The proposed (k, n)-TVCS is described in Section III. In Section IV, the performance of the proposed scheme is evaluated by experiments. We conclude this work in Section V.

II. RELATED WORKS
The TVCS encryption procedure usually consists of the steps of sharing a secret image and encrypting the image tags. According to the encryption methods, we divide the

literatures into two categories: codebook-based TVCS and random-grid-based TVCS.

A. CODEBOOK-BASED TVCS

In 2011, Wang and Hsu [20] first proposed a threshold tagged visual cryptography scheme; (k, n)-TVCS. The encoding procedure has two steps: 1) secret image sharing and 2) tag image stamping. In Step 1, one secret image is encrypted with (k, n)-VCS, where the n generated noise-like shares are called base shares. Step 2 involves marking tag patterns into n noise-like shares under security and contrast requirements.

The codebooks for (k, n)-VCS in the first step can adopt any existing basis matrices in traditional (k, n)-VCSs, or any reported collections of column vectors in probabilistic VCSs. Therefore, according to the VCS approach, we call Wang and Hsu's approach codebook-based (k, n)-TVCS.

Suppose each tag image is reconstructed by folding each base share in half in horizontal direction. The size of the tag image is half that of the secret image. Then, the width and height of the tag image are respectively denoted w and h, and the secret image has width $w/2$ and height h. Wang and Hsu's (k, n)-TVCS is briefly presented in ALGORITHM 1.

Algorithm 1 Codebook-Based TVCS Approach [20]

Step 1: Encrypt a secret image to n base shares according to a given (k, n)-VCS codebook. Let P be the average ratios of white pixels in base shares.

Step 2: Stamping tag images into corresponding base shares to generate n tagged shares. For each tag image, Steps 2.1 to 2.3 are executed.
Suppose the stamping tag pixel is denoted as t, as well as the corresponding two symmetrical pixels in base share are represented as b_L and b_R.

   Step 2.1: If color of tag pixel t is black ($t = 1$) and color of b_L and b_R are both white, then one of two pixels (b_L or b_R) is modified as black with equal probability.

   Step 2.2: If color of tag pixel t is white ($t = 0$) and color of b_L and b_R are both white, then randomly select one pixel from b_L and b_R and modified into black with probability, $P/2$.

   Step 2.3: Repeat Steps 2.1 and 2.2, until all tag pixels are stamped.

An advantage of the codebook-based TVCS is that it allows multiple folding-up operations and folding up by different directions and angles. However, it is inevitable that tags will degrade the visual quality of recovered images. Therefore, it is worth studying how to improve visual quality in TVCS approaches.

B. RANDOM-GRID-BASED TVCS

Based on the random-grid (RG) technique, Wu and Sun proposed an encryption method for (k, n)-TVCS [21]. The RG-based (k, n)-TVCS is a two-phase algorithm, where the first phase is interim share generation and the second phase is tagged share modification. The algorithm for RG-based TVCS encoding is briefly described in ALGORITHM 2.

Algorithm 2 Random-Grid-Based TVCS [21]

Input: A binary secret image S with $M \times N$ pixels and n tag images $T_1, \ldots, T_n$ with $M \times (\frac{N}{2})$ pixels.

Output: n tagged shares $R_1, \ldots, R_n$.

Step 1: Construct n interim-shares $\left[\overline{R}_x(i, j), \overline{R}_x(i, N - j + 1)\right]$ from the n tag images by RG-based $(2, 2)$-VCS, $1 \leq i \leq M$, $1 \leq j \leq N/2$, and $1 \leq x \leq n$.

Step 2: Modify the n interim shares $\overline{R}_1, \ldots, \overline{R}_n$ to form n tagged shares by **Steps 3** and **4** when $1 \leq i \leq M$, $1 \leq j \leq N/2$.

Step 3: A number d_1 is randomly selected from $1, \ldots, n$ and $k-1$ numbers $g_1, \ldots, g_{k-1}$ are randomly chosen from $1, \ldots, n$ besides d_1, the value of $R_{d_1}(i, j)$ is modified by

$$R_{d_1}(i, j) = S(i, j) \oplus \overline{R}_{g_1}(i, j) \oplus \ldots \oplus \overline{R}_{g_{k-1}}(i, j).$$

Step 4: Similarly, a number d_2 is randomly selected from $1, \ldots, n$ besides d_1 and $k - 1$ numbers $h_1, \ldots, h_{k-1}$ are randomly chosen. The value of $R_{d_2}(i, N - j + 1)$ is modified by

$$\begin{aligned} R_{d_2}&(i, N - j + 1) \\ &= S(i, N - j + 1) \oplus \overline{R}_{h_1}(i, N - j + 1) \oplus \\ &\quad \ldots \oplus \overline{R}_{h_{k-1}}(i, N - j + 1). \end{aligned}$$

Step5: Output n tagged shares $R_1, \ldots, R_n$.

Assume the size of each tag image is half that of the secret image. The first phase (Step 1) encodes each tag image by $(2, 2)$-VCS constructing methods [25] and generates n interim shares $(\overline{R}_1, \ldots, \overline{R}_n)$ of the same size as the secret image, i.e., double the size of the tag image. Each tag pixel is encrypted as two share pixels by $(2, 2)$-VCS. Then, the two corresponding share pixels are placed at two coordinates symmetrized to the folding line in the interim share.

The second phase (Steps 2, 3, and 4) modifies the pixels in the interim shares according to the secret image such that the generated tagged shares $(R_1, \ldots, R_n)$ satisfy the constraints for the TVCS. The symbol "$\oplus$" represents a XOR-ed operation. For one coordinate, Steps 3 and 4 randomly select one of n pixels in the same coordinate to be modified according to the secret pixel and $k - 1$ share pixels.

Wu and Sun's RG-based (k, n)-TVCS includes several features. The recovered secret images in RG-based (n, n)-TVCS have the optimal contrast values that the same as that of (n, n)-VCS [25]. However, when $k < n$, the contrast can be further improve. The contrast of recovered tag images depends on amount of participant, n. That is, the contrast value of tag images is fixed regardless k value variant in the (k, n)-TVCS.
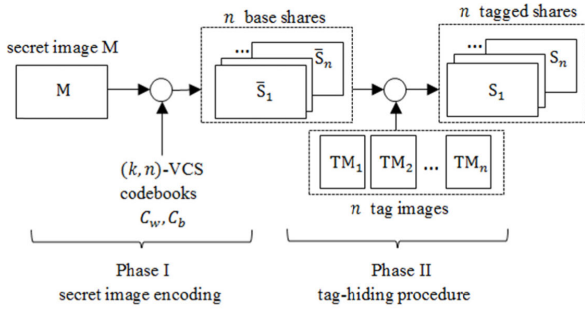
FIGURE 2. Two-phase (k, n)-TVCS encryption procedure.

Moreover, the RG-based TVCS algorithm cannot be applied to (2, 2)-TVCS, which is a key drawback to be addressed.

In summary, the two approaches of TVCSs have different strength. In addition, codebook-based TVCS first encrypts the secret image, and then embeds tag pixels into the shares such that the tag patterns can appear by folding up. On the contrary, random-grid-based TVCS first encrypts each tag images, and then modifies the share pixels to share secret image. Although RG-based method does not require the preparation of codebooks in advance, this study will use traditional codebook-based methods to improve visual quality.

III. THE PROPOSED (k, n)-TVCS

To improve the visual quality of recovered images, we propose a (k, n)-TVCS based on the codebook-based TVCS. Thus, the encryption process includes two phases: Phase I encrypts one secret image, and Phase II hides tag images to n tagged shares, as FIGURE 2 shown. The optimization-based probabilistic (k, n)-VCS codebooks [24] are adopted in Phase I. This section first reviews the probabilistic VCS, then describes the proposed (k, n)-TVCS encryption algorithm and analyzes the constraints and visual qualities of the TVCS.

A. BACKGROUND OF THE PROBABILISTIC VCS

The binary secret image consists of a collection of white and black pixels, in which each secret pixel is encrypted separately. The conventional visual secret sharing schemes (VSSSs) share a binary secret image by 0/1 basis matrices of $n \times m$ size. Each secret pixel is encoded as n blocks to n shares and each block consists of m white and black subpixels. The shares are m times the size of the original secret image. For example, a pair of well-known basis matrices, B_w and B_b, for (2, 2)-VCS are used to encode white and black secret pixels, as below.

$$B_w = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}, \quad \text{and } B_b = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \quad (1)$$

In practice, the codebooks have two collections, C_w and C_b, which include all the matrices obtained by permuting the columns of B_w and B_b, as:

$$C_w = \left\{ \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \right\},$$

$$C_b = \left\{ \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right\}. \quad (2)$$

One white (resp. black) secret pixel is encoded by randomly selecting one matrix in collection C_w (resp. C_b). The typical conventional (2, 2)-VCS codebooks cause two times ($m = 2$) pixel expansion.

In 1999, Ito et al. [26] first proposed a probabilistic model for (k, n)-VCS that addresses the pixel expansion problem. The collections, C_w and C_b, consist of m n-tuple column vectors that are transformed from the existing conventional basis matrices. To share a white (resp. black) secret pixel, one of the column vectors in C_w (resp. C_b) should be randomly selected and the i-th element in the column vector allocated to the i-th share. For example, according to the basis matrices in (1), Ito et al.'s collections C_w and C_b of (2, 2)-VCS should be:

$$C_w = \left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\}, \quad C_b = \left\{ \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right\}. \quad (3)$$

When the secret pixel is black, suppose the column vector $\begin{bmatrix} 1 & 0 \end{bmatrix}^T$ in C_b is randomly selected. Then, pixels 1 and 0 are distributed to shares 1 and 2, respectively.

Each secret pixel is encoded to only one pixel for each corresponding share. Thus, both the image size of the shares and the recovered image remain invariant with the original secret image. Moreover, in 2004, Yang [23] proposed general rules to construct probabilistic (k, n)-VCS and proved that probabilistic VCS satisfies security constraints. Subsequently, Chiu and Lee [24] proposed an optimization model to obtain the codebooks for probabilistic (k, n)-VCS, which determined the chosen probabilities of the collections in codebooks C_w and C_b. Suppose the base code collection is denoted μ_i^n and consists of all n-tuple 0/1 column vectors with Hamming weight i. Furthermore, the chosen probabilities of μ_i^n are f_i^w (resp. f_i^b) in C_w (resp. C_b). Assume set μ_i^n includes n_μ column vectors, each column vector is selected with probability n_μ / f_i^w (resp. n_μ / f_i^b). Then, by the Chiu and Lee's model, the typical probabilistic (2, 2)-VCS is denoted as:

$$C_w = \left\{ \underbrace{\mu_0^2}_{f_0^w = 1/2}, \underbrace{\mu_2^2}_{f_2^w = 1/2} \right\}, \quad \text{where}$$

$$\mu_0^2 = \left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\}, \quad \mu_2^2 = \left\{ \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\}, \quad (4)$$

and

$$C_b = \left\{ \underbrace{\mu_1^2}_{f_0^w = 1/2} \right\}, \quad \text{where } \mu_1^2 = \left\{ \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right\}. \quad (5)$$

In addition, the optimization method obtains one of the (2, 3)-VCS codebooks, which we adopt in Section IV for

performance evaluation and demonstrations as below.

$$
C_w = \left\{ \underbrace{\mu_0^3}_{f_0^w=1/2}, \underbrace{\mu_3^3}_{f_3^w=1/2} \right\} = \left\{ \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}}_{f_0^w=1/2}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}}_{f_3^w=1/2} \right\}, \quad (6)
$$

and

$$
C_b = \left\{ \underbrace{\mu_1^3}_{f_1^b=1/2}, \underbrace{\mu_2^3}_{f_2^b=1/2} \right\}
$$

$$
= \left\{ \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}_{f_1^b=1/2}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}}_{f_2^b=1/2} \right\}. \quad (7)
$$

The advantage of optimization-based probabilistic (k, n)-VCS is that the visual quality (contrast and blackness level) of the recovered image can be controlled efficiently. Hence, this study adopts codebooks in the optimization-based probabilistic (k, n)-VCS to develop a TVCS. The codebooks for (k, n)-VCS, $2 \leq n \leq 5$, $2 \leq k \leq n$, used in the proposed TVCSs [24] are shown in APPENDIX.

B. THE PROPOSED ENCRYPTION ALGORITHM FOR THE (k, n)-TVCS

To improve the visual quality of recovered tag images, we modify the encryption method based on Wang and Hsu's method. The proposed probabilistic (k, n)-TVCS encryption algorithm consists of two phases, as per previous related work [27].

The proposed two-phase (k, n)-TVCS encryption algorithm shares a binary secret image and yields n tagged shares. The black and white pixel colors are presented as logical 1 and 0, respectively. The related notations for the encryption algorithm are listed in TABLE 2. Note that the average density of black pixels in base shares, d, can be obtained from the codebooks of (k, n)-VCS or from the yielded shares. In addition, a given parameter, q, $0 < q \leq 1$, can provide adjustable contrasts of recovered tag images and secret images.

The encryption algorithm is shown in ALGORITHM 3. The input images include one secret image (M) and n tag images (TM_i). Phase I, consisting of Steps 1–4, performs the visual secret sharing procedures. For each coordinate in the secret image (M), the algorithm encrypts a black (white) pixel by the code collections, μ_i^n, in C_b (C_w) with a chosen probability distribution $f_{i,n}^b$ ($f_{i,n}^w$). If set μ_i^n includes n_μ column vectors, each column vector is selected with probability $n_\mu/f_{i,n}^w$ ($n_\mu/f_{i,n}^b$). Then each element of the selected column will be distributed to relative base share. Finally, Phase I generates base shares, $\bar{S}_j$, for $1 \leq j \leq n$.

TABLE 2. List of the notations.

Notations	Descriptions
n	The number of participants.
k	The threshold of decryption.
M	The secret image with size $w \times h$ pixels.
TM_j	The tag images with size $\frac{w}{2} \times h$ pixels, $1 \leq j \leq n$.
q	A given ratio in Phase II, $0 < q \leq 1$.
p_m	A modified ratio while tag pixel value is white.
$\bar{S}_j$	The base shares generated by the first phase with size $w \times h$ pixels, $1 \leq j \leq n$.
S_j	The tagged shares with size $w \times h$ pixels, $1 \leq j \leq n$.
(x, y)	The coordinate in left side of central folding line, $1 \leq x \leq \frac{w}{2}$, $1 \leq y \leq h$.
$(\bar{x}, y)$	The coordinate in right side of central folding line, $\bar{x} = w + 1 - x$.
C_b/C_w	The codebooks of (k, n)-VCS; C_b / C_w is used in the black/white secret pixels.
$f_{i,n}^w/f_{i,n}^b$	The chosen probability distributions of C_w and C_b, respectively. For example, a black secret pixel will be encoded by code $iB(n - i)W$ with probability $f_{i,n}^b$.
d	The density of black pixels in base shares generated by probabilistic (k, n)-VCS.

For example, assume a $(2, 4)$-TVCS is encrypted. First, Phase I encodes white and black secret pixels due to codebooks $(2, 4)$-VCS as following C_w and C_b, respectively.

$$
C_w = \left\{ \underbrace{\mu_0^4}_{0.5}, \underbrace{\mu_4^4}_{0.5} \right\} = \left\{ \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{0.5}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{0.5} \right\},
$$

$$
C_b = \left\{ \underbrace{\mu_2^4}_{1} \right\} \left\{ \underbrace{\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}}_{1.0} \right\}.
$$

When encrypting a white (resp. black) secret pixel, Phase I algorithm randomly selects one column vector from C_w (resp. C_b) by corresponding chosen probabilities. Assume the color of secret pixel is white, column $[0\,0\,0\,0]^T$ and $[1\,1\,1\,1]^T$ are selected with probability 0.5, respectively. When sharing a black secret pixel, because of 6 column vectors in set μ_2^4, each column vector ($[1\,1\,0\,0]^T$, $[1\,0\,1\,0]^T$, ..., $[1\,0\,0\,1\,1]^T$) is selected with probability 1/6. Assume $[1\,1\,0\,0]^T$ is selected, then base share 1 to 4 will be encrypted as "1100", respectively. In addition, according to the codebooks, the density of black pixels in a base share is $d = 0.5$.

Then, in Phase II (Steps 5–21), the tag-hiding procedure encodes tag images (TM_j) into the base shares ($\bar{S}_j$) one by one. In Steps 5 and 6, the left and right symmetric coordinates along the central folding line are regarded as one pair of share pixels ($S(x, y)$ and $S(\bar{x}, y)$) and are processed together in Phase II.

Algorithm 3 The Proposed Encryption Algorithm for the (k, n)-TVCS

Input: A binary secret image M, and Tag images $TM_1, \ldots, TM_n$.

Output: Tagged shares $S_1, \ldots, S_n$.

      // Phase I: Steps 1 to 4

1.   **Input** secret image M
2.   **For** $1 \le x \le w, 1 \le y \le h$, **Repeat** Steps 3 and 4
3.   **If** $M(x, y) = 1$, **then** encrypts $\overline{S}_j(x, y)$, $1 \le j \le n$, by C_b with $f_{i,n}^b$, $0 \le i \le n$.
4.   **else** encrypts $\overline{S}_j(x, y)$, $1 \le j \le n$, by C_w with $f_{i,n}^w$, $0 \le i \le n$.

      // Phase II: Steps 5 to 21

5.   **For** $1 \le x \le \frac{w}{2}, 1 \le y \le h$, **Repeat** Steps 6 to 20   // for each pair of coordination
6. $\overline{x} \leftarrow w + 1 - x$
7.   **For** $1 \le j \le n$, **Repeat** Steps 8 to 20
8. $S_j(x, y) \leftarrow \overline{S}_j(x, y), S_j(\overline{x}, y) \leftarrow \overline{S}_j(\overline{x}, y)$.
9.   **If** $TM_j(x, y) = 1$ **then**
10.   **If** $S_j(x, y) = 0$ **AND** $S_j(\overline{x}, y) = 0$ **then** **Execute** Steps 11 to 13
11.   Generate a random number r
12.   **If** $r \le q$ **then**
13.   Randomly perform $S_j(x, y) \leftarrow 1$ or $S_j(\overline{x}, y) \leftarrow 1$.
14.   **else** GoTo Step 5
15.   **else Execute** Steps 16 to 20 // $TM_j(x, y) = 0$,
16.   Generate a random number r
17.   **If** $S_j(x, y) = 0$ **AND** $S_j(\overline{x}, y) = 1$ **then**
18.   **If** $r \le p_m$ **then** $S_j(x, y) \leftarrow 1$. GoTo Step 5
19.   **If** $S_j(x, y) = 1$ **AND** $S_j(\overline{x}, y) = 0$ **then**
20.   **If** $r \le p_m$ **then** $S_j(\overline{x}, y) \leftarrow 1$. GoTo Step 5
21.   **Output** Tagged shares $S_1, \ldots, S_n$.

In Step 7, n pairs of share pixels are modified one by one. Step 8 copies each pixel from the base shares $\overline{S}_j$ to the tagged shares S_j. If the value of the tag pixel $TM_j(x, y)$ is "1" (black) and the corresponding pair of pixels in the tagged share (i.e., $S_j(x, y)$ and $S_j(\overline{x}, y)$) are both white, then $S_j(x, y)$ and $S_j(\overline{x}, y)$ cannot be stacked and decoded to black for $TM_j(x, y)$. Hence, Steps 11–13 are a modification process. One pixel of $S_j(x, y)$ and $S_j(\overline{x}, y)$ is modified from 0 to 1 with probability q such that the tag pixel $TM_j(x, y)$ will be decoded as black. However, both the pixels in $S_j(x, y)$ and $S_j(\overline{x}, y)$ may not have changed, which has probability $1 - q$.

To avoid revealing the trace of the tags in shares, an equal percentage of white share pixels must be modified to black when the corresponding tag pixel is white. Hence, when the tag pixel is white, Steps 16–20 are executed. The algorithm selects $S_j(x, y)$ and $S_j(\overline{x}, y)$ with pixel values "1W1B" to modify to "2B" by ratio p_m, as shown in Steps 16–20. The ratio p_m will be explained in the next section. Finally, in Step 21, when $w \times h$ coordinates are handled, the algorithm can output n tagged shares.

TABLE 3. An example for encrypting $(2, 4)$-TVCS.

| | Phase I | | Phase II | | | |
| | Base shares | | Tagged shares | | Tag (TM) | Relative encryption steps |
Location	x	$\overline{x}$	x	$\overline{x}$		
Secret (M)	□	■	□	■		
Share 1 (S_1)	□	□	□	■	■	Steps 9-13 modify $S_1(\overline{x})$
Share 2 (S_2)	□	■	□	■	■	No change
Share 3 (S_3)	□	□	□	□	□	No change
Share 4 (S_4)	□	■	■	■	□	Steps 16-18 modify $S_4(x)$

For example, TABLE 3 shows the $(2, 4)$-TVCS encryption process of a pair of pixels located in (x, y) and $(\overline{x}, y)$ the secret pixels are □ and ■. Notations of y-axis coordinates are omitted in TABLE 3. First, assume phase I encrypts the white secret pixel to " □□□□ " and the black pixel is encode to " □■□■ " due to codebooks of probabilistic $(2, 4)$-VCS. Phase II considers relative tag pixels (TM) "■■□□". In row of Share 1, tag is "■" and share pixels are " □□ ". Assume $q = 0.8$ and Step 11 generates random number r, $r < q$, thus Step 13 randomly selects one location from $S_1(x)$ and $S_1(\overline{x})$. Consequently, Steps 9–13 in algorithm modifies $S_1(\overline{x})$ to "■".

In row of Share 4, tag is "□" and share pixels are " □■ ", assume Step 15 generates random number r, and $r < p_m$, thus Step 18 changes $S_4(x)$ to "■". Consequently, the tagged shares $(S_1, \ldots, S_4)$ in location x and $\overline{x}$ are " □□□■ " and " ■■□■ ", respectively. While folding up tagged shares, $S_i(x) + S_i(\overline{x})$, ("+" is logical OR), the recovered tag pixels are " ■■□■ ", the black tag pixels are perfect decrypted.

C. THEORETICAL PERFORMANCE ANALYSIS

In this subsection, we prove that the proposed (k, n)-TVCS satisfies the constraints of a tagged VCS. We also theoretically analyze the visual effects of the proposed (k, n)-TVCS approach.

Property 1: Let modification ratio $p_m = q(1 - d)/2d$, traces of tag image can be eliminated from tagged shares.

    Proof: TABLE 4 lists the related probabilities for analysis. Column 3 shows the appearance probabilities for all combinations ("WW", "WB", "BW", and "BB") of $S_j(x, y)$ and $S_j(\overline{x}, y)$. According to the proposed algorithm, columns 4 and 5 list the modification probabilities, Prob. $(W \rightarrow B)$, for individual base-share-pixel when tag pixel is black and white, respectively.

Next, we consider the whole ratios that white base-share-pixels are changed to black. When color of tag pixel is black and white, the whole ratio is denoted as $P_{tag\ is\ B}(W \rightarrow B)$ and

$S_j(x, y)$	$S_j(\bar{x}, y)$	Appearance probability	Prob. (W → B) for base pixel	
			Color of Tag Pixel	
			Black	White
W	W	$(1-d)^2$	$q \times (1/2)$	0
W	B	$d(1-d)$	0	p_m
B	W	$d(1-d)$	0	p_m
B	B	d^2	0	0
"W": white; "B": black				

$P_{tag\ is\ W}$(W → B), respectively. Hence,

$$P_{tag\ is\ B}(W \to B) = (1-d)^2 \times \left(\frac{q}{2}\right) \times 2 = q(1-d)^2, \quad (8)$$

$$P_{tag\ is\ W}(W \to B) = d(1-d) \times p_m \times 2. \quad (9)$$

To maintain the density balance of black pixels in black and white regions of tag images, the values of $P_{tag\ is\ B}$(W → B) and $P_{tag\ is\ W}$(W → B) have to equal:

$$q(1-d)^2 = 2d(1-d) \cdot p_m. \quad (10)$$

Finally, modification probability p_m is obtained by:

$$p_m = q(1-d)/2d. \quad (11)$$

When modification ratio p_m is set to $q(1-d)/2d$, the proposed encryption algorithm can balance black pixel density between black and white regions of the tag pattern. Therefore, no trace of tag image is revealed in any tagged share is true.□

We now present several notations and definitions to analyze the visual quality. Usually, contrast is used as a metric to measure visual quality and evaluate performance in VCS studies [28]. We refer to the previous researches and define the contrast of recovered tag images and secret images as follows.

Definition 1: The contrast of a binary image is defined as $\alpha = (pw^w - pw^b)/(1 + pw^b)$, where parameters pw^w and pw^b respectively represent the white pixel density in the white and black regions of the secret images.

Based on **Definition 1**, we define and prove the contrast of a recovered tag image.

Property 2 (Contrast of Recovered Tag Image): Let notation α^T represent the contrast of a recovered tag image. Then,

$$\alpha^T = \frac{tw^w - tw^b}{1 + tw^b} = \frac{q(1-d)^2}{1 + (1-q)(1-d)^2}.$$

Notations tw^w and tw^b respectively represent the probabilities that black and white tag pixels are decoded as white in a recovered tag image.

Proof: As per column 4 in TABLE 4 (tag pixel is black), when pixels $S_j(x, y)$ and $S_j(\bar{x}, y)$ are "WB", "BW", or "BB", the recovered tag pixel will be black. However, when pixels $S_j(x, y)$ and $S_j(\bar{x}, y)$ are "WW", the probability for no modification is $1 - q$, so partial black tag pixels are recovered as white. Thus, the probability, tw^b, that a black tag pixel is recovered as black is:

$$tw^b = (1-q)(1-d)^2. \quad (12)$$

Next, as in column 5 in TABLE 4 (tag pixel is white), only when both $S_j(x, y)$ and $S_j(\bar{x}, y)$ are "white" is the recovered tag pixel white. Thus, the probability tw^w is:

$$tw^w = (1-d)^2. \quad (13)$$

Therefore, the contrast of a recovered tag image is obtained, as follows:

$$\alpha^T = \frac{tw^w - tw^b}{1 + tw^b} = \frac{q(1-d)^2}{1 + (1-q)(1-d)^2}. \quad (14)$$

□

Afterwards, for discussing the security and contrast constraints, we define following notations.

Definition 2 (For VCS): Notations $\overline{pw}_t^w$ and $\overline{pw}_t^b$ represent the densities of white pixels in base shares ($\bar{S}_j$) or stacked images, where superscripts "w" and "b" denote colors of secret pixels, white and black, respectively. Subscript t is number of stacked shares.

Definition 3: (For TVCS): Notations pw_t^w and pw_t^b represent the densities of white pixels in tagged shares (S_j) or stacked images, where superscripts "w" and "b" denote colors of security pixels, white and black, respectively. Subscript t is number of stacked shares.

To derive the relationship between $\overline{pw}_t^w$ (resp. $\overline{pw}_t^b$) and pw_t^w (resp. pw_t^b) in VCS and TVCS, respectively, we define a notation u. The notation u represents the remaining ratio that one white base share pixel keeps white in tagged shares after Phase II.

$pw_1^w = \overline{pw}_1^w \cdot u$, for white secret pixels.
$pw_1^b = \overline{pw}_1^b \cdot u$, for black secret pixels.

Property 3: When stacking t shares, the values of pw_t^w (resp. pw_t^b) in tagged shares can be obtained by the given $\overline{pw}_t^w$ (resp. $\overline{pw}_t^b$) in base shares.

$$pw_t^w = \overline{pw}_t^w u^t, \quad (15)$$

$$pw_t^b = \overline{pw}_t^b u^t, \quad (16)$$

where $\quad u = 1 - q(1-d)/2. \quad (17)$

Proof: According to column 4 in TABLE 4 (tag is black pixel), the probability of one share pixel being modified is $q(1-d)^2/2$, which is same as obtain from column 5 in TABLE 4.

Next, the probability that one base-share-pixel is white is $(1-d)$. Therefore, the white share-pixels remaining ratio u is calculated by the following conditional probability:

$$u = 1 - \left(\frac{q(1-d)^2}{2}\right) / (1-d) = 1 - \frac{q(1-d)}{2}.$$

In Phase II of the proposed algorithm, the white pixel modification is independent of the secret pixel's color (white or black), and so the remaining ratios of white pixels in white and black regions of the secret image are equal (and equivalent to u).

If t share pixels in VCS stacked as white, and the t pixels simultaneously keep no change in Phase II of TVCS, then the stacking result is white, which probability is u^t.

Hence, the probabilities pw_t^w and pw_t^b in TVCS can be obtained from the given $\overline{pw}_t^w$ and $\overline{pw}_t^b$ in VCS, as below.

$$pw_t^w = \overline{pw}_t^w u^t \quad \text{and} \quad pw_t^b = \overline{pw}_t^b u^t.$$

Thus, Equations (15), (16), and (17) hold. $\qquad\square$

Moreover, in the special case, q is 1, we obtain

$$u = 1 - (1 - d)/2 = (1 + d)/2$$

such that

$$pw_t^w = \overline{pw}_t^w \left(\frac{1+d}{2}\right)^t \quad \text{and} \quad pw_t^b = \overline{pw}_t^b \left(\frac{1+d}{2}\right)^t \quad \text{are true.}$$

Property 4 (Contrast of Recovered Secret Image): Let α denote the contrast of the recovered image for the proposed probabilistic (k, n)-TVCS, then

$$\alpha = \frac{\overline{pw}_k^w - \overline{pw}_k^b}{u^{-k} + \overline{pw}_k^b}. \qquad (18)$$

Proof: According to **Definition 1**, the contrast is

$$\alpha = \frac{pw_k^w - pw_k^b}{1 + pw_k^b} = \frac{\overline{pw}_k^w u^k - \overline{pw}_k^b u^k}{1 + \overline{pw}_k^b u^k} = \frac{\overline{pw}_k^w - \overline{pw}_k^b}{u^{-k} + \overline{pw}_k^b}.$$

$\qquad\square$

Moreover, when q is 1, we obtain:

$$\alpha = \frac{\overline{pw}_k^w - \overline{pw}_k^b}{\left(\frac{1+d}{2}\right)^{-k} + \overline{pw}_k^b}.$$

In addition, according to Yang [23], Yang *et al.* [29] define security condition for probabilistic VCS, the study redefines the security condition for VCS, and TVCS as following **Definition 4** and **Definition 5**.

Definition 4 (Security Condition for (k, n)-VCS): For any t rows in codebooks of (k, n)-VCS with $t < k$, the values of $\overline{pw}_t^w$ and $\overline{pw}_t^b$ are the same, i.e., $\overline{pw}_t^w = \overline{pw}_t^b$, $t < k$.

The security condition for the optimization based probabilistic (k, n)-VCS have been proved in reference [24], therefore, the base shares (output of Phase I) are satisfied the security condition.

Next, giving an example, $(3, 4)$-VCS, to explain the security constraints in VCSs. Codebooks C_w and C_b of $(3, 4)$-VCS are:

$$C_w = \left\{ \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{1/3}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{2/3} \right\},$$

$$C_b = \left\{ \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}}_{2/3}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{1/3} \right\}.$$

Let $\lambda(\cdot)$ represent the stacking operation that stacks t-tuple of sets. For $t = 1$ to 3, the analysis is as below.

$$\lambda(C_w, t=1) = \lambda\left(\underbrace{[0]}_{1/3}, \underbrace{[1][1][1][0]}_{2/3} \right), \quad \text{thus,}$$
$$\overline{pw}_1^w = 1/3 + 1/6 = 1/2.$$

$$\lambda(C_b, t=1) = \lambda\left(\underbrace{[1][0][0][0]}_{2/3}, \underbrace{[1]}_{1/3} \right), \quad \text{thus,}$$
$$\overline{pw}_1^b = (2/3) \times (3/4) = 1/2.$$

$$\lambda(C_w, t=2) = \lambda\left(\underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{1/3}, \underbrace{\begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{2/3} \right)$$
$$= \left(\underbrace{[0]}_{1/3}, \underbrace{[1][1][1][1]}_{2/3} \right), \quad \text{then } \overline{pw}_2^w = 1/3.$$

$$\lambda(C_b, t=2) = \lambda\left(\underbrace{\begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{2/3}, \underbrace{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}_{1/3} \right)$$
$$= \left(\underbrace{[1][1][0][0]}_{2/3}, \underbrace{[1]}_{1/3} \right), \quad \text{thus}$$
$$\overline{pw}_2^b = (2/3) \times (2/4) = 1/3.$$

Hence, due to above analysis, $\overline{pw}_1^w = \overline{pw}_1^b$, and $\overline{pw}_2^w = \overline{pw}_2^b$, $1 \leq t < 3$, the security conditions are true in $(3, 4)$-VCS.

Definition 5 (Security Condition for (k, n)-TVCS): For any t rows in codebooks of (k, n)-TVCS with $t < k$, the values of pw_t^w and pw_t^b are the same, i.e., $pw_t^w = pw_t^b$, $t < k$.

Property 5: The security constraint for the proposed probabilistic (k, n)-TVCS is true.

Proof: According to Equations (15) and (16) in **Property 3**, pw_t^w and pw_t^b are:

$$pw_t^w = \overline{pw}_t^w u^t, pw_t^b = \overline{pw}_t^b u^t, \quad \text{for } 1 \leq t \leq n.$$

Moreover, due to the security condition in VCS (**Definition 4**), we obtain $\overline{pw}_t^w = \overline{pw}_t^b$, $t < k$ are hold, which makes $\overline{pw}_t^w u^t = \overline{pw}_t^b u^t$. Thus, in TVCS, $pw_t^w = pw_t^b$, for $t < k$, are true. So the security condition is hold in tagged shares. $\qquad\square$

IV. PERFORMANCE EVALUATION

This section compares the proposed approach with related work in terms of the visual qualities of the recovered images. Then, we demonstrate experimental results.

A. PERFORMANCE COMPARISON WITH WANG AND HSU'S APPROACH

This subsection compares the proposed probabilistic (k, n)-TVCS with Wang and Hsu's approach [20]. Wang and Hsu's approach allows the adoption of any reported (k, n)-VCS constructions in the first phase. Thus, for fairness, we adopt the optimization-based probabilistic (k, n)-VCS to encrypt the base shares for both Wang and Hsu's approach and our approach. Moreover, when parameter q is set to 1, the proposed (k, n)-TVCS and Wang and Hsu's approach have equal modification ratios for white pixels in a base share. Thus, the contrasts of the recovered secret images for both approaches are equal. Hence, this subsection only compares the proposed approach with Wang and Hsu's approach in terms of the contrast of recovered tag images (α^T).

In Wang and Hsu's approach [20], white pixel density is denoted as P_0, $P_0 = 1 - d$. When the tag pixel is white and the corresponding base-share-pixel is white, the modification probability is $P_0/2$. The probability that one pair of the base-share-pixel is "WW" is $(P_0)^2$. The probability that the two share pixels stay "WW" after modification is $(1 - P_0/2)^2$. Thus, in the recovered tag image, the probability that one white tag pixel is decoded as white is

$$P_0{}^2 \left(1 - P_0/2\right)^2 = (1 - d)^2 \left(\frac{1 + d}{2}\right)^2.$$

Moreover, in the approach of Wang and Hsu, black tag pixels are fully decoded as black. Therefore the contrast of the recovered tag image, α^T_{Wang}, is:

$$\alpha^T_{Wang} = \frac{tw^w - tw^b}{1 + tw^b} = (1 - d)^2 \left(\frac{1 + d}{2}\right)^2. \quad (19)$$

Next, based on the theoretical analysis results, we list the contrast of recovered tag images, α^T_{This}, from $n = 2$ to 5, i.e., $(2, 2)$ to $(5, 5)$ in TABLE 5. The proposed (k, n)-TVCS adopts parameter $q = 1$. To compare with α^T_{Wang}, improvement factor $\Delta \alpha^T$ is defined as the following equation:

$$\Delta \alpha^T = \frac{\left| \alpha^T_{This} - \alpha^T_{Wang} \right|}{\alpha^T_{Wang}}. \quad (20)$$

Obviously, the proposed (k, n)-TVCS algorithm outperforms Wang and Hsu's approach in terms of the contrast of recovered tag (α^T). For example, with $(2, 2)$-TVCS, due to the codebook in Equations (4) and (5), the proposed approach obtains $d = 0.5$. Then according to **Property 2**, let q be 1. Thus, contrast $\alpha^T_{This} = 25\%$, as below:

$$\alpha^T_{This} = \frac{q(1 - d)^2}{1 + (1 - q)(1 - d)^2} = \frac{q(0.25)}{1 + (1 - q)(0.25)} = 25\%.$$

By comparing, according to Equations (19) and (20), Wang and Hsu's approach obtains:

$$\alpha^T_{Wang} = (0.5)^2 \left(\frac{1.5}{2}\right)^2 = 14.1\%.$$

TABLE 5. Comparing contrasts of recovered tag images (α^T) with Wang *et al.*'s approach.

k	Approach	n			
		2	3	4	5
2	Wang and Hsu	14.1%	14.1%	14.1%	10.2%
	This study	25%	25%	25%	16%
	$\Delta \alpha^T$	77.8%	77.8%	77.8%	56.3%
3	Wang and Hsu		14.1%	14.1%	7.8%
	This study		25%	25%	11.17%
	$\Delta \alpha^T$		77.8%	77.8%	44.1%
4	Wang and Hsu			14.1%	10.2%
	This study			25%	16.01%
	$\Delta \alpha^T$			77.8%	56.3%
5	Wang and Hsu				14.1%
	This study				25%
	$\Delta \alpha^T$				77.8%

Note: This study uses $q = 1$.

Thus, the proposed approach improves on Wang and Hsu's method by 77.8%.

TABLE 5 lists all α^T for (k, n)-TVCS, $2 \leq n \leq 5$, $2 \leq k \leq n$. The improvements range from 44.1% to 77.8%. The analysis results show that our approach significantly improves the contrasts of recovered tag images.

B. PERFORMANCE COMPARISON WITH WU AND SUN'S APPROACH

This subsection compares Wu and Sun's RG-based method [21] and the proposed encryption approach (parameter $q = 1$) in terms of visual quality, according to the theoretical analysis results.

First, we refer to Wu and Sun's analysis for the RG-based (k, n)-TVCS [21]. The contrast of the recovered image (denoted as α_{Wu}) and the contrast of the recovered tag image (denoted as α^T_{Wu}) are:

$$\alpha_{Wu} = \frac{2}{\left(2^k + 1\right) \binom{n}{k} - 1}. \quad (21)$$

$$\alpha^T_{Wu} = \frac{n - 2}{2n + 1}. \quad (22)$$

Next, the contrasts of the recovered secret images and recovered tag images, $2 \leq n \leq 5$ and $2 \leq k \leq n$, are evaluated. The contrasts of Wu and Sun's approach are obtained by Equations (21) and (22). Moreover, the contrasts of the proposed (k, n)-TVCS are obtained from **Property 2**, **Property 3**, and **Property 4**, and $q = 1$. The probabilities $\overline{pw}^w$ and $\overline{pw}^b$ refer to the codebooks in the optimization-based probabilistic (k, n)-VCS [24].

Example: The contrasts of $(2, 3)$-TVCS.

TABLE 6. Comparing contrasts of recovered images (α) with Wu *et al.*'s approach.

k	Approach	n			
		2	3	4	5
2	Wu and Sun	none	14.3%	6.9%	4.1%
	This study	28.1%	17.1%	17.1%	18.1%
3	Wu and Sun		25%	5.7%	2.0%
	This study		10.6%	6.6%	6.2%
4	Wu and Sun			12.5%	2.4%
	This study			4.0%	2.7%
5	Wu and Sun				6.25%
	This study				1.5%

Note: This study sets parameter $q = 1$.

TABLE 7. Comparing contrasts of recovered tag images (α^T) with Wu *et al.*'s approach.

k	Approach	n			
		2	3	4	5
2	Wu and Sun	none	14.3%	22.2%	27.3%
	This study	25.0%	25.0%	25.0%	16%
3	Wu and Sun		14.3%	22.2%	27.3%
	This study		25.0%	25.0%	11.2%
4	Wu and Sun			22.2%	27.3%
	This study			25.0%	16.0%
5	Wu and Sun				27.3%
	This study				25.0%

Note: This study sets parameter $q = 1$.

1) Wu and Sun's approach will obtain the following contrasts of recovered secret images and recovered tag images:

$$\alpha_{Wu} = \frac{2}{\left(2^2 + 1\right)\binom{3}{2} - 1} = \frac{1}{7} = 0.143,$$

$$\alpha_{Wu}^T = \frac{3 - 2}{6 + 1} = \frac{1}{7} = 0.143.$$

2) Our study encodes a $(2, 3)$-TVCS, due to codebooks in (6) and (7), then density $d = 0.5$, $\overline{pw}^w = 1/2$ and $\overline{pw}^b = 1/6$. According to **Property 2**, **Property 3**, and **Property 4**, the visual qualities are:

$$\alpha_{This}^T = \frac{(1 - d)^2}{1 + (1 - d)^2} = 20.0\%,$$

Since $u = 1 - \frac{1-d}{2} = \frac{3}{4}$, then $\alpha_{This} = \frac{6}{35} = 17.1\%$. $\quad\square$

Subsequently, TABLE 6 and TABLE 7 list the contrasts of recovered images (α) and recovered tag images (α^T), respectively. The better contrast between the two approaches is highlighted gray. The contrasts of the proposed approach are larger than those of Wu and Sun's RG-based method for $k = 2$, i.e., $(2, 2)$- to $(2, 5)$-TVCSs, and for the $(3, 4)$-, $(3, 5)$-, and $(4, 5)$-TVCSs.

Wu and Sun's RG-based method has better contrasts of recovered images in the $(3, 3)$-, $(4, 4)$-, and $(5, 5)$-TVCSs, suggesting that Wu and Sun's method can obtain better contrasts than our approach for $k = n, n \geq 3$. However, when $k < n$, our approach obtains higher contrasts of recovered images than Wu and Sun's method.

Moreover, the contrasts of recovered tag images from the theoretical analysis results are illustrated in TABLE 7. The results show that when $n = 2, 3$, or 4, the proposed approach obtains better contrasts of recovered tag images than Wu and Sun's method. However, when amount of participant n is greater than or equal to 5, Wu and Sun's method has higher contrasts of recovered tag images than our approach.

In summary, when amount of participant n is less than 5 and $k < n$, the proposed (k, n)-TVCS outperforms the related methods in terms of visual quality of recovered images and recovered tag images. In addition, Wu and Sun's approach

FIGURE 3. Adjustable contrasts in $(2, 2)$-TVCS.

cannot provide $(2, 2)$-TVCS encryption, which is addressed in the proposed approach.

C. EVALUATION FOR ADJUSTABLE VISUAL QUALITY
In this subsection, we observe the visual quality by adjusting parameter q in the proposed $(2, 2)$- and $(2, 3)$-TVCSs.

1) $(2, 2)$-TVCS
According to the results of the theoretical analysis, the visual quality values of the proposed $(2, 2)$-TVCS for recovered images are $\alpha = 28.1\%$ and $\alpha^T = 25\%$ when $q = 1$.

In the experiment, parameter q varies from 0 to 1 and the corresponding contrasts of the recovered tag images and secret image are as shown in FIGURE 3. FIGURE 3 shows that when parameter q decreases, the contrast of the recovered secret image (α) increases, but the contrast of the tag image (α^T) decreases. For example, when q decreases from 1.0 to 0.5, the contrast of the recovered secret image increases from 28.1% to 38.3%, but the contrast of the tag image decreases from 25.0% to 11.1%. A special case, when $q = 0$, the result is a $(2, 2)$-VCS, and contrast of the tag image (α^T) is zero.

2) $(2, 3)$-TVCS
We use the optimization-based $(2, 3)$-VCS codebooks in Equations (6) and (7) to encode the $(2, 3)$-TVCS.
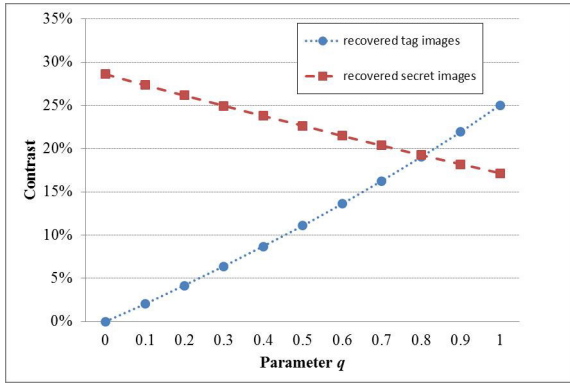
FIGURE 4. Adjustable contrasts in (2, 3)-TVCS.

By theoretical analysis, the visual qualities for the proposed (2, 3)-TVCS are $\alpha = 17.1\%$ and $\alpha^T = 25.0\%$ when $q = 1$.

We varied parameter q from 0 to 1.0, the values of α and α^T are shown in FIGURE 4. When q decreases from 1.0 to 0.5, the contrast of the recovered secret image increases (from 17.1% to 22.6%) but the contrast of the tag image decreases (from 25.0% to 11.1%). When q decreases to zero, the values of α^T is zero, which is (2, 3)-VCS with $\alpha = 28.6\%$.

This experiment demonstrates the proposed approach provides adjustable contrasts between α and α^T by setting parameter q. The adjustable flexibility makes users can adjust the encryption outcomes according to their requirements.

D. CONTRAST VARIATIONS
The subsection conducts a set of experiments to observe the contrast variations with number of stacking shares increase. The experiments implement the proposed (k, n)-TVCS in several scenarios, including $n = 2$ to 5, $k = 2$ to n, by secret image "MCU" and tag images "A" to "E" in FIGURE 5. We set the parameter q as 1 and 0, each scenario runs 10 times and the statistical results are illustrated in TABLE 8. First, in TABLE 8, for $q = 1$, the contrasts variation of recovered images (α) are listed for $k \leq t \leq n$, from (2, 2)-TVCS to (5, 5)-TVCS.

For (k, n)-TVCS, $k < n$, when number of staked shares (t) is greater than k, the contrast value (α) progressively increases, then even decreases if $n - k \geq 2$. For example, (3, 4)-TVCS obtains $\alpha = 7.37\%$ and 12.92%, that progressively increase for $t = 3$ and 4, respectively. For example, in (2, 5)-TVCS, when t are 2 to 3, contrast values increase from 18.62% to 22.13%. But when t are 4 and 5, contrast values decrease to 19.14% and 16.96%. We observe the experimental results and explain the reason for the variation. When t is 2, few black secret pixels still are decoded as white. When number of stacked shares t is 3, the black secret pixels have been fully recovered as black, which causes the contrast value increase. When the number of stacked shares continues to increase ($t = 4, 5$), the few secret pixels in the
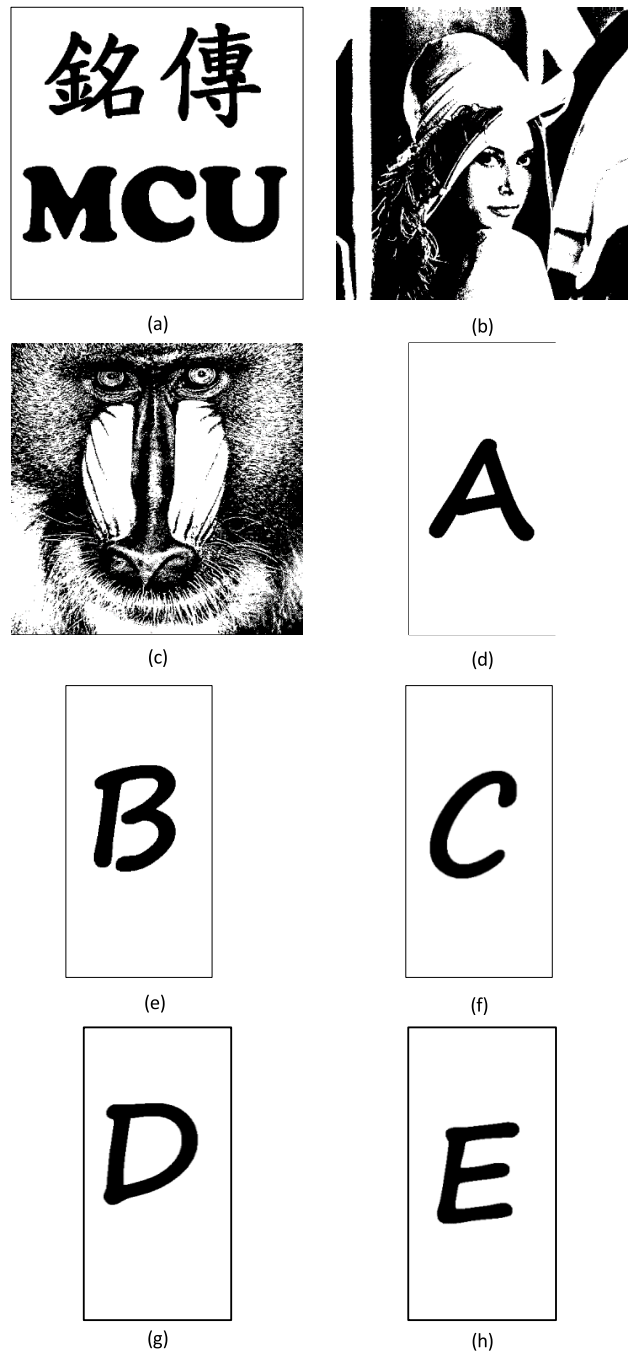
FIGURE 5. The images used in the experiment: (a)–(c) are secret images, "MCU", "Lena", and "Baboon". (d)–(h) are tag images, "A" to "E".

white area are decoded to the black pixels, which will make the contrast α decrease instead. Next, the recovered contrast (α), maximum and minimum α are sequentially listed in TABLE 8.

Moreover, the contrasts of recovered tag images (α_T) in the experiments are also listed in TABLE 8. Last, while the parameter q is set as 0, the (k, n)-TVCS become a (k, n)-VCS, the experimental results of contrasts (α) are listed in TABLE 8.

TABLE 8. Implementation results for contrast variations.

		$q = 1$								$q = 0$
		contrasts variation with # of staked shares (t)								
K	N	2	3	4	5	α	$max\ \alpha$	$min\ \alpha$	α_T	α
2	2	29.98%				29.98%	29.98%	29.98%	24.95%	49.96%
2	3	18.88%	25.80%			18.88%	25.80%	18.88%	25.07%	33.36%
3	3		10.76%			10.76%	10.76%	10.76%	24.99%	25.00%
2	4	18.84%	25.71%	22.88%		18.84%	25.71%	18.84%	25.01%	28.60%
3	4		7.37%	12.92%		7.37%	12.92%	7.37%	25.00%	14.29%
4	4			3.97%		3.97%	3.97%	3.97%	24.96%	12.49%
2	5	18.62%	22.13%	19.14%	16.96%	18.62%	22.13%	16.96%	15.96%	27.26%
3	5		6.31%	8.26%	7.06%	6.31%	8.26%	6.31%	11.19%	10.55%
4	5			2.78%	4.67%	2.78%	4.67%	2.78%	16.01%	6.27%
5	5				1.48%	1.48%	1.48%	1.48%	25.00%	6.24%
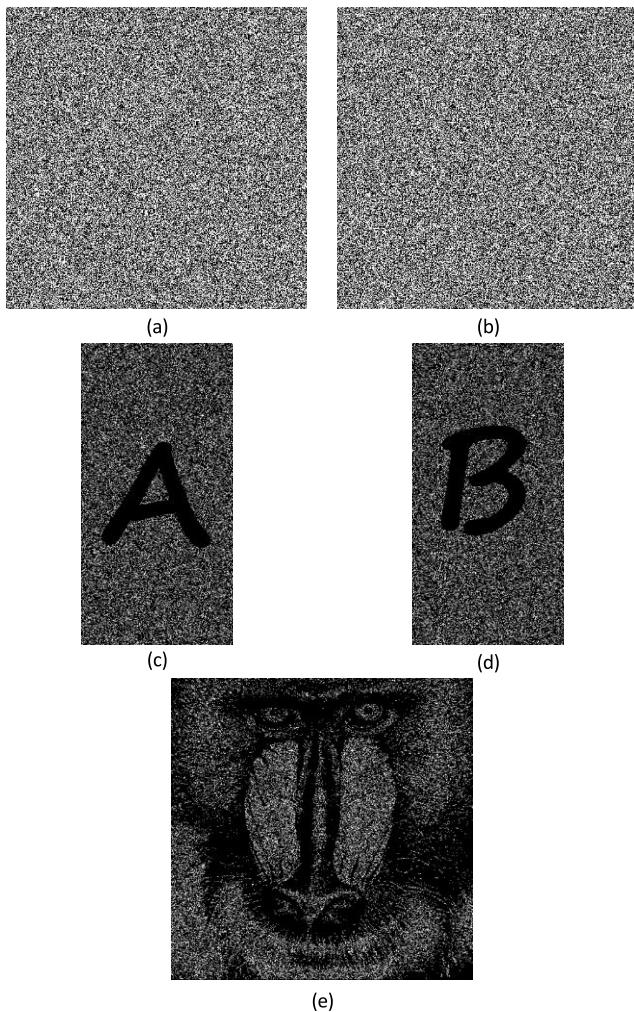
FIGURE 6. The implementation results of (2, 2)-TVCS in Experiment I, $q = 1$. (a) Tagged share 1. (b) Tagged share 2. (c) Recovered tag image 1. (d) Recovered tag image 2. (e) Recovered secret image.

E. DEMONSTRATIONS

In this subsection, we evaluate the visual effects of the proposed encryption algorithm by observing the implementation results. The experiments adopt binary secret images, "MCU", "Lena", and "Baboon", with 512×512 pixels as shown in FIGURE 5(a), (b), and (c), respectively. In FIGURE 5(d)–(h), images "A", "B", to "E" with 256×512 pixels are adopted as tag images in the experiments. Assume the tagged shares are folded up along the central vertical axis.

1) EXPERIMENT I: (2, 2)-TVCS

The first experiment adopts "Baboon", "A", and "B" as one secret image and two tag images for constructing a (2, 2)-TVCS. The implementation results are illustrated in FIGURE 6. The two tagged shares appear noise-like, as shown in FIGURE 6(a) and (b).

When folding up tagged shares, FIGURE 6(c) and (d) show recovered tag images. The contrasts α^T are 25.2%, and 25.0%, which are very close to its theoretical value 25%. Moreover, FIGURE 6(e) shows the recovered secret image. The experimental contrast value (α) is 28.6%, which is close to the theoretical contrast of 28.1%.

2) EXPERIMENT II: (3, 3)-TVCS

This experiment presents the implementation results of a (3, 3)-TVCS with $q = 1$. Image "Lena" is adopted as the secret image, and images "A", "B", and "C" are tag images. The implementation results of (3, 3)-TVCS are illustrated in FIGURE 7. The tagged shares appear noise-like, as shown in FIGURE 7(a), (b), and (c).

When stacking any two tagged shares, the stacked images appear noise-like, which is a security constraint of VCS, as illustrated in FIGURE 7(d), (e) and (f). When the three tagged shares are stacked, the secret image "Lena" is decoded as FIGURE 7(g). The recovered contrast of the secret image (α) is 10.5%, which is very close to the theoretical contrast of 10.6%. Moreover, while folding up tagged shares, FIGURE 7(h), (i) and (j) show three recovered tag images. The recovered contrasts α^T are 24.9%, 25.1%, and 25.3%, which align with the theoretical contrast 25%.
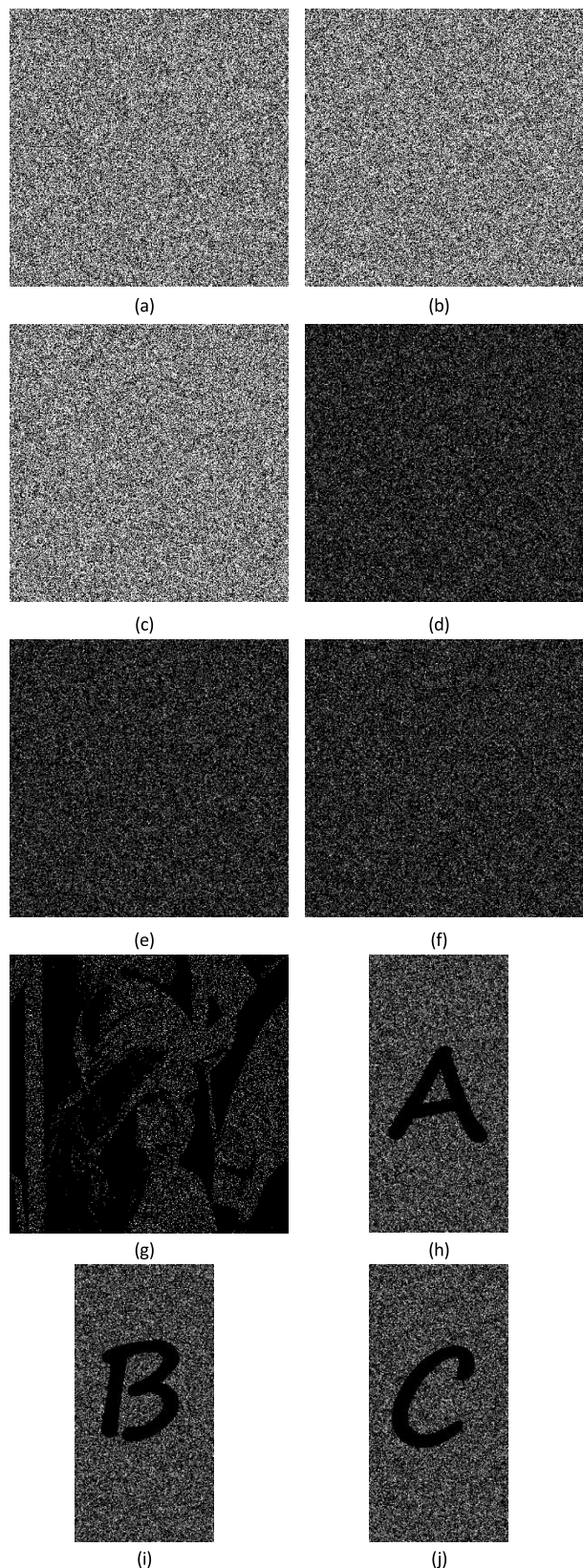
FIGURE 7. The implementation results of (3, 3)-TVCS in Experiment II, $q = 1$. (a) Tagged share 1. (b) Tagged share 2. (c) Tagged share 3. (d) Stacked image (1+2). (e) Stacked image (1+3). (f) Stacked image (2+3). (g) Recovered secret image. (h)-(j) Recovered tag images 1, 2, and 3.
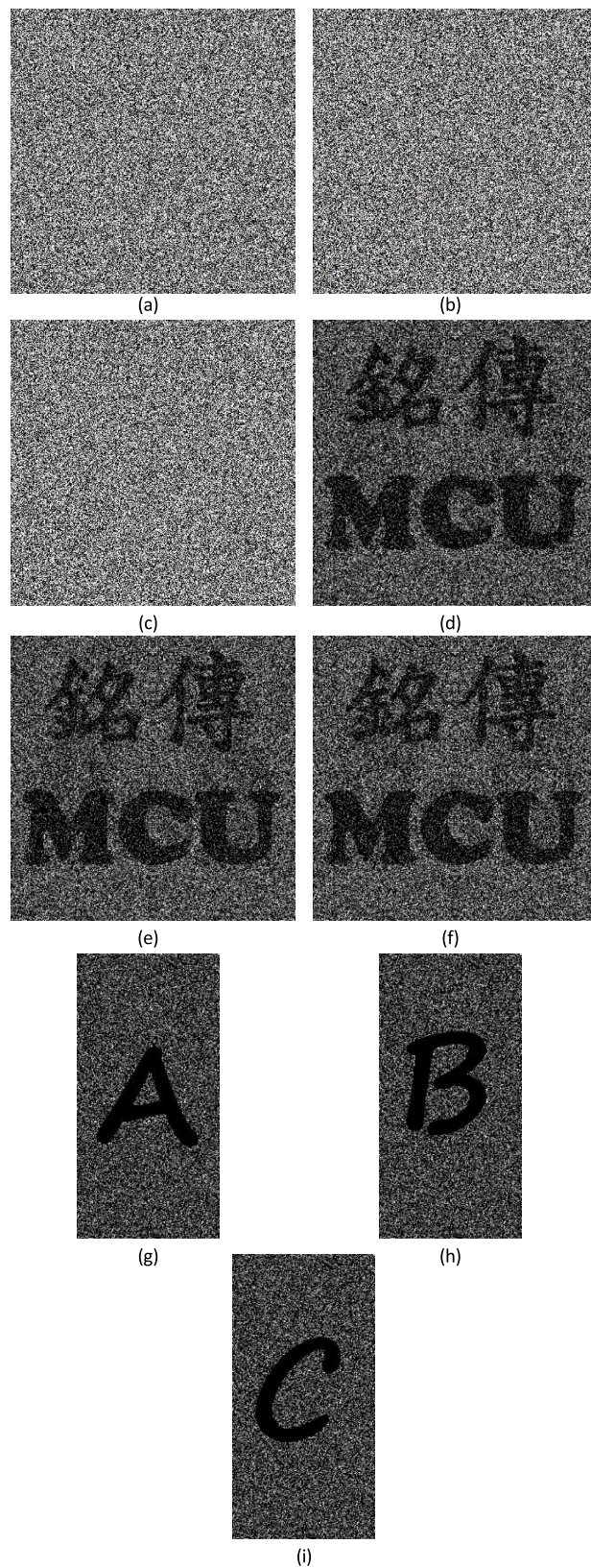
FIGURE 8. The implementation results of (2, 3)-TVCS in Experiment III, $q = 1$. (a)-(c) Tagged shares 1,2,and 3. (d) Stacked image (1+2). (e) Stacked image (1+3). (f) Stacked image (2+3). (g)-(i) Recovered tag images 1, 2, and 3.
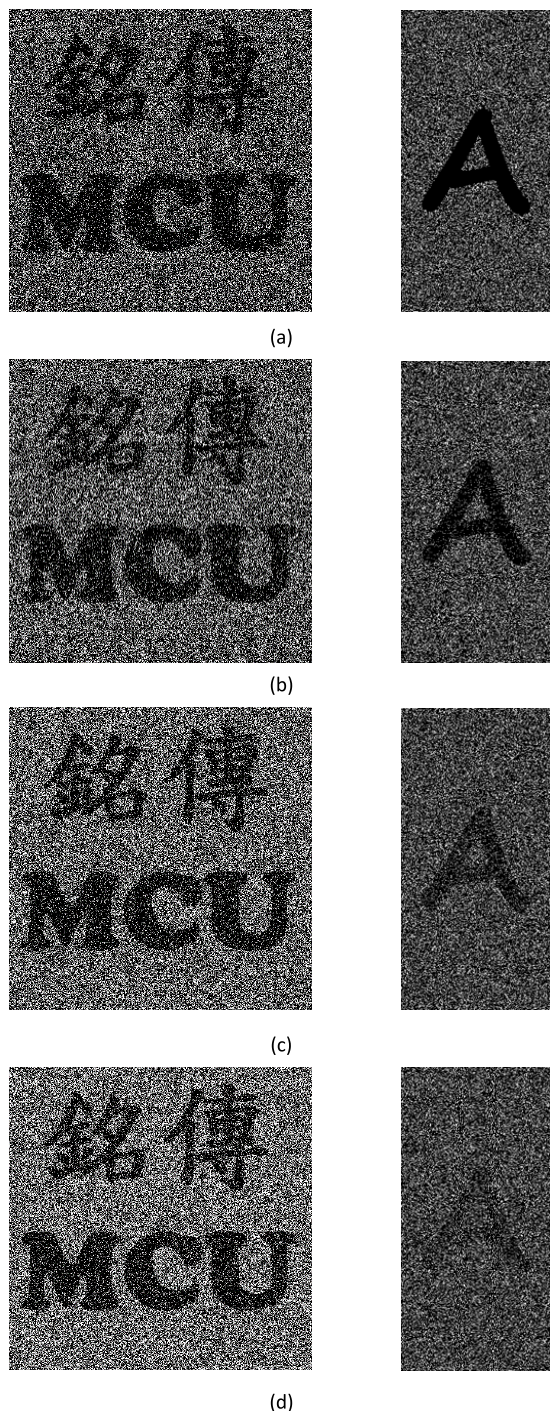
FIGURE 9. The implementation results of (2, 3)-TVCS with various *q* values, including one of recovered secret images and one of tag images. (a) *q* = 1. (b) *q* = 0.8. (c) *q* = 0.5. (d) *q* = 0.3.

3) EXPERIMENT III: (2, 3)-TVCS

This experiment implements (2, 3)-TVCS with various *q* values. The secret image is "MCU", and the three tag images are "A", "B", and "C". First, set $q = 1$. The implementation results are illustrated in FIGURE 8, where (a), (b), and (c) are noise-like tagged shares. When any two shares are stacked, the secret image is decrypted as shown

in FIGURE 8(d), (e), and (f). The recovered contrasts α are 19.0%, 18.5%, and 19.0%, which are slightly higher than the theoretical contrast of 17.1%. Moreover, the tag images can be revealed by folding up each tagged share, as shown in FIGURE 8(g), (h), and (i). The contrasts of tag images α^T are all 24.8%, which are very close to the theoretical contrast of 25.0%.

4) EXPERIMENT IV: (2,3)-TVCS WITH VARIOUS Q

Next, we implement (2, 3)-TVCSs for four scenarios, i.e., values of *q* are 1.0, 0.8, 0.5, and 0.3, to verify that the recovered contrasts can be slightly adjusted by parameter *q*. Due to space limitations, we select only one of the recovered images (stacking shares 1 and 2) and one of the recovered tag images, "A", for each scenario to be illustrated in FIGURE 9 (a) to (d). As shown in FIGURE 9(a) to (d), when the value of *q* decreases from 1.0 to 0.3, the contrasts (α) of the recovered images "MCU" gradually increase, from 19.0%, 20.0%, 23.1%, to 25.0%, providing clearer visual percepts. On the other hand, the contrasts of tag images "A" (α^T) gradually reduces, from 24.8%, 18.8%, 11.2%, to 6.5%, significantly decreasing the visual quality. According to requirements, the user can choose an appropriate *q* value. The experiment shows that the proposed (k, n)-TVCS can provide the feature of adjustable contrasts for users.

In summary, the experimental results show the proposed approach: 1) provides correct (k, n)-TVCSs, 2) provides contrasts trade-off between recovered secret and tag images. Moreover, the theoretical analysis performance is aligned with the implemental results.

V. CONCLUSION

In this paper, we proposed a two-phase tagged visual cryptography construction method, the (k, n)-TVCS, that adopts an optimized probabilistic visual cryptography scheme (VCS) to encode base shares and hide tag images. The proposed approach has the following key features. First, the proposed (k, n)-TVCS encryption algorithm is a systematic approach that can encrypt threshold TVCS for any *k* and *n*. Second, it addresses the pixel-expansion problem in traditional VCS. Third, the proposed approach provides a feature for tuning the contrasts of recovered secret images and tag images by a parameter so that users can adjust visual qualities according to their requirements.

This study presents a theoretical analysis of visual quality. Then, experiments showed that the results of the theoretical analysis and implementation are very close, thus confirming the accuracy and effectiveness of the proposed encryption algorithm. Moreover, we compared the performance of the proposed algorithm with related works. The results showed that when amount of participant *n* is less than 5 and threshold, $k < n$, the proposed (k, n)-TVCS outperforms the other methods in terms of the visual quality of recovered secret and tag images. In addition, this study solves the problem that RG-based TVCS does not work for (2, 2)-TVCS. Finally, this study makes helpful contributions to the field of threshold tagged visual cryptography.

APPENDIX

CODEBOOKS FOR PROBABILISTIC (k, n)-VCS IN PHASE I

1) (2,2)

$$C_w = \left\{ \underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{1/2}, \underbrace{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}_{1/2} \right\}, \quad C_b = \left\{ \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}}_{1} \right\},$$

$$d = 0.5, \quad \alpha = 50\%.$$

2) (2,3)

$$C_w = \left\{ \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}}_{1/2}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}}_{1/2} \right\},$$

$$C_b = \left\{ \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}_{1/2}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}}_{1/2} \right\},$$

$$d = 0.5, \quad \alpha = 28.57\%.$$

3) (3,3)

$$C_w = \left\{ \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}}_{1/4}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}}_{3/4} \right\},$$

$$C_b = \left\{ \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}_{3/4}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}}_{1/4} \right\}, \quad d = 0.5, \ \alpha = 25\%.$$

4) (2,4)

$$C_w = \left\{ \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{1/2}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{1/2} \right\}$$

$$C_b = \left\{ \underbrace{\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}}_{1} \right\},$$

$$d = 0.5, \ \alpha = 28.57\%.$$

5) (3,4)

$$C_w = \left\{ \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{1/3}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{2/3} \right\},$$

$$C_b = \left\{ \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}}_{2/3}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{1/3} \right\},$$

$$d = 0.5, \quad \alpha = 14.28\%.$$

6) (4,4)

$$C_w = \left\{ \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{1/8}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}}_{3/4}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{1/8} \right\},$$

$$C_w = \left\{ \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{2,5}, \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{3/5} \right\}, \quad C_b = \left\{ \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{1} \right\}, \quad d = 0.6, \ \alpha = 27.27\%$$

$$
C_w = \left\{
\underbrace{\begin{bmatrix}0\\0\\0\\0\\0\end{bmatrix}}_{0.168},\;
\underbrace{\begin{bmatrix}1\\1\\1\\0\\0\end{bmatrix}\begin{bmatrix}1\\1\\1\\1\\0\end{bmatrix}\begin{bmatrix}1\\1\\0\\0\\1\end{bmatrix}\begin{bmatrix}1\\0\\1\\1\\0\end{bmatrix}\begin{bmatrix}1\\0\\1\\0\\1\end{bmatrix}\begin{bmatrix}1\\0\\0\\1\\1\end{bmatrix}\begin{bmatrix}0\\1\\1\\1\\0\end{bmatrix}\begin{bmatrix}0\\1\\1\\0\\1\end{bmatrix}\begin{bmatrix}0\\1\\0\\1\\1\end{bmatrix}\begin{bmatrix}0\\0\\1\\1\\1\end{bmatrix}}_{0.001},\;
\underbrace{\begin{bmatrix}1\\1\\1\\1\\0\end{bmatrix}\begin{bmatrix}1\\1\\1\\0\\1\end{bmatrix}\begin{bmatrix}1\\1\\0\\1\\1\end{bmatrix}\begin{bmatrix}1\\0\\1\\1\\1\end{bmatrix}\begin{bmatrix}0\\1\\1\\1\\1\end{bmatrix}}_{0.831}
\right\},
$$

$$
C_b = \left\{
\underbrace{\begin{bmatrix}1\\0\\0\\0\\0\end{bmatrix}\begin{bmatrix}0\\1\\0\\0\\0\end{bmatrix}\begin{bmatrix}0\\0\\1\\0\\0\end{bmatrix}\begin{bmatrix}0\\0\\0\\1\\0\end{bmatrix}\begin{bmatrix}0\\0\\0\\0\\1\end{bmatrix}}_{0.003},\;
\underbrace{\begin{bmatrix}1\\1\\0\\0\\0\end{bmatrix}\begin{bmatrix}1\\0\\1\\0\\0\end{bmatrix}\begin{bmatrix}1\\0\\0\\1\\0\end{bmatrix}\begin{bmatrix}1\\0\\0\\0\\1\end{bmatrix}\begin{bmatrix}0\\1\\1\\0\\0\end{bmatrix}\begin{bmatrix}0\\1\\0\\1\\0\end{bmatrix}\begin{bmatrix}0\\1\\0\\0\\1\end{bmatrix}\begin{bmatrix}0\\0\\1\\1\\0\end{bmatrix}\begin{bmatrix}0\\0\\1\\0\\1\end{bmatrix}\begin{bmatrix}0\\0\\0\\1\\1\end{bmatrix}}_{0.553},\;
\underbrace{\begin{bmatrix}1\\1\\1\\1\\1\end{bmatrix}}_{0.444}
\right\}
$$

$$d = 0.665, \quad \alpha = 10.51\%$$

$$
C_w = \left\{
\underbrace{\begin{bmatrix}0\\0\\0\\0\\0\end{bmatrix}}_{0.134},\;
\underbrace{\begin{bmatrix}1\\1\\1\\0\\0\end{bmatrix}\begin{bmatrix}1\\1\\0\\1\\0\end{bmatrix}\begin{bmatrix}1\\1\\0\\0\\1\end{bmatrix}\begin{bmatrix}1\\0\\1\\1\\0\end{bmatrix}\begin{bmatrix}1\\0\\1\\0\\1\end{bmatrix}\begin{bmatrix}1\\0\\0\\1\\1\end{bmatrix}\begin{bmatrix}0\\1\\1\\1\\0\end{bmatrix}\begin{bmatrix}0\\1\\1\\0\\1\end{bmatrix}\begin{bmatrix}0\\1\\0\\1\\1\end{bmatrix}\begin{bmatrix}0\\0\\1\\1\\1\end{bmatrix}}_{0.666},\;
\underbrace{\begin{bmatrix}1\\1\\1\\1\\1\end{bmatrix}}_{0.2}
\right\},
$$

$$
C_b = \left\{
\underbrace{\begin{bmatrix}1\\0\\0\\0\\0\end{bmatrix}\begin{bmatrix}0\\1\\0\\0\\0\end{bmatrix}\begin{bmatrix}0\\0\\1\\0\\0\end{bmatrix}\begin{bmatrix}0\\0\\0\\1\\0\end{bmatrix}\begin{bmatrix}0\\0\\0\\0\\1\end{bmatrix}}_{0.334},\;
\underbrace{\begin{bmatrix}1\\1\\1\\1\\0\end{bmatrix}\begin{bmatrix}1\\1\\1\\0\\1\end{bmatrix}\begin{bmatrix}1\\1\\0\\1\\1\end{bmatrix}\begin{bmatrix}1\\0\\1\\1\\1\end{bmatrix}}_{0.666}
\right\}, \; d = 0.6, \alpha = 6.26\%
$$

$$
c_w = \left\{
\underbrace{\begin{bmatrix}0\\0\\0\\0\\0\end{bmatrix}}_{0.0625},\;
\underbrace{\begin{bmatrix}1\\1\\0\\0\\0\end{bmatrix}\begin{bmatrix}1\\0\\1\\0\\0\end{bmatrix}\begin{bmatrix}1\\0\\0\\1\\0\end{bmatrix}\begin{bmatrix}1\\0\\0\\0\\1\end{bmatrix}\begin{bmatrix}0\\1\\1\\0\\0\end{bmatrix}\begin{bmatrix}0\\1\\0\\1\\0\end{bmatrix}\begin{bmatrix}0\\1\\0\\0\\1\end{bmatrix}\begin{bmatrix}0\\0\\1\\1\\0\end{bmatrix}\begin{bmatrix}0\\0\\1\\0\\1\end{bmatrix}\begin{bmatrix}0\\0\\0\\1\\1\end{bmatrix}}_{0.625},\;
\underbrace{\begin{bmatrix}1\\1\\1\\1\\0\end{bmatrix}\begin{bmatrix}1\\1\\1\\0\\1\end{bmatrix}\begin{bmatrix}1\\1\\0\\1\\1\end{bmatrix}\begin{bmatrix}1\\0\\1\\1\\1\end{bmatrix}\begin{bmatrix}0\\1\\1\\1\\1\end{bmatrix}}_{0.3125}
\right\},
$$

$$
C_b = \left\{
\underbrace{\begin{bmatrix}1\\0\\0\\0\\0\end{bmatrix}\begin{bmatrix}0\\1\\0\\0\\0\end{bmatrix}\begin{bmatrix}0\\0\\1\\0\\0\end{bmatrix}\begin{bmatrix}0\\0\\0\\1\\0\end{bmatrix}\begin{bmatrix}0\\0\\0\\0\\1\end{bmatrix}}_{0.1325},\;
\underbrace{\begin{bmatrix}1\\1\\1\\0\\0\end{bmatrix}\begin{bmatrix}1\\1\\0\\1\\0\end{bmatrix}\begin{bmatrix}1\\1\\0\\0\\1\end{bmatrix}\begin{bmatrix}1\\0\\1\\1\\0\end{bmatrix}\begin{bmatrix}1\\0\\1\\0\\1\end{bmatrix}\begin{bmatrix}1\\0\\0\\1\\1\end{bmatrix}\begin{bmatrix}0\\1\\1\\1\\0\end{bmatrix}\begin{bmatrix}0\\1\\1\\0\\1\end{bmatrix}\begin{bmatrix}0\\1\\0\\1\\1\end{bmatrix}\begin{bmatrix}0\\0\\1\\1\\1\end{bmatrix}}_{0.625},\;
\underbrace{\begin{bmatrix}1\\1\\1\\1\\1\end{bmatrix}}_{0.0625}
\right\},
$$

$$d = 0.5, \quad \alpha = 6.26\%$$

$$C_b = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \underbrace{}_{1/2}, \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} \underbrace{}_{1/2} \right\},$$

$$d = 0.5, \quad \alpha = 12.5\%.$$

7) (2,5), the equation can be derived, as shown at the bottom of the page 15

8) (3,5), the equation can be derived, as shown at the top of the previous page

9) (4,5), the equation can be derived, as shown at the top of the previous page

10) (5,5), the equation can be derived, as shown at the top of the previous page.

REFERENCES

[1] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979.

[2] M. Naor and A. Shamir, "Visual cryptography," in *Advances in Cryptology-EUROCRYPT* (Lecture Notes in Computer Science), vol. 950. Berlin, Germany: Springer, 1995, pp. 1–12.

[3] H. Koga and E. Ueda, "Basic properties of the (t, n)-threshold visual secret sharing scheme with perfect reconstruction of black pixels," *Des., Codes Cryptogr.*, vol. 40, no. 1, pp. 81–102, Jul. 2006.

[4] J. Weir and W. Q. Yan, *Visual Cryptography and Its Applications*. Frederiksberg, Denmark: Ventus Publishing ApS, 2012.

[5] Z. Zhou, C.-N. Yang, S.-R. Cai, and D.-S. Wang, "Boolean operation based visual cryptography," *IEEE Access*, vol. 7, pp. 165496–165508, 2019.

[6] P. Li, J. Ma, L. Yin, and Q. Ma, "A construction method of (2, 3) visual cryptography scheme," *IEEE Access*, vol. 8, pp. 32840–32849, 2020.

[7] G. Ateniese, C. Blundo, A. D. Santis, and D. R. Stinson, "Extended capabilities for visual cryptography," *Theor. Comput. Sci.*, vol. 250, nos. 1–2, pp. 143–161, Jan. 2001.

[8] T.-H. Chen and K.-H. Tsao, "User-friendly random-grid-based visual secret sharing," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 11, pp. 1693–1703, Nov. 2011.

[9] P.-L. Chiu and K.-H. Lee, "User-friendly threshold visual cryptography with complementary cover images," *Signal Process.*, vol. 108, pp. 476–488, Mar. 2015.

[10] L. Bao and Y. Zhou, "Image encryption: Generating visually meaningful encrypted images," *Inf. Sci.*, vol. 324, pp. 197–207, Dec. 2015.

[11] S. Shivani and S. Agarwal, "Progressive visual cryptography with unexpanded meaningful shares," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 12, no. 4, pp. 50:1–50:24, 2016.

[12] S. Shivani, "VMVC: Verifiable multi-tone visual cryptography," *Multimedia Tools Appl.*, vol. 77, no. 5, pp. 5169–5188, Mar. 2018.

[13] S. Shivani and S. Agarwal, "Novel basis matrix creation and preprocessing algorithms for friendly progressive visual secret sharing with space-efficient shares," *Multimedia Tools Appl.*, vol. 76, no. 6, pp. 8711–8744, Mar. 2017.

[14] K.-H. Lee and P.-L. Chiu, "Sharing visual secrets in single image random dot stereograms," *IEEE Trans. Image Process.*, vol. 23, no. 10, pp. 4336–4347, Oct. 2014.

[15] Z. Fu, Y. Cheng, and B. Yu, "Visual cryptography scheme with meaningful shares based on QR codes," *IEEE Access*, vol. 6, pp. 59567–59574, 2018.

[16] Z. Zhou, G. R. Arce, and G. Di Crescenzo, "Halftone visual cryptography," *IEEE Trans. Image Process.*, vol. 15, no. 8, pp. 2441–2453, Aug. 2006.

[17] C.-N. Yang, C.-H. Wu, Z.-X. Yeh, D.-S. Wang, and C. Kim, "A new sharing digital image scheme with clearer shadow images," *Comput. Standards Interfaces*, vol. 51, pp. 118–137, Mar. 2017.

[18] P. Singh, B. Raman, and M. Misra, "A (n, n) threshold non-expansible XOR based visual cryptography with unique meaningful shares," *Signal Process.*, vol. 142, pp. 301–319, Jan. 2018.

[19] P.-L. Chiu and K.-H. Lee, "Efficient constructions for progressive visual cryptography with meaningful shares," *Signal Process.*, vol. 165, pp. 233–249, Dec. 2019.

[20] R.-Z. Wang and S.-F. Hsu, "Tagged visual cryptography," *IEEE Signal Process. Lett.*, vol. 18, no. 11, pp. 627–630, Nov. 2011.

[21] X. Wu and W. Sun, "Improved tagged visual cryptography by random grids," *Signal Process.*, vol. 97, pp. 64–82, Apr. 2014.

[22] Y.-H. Chen, C.-S. Chan, P.-Y. Hsu, and W.-L. Huang, "Tagged visual cryptography with access control," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Chengdu, China, Jul. 2014, pp. 14–18.

[23] C.-N. Yang, "New visual secret sharing schemes using probabilistic method," *Pattern Recognit. Lett.*, vol. 25, no. 4, pp. 481–494, 2004.

[24] P.-L. Chiu and K.-H. Lee, "A simulated annealing algorithm for general threshold visual cryptography schemes," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 992–1001, Sep. 2011.

[25] T.-H. Chen and K.-H. Tsao, "Threshold visual secret sharing by random grids," *J. Syst. Softw.*, vol. 84, no. 7, pp. 1197–1208, Jul. 2011.

[26] R. Ito, H. Kuwakado, and H. Tanaka, "Image size invariant visual cryptography," *IEICE Trans. Fundamentals Electron., Commun. Comput. Sci.*, vol. 82, no. 10, pp. 2172–2177, 1999.

[27] P.-L. Chiu and K.-H. Lee, "A probabilistic tagged visual cryptography," in *Proc. ICEAS*, Hong Kong, 2017, pp. 212–223.

[28] P. A. Eisen and D. R. Stinson, "Threshold visual cryptography schemes with specified whiteness levels of reconstructed pixels," *Des., Codes Cryptogr.*, vol. 25, no. 1, pp. 15–61, 2002.

[29] C.-N. Yang, C.-C. Wu, and D.-S. Wang, "A discussion on the relationship between probabilistic visual cryptography and random grid," *Inf. Sci.*, vol. 278, pp. 141–173, Sep. 2014.

PEI-LING CHIU received the Ph.D. degree in information management from National Taiwan University, Taipei, in 2007. In 2008, she joined Ming Chuan University, Taipei. She is currently a Professor with the Department of Risk Management and Insurance. Her research interests include visual cryptography, wireless sensor networks, insurance technology, and optimizing technologies.

KAI-HUI LEE received the Ph.D. degree in electronic engineering from the National Taiwan University of Science and Technology, Taipei, in 2002. Since 2003, he has been with Ming Chuan University, Taipei. He is currently a Professor with the Department of Computer Science and Information Engineering. His research interests include visual cryptography, wireless networks, the Internet of Things, and network resource managements.

● ● ●