New Capabilities of Visual Cryptography

Maged Hamada Ibrahim[1]

**[1] Department of Electronics, Communications and Computers Engineering,
Faculty of Engineering,
Helwan University,
Helwan, Cairo, Egypt**

Abstract

Visual cryptography schemes (VCS) consider the problem of encrypting and sharing images (printed text, hand written notes, pictures, etc.) in a perfectly secure way which can be decrypted directly by the human visual system. A k out of n VCS for is a technique by which a secret image (SI) is shared among n users called share-holders, when k or more users stack their shares (transparencies) together, the SI becomes visible, while k-1 or less users gain no information about SI. This paper presents a strategy by which more than one SI can be shared simultaneously among the n users using exactly the same shares required for the existing (*k*, *n*)-VCS without any extra pixel expansions or any other type of overheads over that required for the already existing schemes. The only requirement is a little effort in orienting and distributing the pixels among the shares. The strategy reduces the number of distributed shares required for multiple SI's and at the same time preserves the perfect security of the existing schemes. Moreover, our strategy enables the visual sharing of a small animated scene.

Keywords: *Cryptography, Secret sharing, Visual cryptography, image processing, Hamming weight, Contrast, Access structures, Threshold secret sharing.*

1. Introduction

Visual cryptography was introduced by Naor and Shamir [1, 2]. They assumed that the image consists of a collection of black and white pixels. Each pixel appears in *n* versions called shares, one for each transparency. Each share is a collection of *m* black and white sub-pixels. The resulting structure can be described by an *nm* Boolean matrix $S= s(i,j)$ where $s(i,j)=1$ if and only if the *j*-th sub-pixel in the *i*-th transparency is black and $s(i,j)=0$ if and only if the *j*-th sub-pixel in the *i*-th transparency is white (transparent). Therefore the gray level of the combined share, obtained by stacking the transparencies $i_1,..., i_s$, is proportional to the Hamming weight $w(V)$ of the *m*-vector $V=OR(r(i_1),...,r(i_s))$ where $r(i_1),...,r(i_s)$ are the rows of *S* associated with the transparencies we stack. We emphasize that the operation is an OR not an XOR. This gray level is interpreted by the visual system of the users as black or as white in accordance with some rule of contrast.

Visual Cryptography Schemes for general access structures were introduced in [3], given a set **P** of *n* participants, it is possible to encode a secret image SI into *n* shadow images called shares, where each participant in **P** receives one share. Certain qualified subsets of participants (Γ_{qual}) can visually recover the secret image, but other, forbidden subsets of participants (Γ_{forb}) have no information (in the information-theoretic sense) on SI. In implementing Visual Cryptography schemes it would be useful to conceal the existence of the secret message, namely, the shares given to participants in the scheme should not look as a random bunch of pixels, but they should be innocent looking images [4]

1.1 Related Work

A scheme to stand against sabotage attacks was given in [5]. In [6] visual cryptography techniques based on Interferometric encryption were presented. In [7], two binary secret images are encrypted as in without pixel expansion and the shares are embedded into the half-toned image to avoid the suspicion of the intruders. These halftone shares are concurrently error diffused to give visually pleasing effect. During decryption process, the halftone shares are directly stacked to reveal the first secret and the second secret could be revealed by stacking one share image and the other with a rotation angle of 180 degrees.

In [8], the authors studied the cheating problem in VC and extended VC. They considered the attacks of malicious adversaries who may deviate from the scheme in any way. They presented three cheating methods and applied them on attacking existent VC or extended VC schemes. They improved one cheat-preventing scheme and proposed a generic method that converts a VCS to another VCS that has the property of cheating prevention. The overhead of the conversion is near optimal in both contrast digressions and pixel expansion. Other techniques for the same purpose were presented in [10, 11, 12, 13, 14, 17]. In [9] using Fourier techniques derived from Fourier Optics concepts, the idea is to enhance and exploit the quasi periodicity of the shadow images, composed by a random distribution of black and white patterns on a periodic sampling grid. The advantage is to speed up the security control or the access time to the message, in particular in the cases of a small pixel size or of large numbers of pixels. The work in [13] shows a real perfect contrast VCS such that the black and white pixels are perfectly reconstructed within finite runs, no matter what VCS (perfect black or non-perfect black) is used. Visual cryptographic techniques that deal with the sharing of color images were introduced in [15, 16, 18, 19, 20, 21]. In [22] the problem of precise alignment of printed and scanned visual cryptography shares was studied.

1.2 Applications of Visual Cryptography

Other than One-Time-Pad encryption, the impact of visual cryptography on designing electronic voting schemes was introduced by David Chaum [28]. The issue of getting the voter's trust is resolved by using ideas of visual cryptography [27]. Other wide range of visual cryptography applications appeared recently. In [23] A core banking application was suggested where the customer has to present the share during all of his transactions. This share is stacked with the first share to get the original signature. The correlation method is used to take the decision on acceptance or rejection of the output and authenticate the customer. Applications for watermarking appeared in [24, 25, 26] and others.

1.3 Our Contribution

In this paper, we show a new strategy that employs the original (*k*, *k*)-VCS and the (*k*, *n*)-VCS described in [1], not only to encode one secret image (SI), but to encode *k* SI's simultaneously utilizing the same shares with no extra pixel expansions or any other type of overheads over that required for the already existing VCS, we will show that if the (*k*, *k*)-VCS encodes a secret image SI_0 of size cr pixels (where *c* is the number of columns and *r* is the number of rows), our strategy enables the encoding of *k* secret images SI_j ($1 \leq j \leq k-1$), each image is of size $(c-1)r$ pixels, in addition to SI_0. We will also show that for the (*k, n*)-VCS it is possible to encode *k* secret images SI_j ($1 \leq j \leq k-1$), each of size $(c - jg)r$ for some parameter *g*, in addition to SI_0. The images are revealed one after the other by a proper sliding (shifting) of the shares one or more column to the right (sliding to the right is not a restriction at all, it may be to the left, upward, downward or even diagonally), however, for clarity, sliding to the right will be used throughout this work. The consequent revealing of the secret images enables the sharing of a small animated scene.

2. Our Strategy

We will start by showing how to apply our strategy to the (2, 2)-VCS with the well-known collection of matrices C_0 (representing the shares for a white pixel) and C_1 (representing the shares for a black pixel) given by:

$$C_0 = \{\begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix} \text{ or } \begin{pmatrix}0 & 1\\ 0 & 1\end{pmatrix}\};\ C_1 = \{\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} \text{ or } \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}\}$$

Our objective is that we have two secret images, SI_0 of size cr pixels and SI_1 of size $(c-1)r$ pixels, and we want to encode both images simultaneously on two shares (transparencies), the size of each transparency is cr pixels and each pixel is split (or expanded) into two subpixels as already known. Our strategy for this case is described as shown in fig.1. As a result, when the two shares are stacked together as shown in fig.2(a), the secret image SI_0 becomes visual to the shareholders. When sliding share 2 one column to the right as shown in fig.2(b), the secret image SI_1 becomes visual and SI_0 totally disappears.

Now we step to the (3, 3)-VCS, since k = 3, it is possible to encode 3 secret images: SI_0 of size cr pixels, SI_1 of size $(c-1)r$ pixels and SI_2 of size $(c-1)r$ pixels, simultaneously on the same three transparencies. The two basis matrices S_0 and S_1 that one may utilize for the (3, 3)-VCS are given by:

$$S_0 = \begin{pmatrix}0 & 0 & 1 & 1\\ 0 & 1 & 0 & 1\\ 0 & 1 & 1 & 0\end{pmatrix} \qquad S_1 = \begin{pmatrix}1 & 1 & 0 & 0\\ 1 & 0 & 1 & 0\\ 1 & 0 & 0 & 1\end{pmatrix}$$

```
Input:
        • Two secret images $SI_0$ and $SI_1$.
        • The two Basis matrices $S_0$ and $S_1$ of the (2, 2)-VCS.
Procedure:
    For $j$ = 1 to $c$ do:
        • Design column number $j$ of share 1 and column number $j$ of share 2 to give the required pixels colors of column number $j$ in $SI_0$.
        • Slide (shift) share 2, one column to the right, such that column number $j$ in share 2 (which is already designed) coincides with column number $j$+1 of share 1.
        • Now, based on the orientation of column number $j$ in share 2 design column number $j$+1 in share 1 to give the required pixels colors of column number $j$ in $SI_1$. (Break this step whenever $SI_1$ is completed).
        • Undo sliding.
Output: Two shares, each player is delivered one share.
```

Fig. 1 The strategy for the (2, 2, 2)-MVCS

The strategy to encode 3 SI's simultaneously is shown in fig.3.

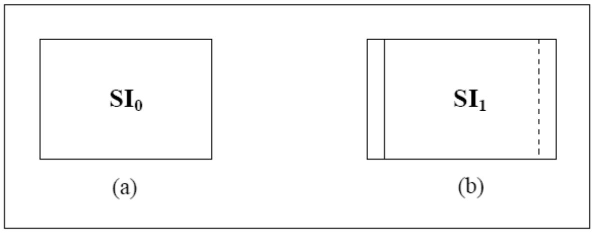

Fig. 2 The visualization of SI_0 and SI_1 for the (2, 2, 2)-MVCS

Input:

- The three secret images SI_0, SI_1 and SI_2.
- The basis matrices of the (3, 3)-VCS.

Procedure:

For j = 1 to c do:

- Column number j in shares 1, 2 and 3 are designed to give the required pixels colors of column number j in SI_0.
- Shares 2 and 3 are both shifted one column to the right such that, column number $j+1$ in share 1, column number j in share 2 and column number j in share 3 coincide.
- Based on the orientation of column number j in shares 2 and 3, column number j+1 in share 1 is designed to give the required pixels colors of column number j in SI_2. (Break this step whenever SI_2 is completed).
- Undo sliding of share 3 such that column number j+1 in share 1 (already designed), column number j in share 2 (already designed) and column number j+1 in share 3 all coincide.
- Design column number j+1 in share 3 to give the required pixels colors of column number j in SI_1. (Break this step whenever SI_1 is completed).
- Undo all shifts.

Output: Three shares, each player is delivered one share.

Fig3. The strategy for the (3, 3, 3)-MVCS

As a result, when the three shares are stacked together as shown in fig.4 (a), the secret image SI_0 becomes visual. When share 2 is shifted one column to the right as shown in fig.4 (b), the secret image SI_1 becomes visual. When share number 3 is shifted one column to the right as shown in fig.4(c), the secret image SI_2 becomes visual. Of course only one secret image is visible at a time with no trace of the other two.

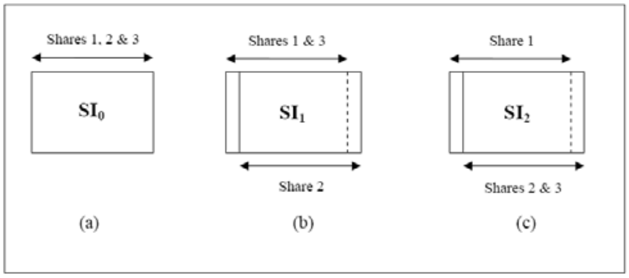

Fig. 4 The visualization of SI_0, SI_1 and SI_2 for the (3, 3, 3) – MVCS

3. The Strategy for the (*M*, *k*, *k*)-MVCS

Defenition 1: Given C_0 and C_1 of the (k, k)-VCS, where C_0 = {the collection of matrices obtained by permuting the columns of the basis matrix S_0} and C_1= {the collection of matrices obtained by permuting the columns of the basis matrix S_1}, the (M, k, k)-MVCS is a scheme by which: it is possible to simultaneously distribute ($M \leq k$) SI's using the same k shares of the (k, k)-VCS with no extra pixel expansion or any other type of overheads. The size of SI_0 is cr pixels and the size of each SI_j ($1 \leq j \leq k$-1) is $(c-1)r$ pixels.

In this section we will generalize our strategy for the general (k, k)-VCS. Again, utilizing the basis matrices S_0 and S_1 of the (k, k)-VCS, we have k SI's that we want to encode on the k transparencies; these are: SI_0 of size cr pixels and SI_j $(1 \leq j \leq k-1)$ each of size $(c-1)r$ pixels. The strategy can be generalized as shown in fig.5.

Input:

- The k secret images SI_j $(1 \leq j \leq k-1)$ to be encoded.
- The two basis matrices of the (k, k)-VCS.

Procedure:

For j = 1 to c do:

- Design column number j in all shares to give the required pixels colors of column number j in SI_0.
- Except shares 1…j, slide all shares one column to the right such that, column number j+1 in share 1…j, column number j in shares j+1…k coincide.
- Design column number j+1 in share 1…j such that, in corporation with column number j in shares j+1…k give the required pixels colors of column number j in SI_j. (Break this step for any completed SI_j).
- Undo all shifts.

Output: k shares, each player is delivered one share.

Fig.5 The strategy for the (k, k, k)-MVCS

As a result of the above strategy, the secret image SI_0 becomes visible when all shares are stacked together. The secret image SI_j $(1 \le j \le k-1)$ becomes visual if shares $j+1,\ldots,k$ are shifted one column to the right as shown in fig. 6

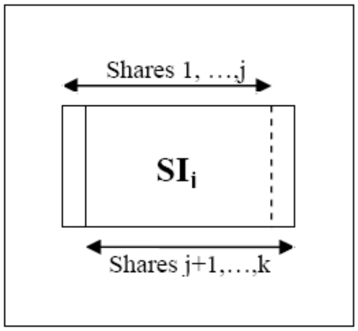

Fig.6 The visualization of SI_j for the (k, k, k)-MVCS

4. The Strategy for the (*M*, *k*, *n*)-MVCS

Definition 2: Given C_0 and C_1 of the (k, n) – VCS, where C_0 = {the collection of matrices obtained by permuting the columns of the basis matrix S_0} and C_1= {the collection of matrices obtained by permuting the columns of the basis matrix S_1}, the (M, k, n)-MVCS is a strategy by which it is possible to simultaneously distribute ($M \le k$) SI's using the same n shares of the (k, n)-VCS with no extra pixel expansion or any other type of overheads. The size of SI_0 is cr pixels and the size of each SI_j ($1 \le j \le k$-1) is $(c - jg)r$ pixels where $g = \lceil n/(k-1) \rceil$ represents the number of groups.

4.1 The strategy for the (2, 2, *n*)-MVCS

First, consider the (2, 3)-VCS, given its basis matrices S_0 and S_1, it is possible to encode two SI's simultaneously. One may try to directly apply the strategy described in the previous section: Designing the first column in all shares to give the required pixels colors of the first column in SI_0, shifting shares 2, 3 one column to the right and designing column number 2 in share 1 to give the required pixels colors of column number 1 in SI_1…etc. This attempt will case the two SI's to be visible at the same time when stacking share 2 and share 3 together which is not desired at all. The reason is that, shares 2 and 3 have the same orientation for both SI's, it is required that the orientation of the shares is always unique per secret image. Due to the importance of such requirement we are going to call it 'Orientation Uniqueness' of the shares. In the case of the (2, 3)-VCS we cannot run away of shifting share 3 one more column to the right to guarantee orientation uniqueness, which means losing one more column in the size of SI_1 to be $(c-2)r$ pixels. We conclude that for the (2, n)-VCS we can encode two SI's, SI_0 of size cr and SI_1 of size $(c-n)r$. Numbering the shares 1… n; SI_0 is visible when two or more of the n shares are stacked together, SI_1 is visible when two or more shares are stacked together and the first column of share j coincide with column number u of share $(j-u+1$ for any $(1 \le j, u \le n)$. A possible strategy for the (2, n)-VCS, utilizing its basis matrices S_0 and S_1, is shown in fig.7.

4.2 The strategy for the (3, 3, *n*) – MVCS

In order to attain '*Orientation Uniqueness*' for the (3, n)-VCS the n shares are divided into $g=n/2$ groups each group consisting of 2 shares. We will assume that n is divisible by 2, this is not a restriction at all; however it just simplifies the discussion. Next, every share is marked with its group number and its number within the group. SI_0 of size cr pixels is visible when any 3 of the n shares are stacked together. Imagine that the n shares are stacked together and except group 1, group j is shifted j-1 columns to the right (for all $2 \le j \le g$), this is the position of the shares that makes SI_1 of size $(c-g)r$ pixels becomes visible, of course, 3 or more shares are enough, but in the specified position. Now, share number 2 in group j is shifted j more columns to the right, this is the position of the shares that makes SI_2 of size $(c-2g)r$ pixels becomes visible, this is illustrated in fig.8. The strategy to encode 3 SI's for the (3, n)-VCS has much similarities to the (2, n)-VCS hence we skip to the general (k, n)-VCS.

```
Input:
        • The two secret images to be encoded.
        • The basis matrices of the (2, n)-VCS.
Procedure:
    1-  Imagine that all the n shares numbered 1…n are stacked
        together (this is of course for simplicity).
    2-  For i = 1 to c do:
        • Design column number i in all shares to give the required
          pixels colors of column number i in $SI_0$.
        • Except share 1, share j (for all $2 \le j \le n$) is shifted (j-1)
          columns to the right.
        • For all ($1 \le j \le n$), design column number (n-j+i) of
          share j to give the required pixels colors of column number
          i in $SI_1$.
Output: n shares, each player is delivered one share.
```

Fig. 7 The strategy for the (2, 2, n)-MVCS

4.3. Generalization for the (*M, k, n*)-MVCS

Divide the *n* shares into $g = \lceil n/(k-1) \rceil$ groups each group contains at most *k-1* shares, this is important to guarantee the 'Orientation Uniqueness' requirement. Next, every share is marked with its group number and its number within the group. Let g_i be the number of shares in group *i*. SI_0 of size cr pixels is visible when any *k* or more shares are stacked together. Imagine all shares are stacked together and shares of group *i* are shifted *i-1* columns to the right (for all $2 \le i \le g$), this is the position of the shares that will visualize SI_1. To visualize SI_q $(2 \le q \le k-1)$ of size $(c-gq)r$ pixels, shares $q,\ldots,g_i$ in each group in addition to all the shares in the groups above them are shifted *i*-1 more columns to the right without undoing any previous shifts, if $q \ge g_i$ continue shifting share g_i. Again, since the strategy utilizes the basis matrices of the (*k, n*)-VCS, only *k* or more shares in their specified position are enough to visualize any of the SI's. The strategy is shown in fig. 9.

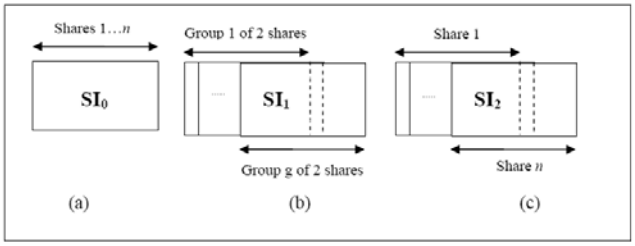

Fig. 8 The visualization of SI_0, SI_1 and SI_2 for the (3, 3, n)-MVCS

5. A Note on Security

It is obvious that our strategy does not tamper - at all - with the basis matrices used to distribute the shares among the share-holders; also each secret image is encoded independently. Again, we emphasize that the operation is an OR function not an XOR function. Thus our strategy preserves the security of the (*k*, *k*)-VCS and the (*k*, *n*)-VCS schemes described in [1]. We conclude that our strategy is secure.

6. The Upper Bound on *M*

Lemma: For the (*M*, *k*, *n*)-MVCS, let $M_{\max}$ be the maximum number of simultaneously distributed SI's then, $M_{\max} = k$.

Proof: In order to visualize any of the distributed SI's, any *k* shares must be enough since this is the threshold value common to all the SI's. Let's assume the contradiction, that it is possible to distribute $M_{\max}>k$ SI's using our strategy and using only *k* shares. For the (2, 2)-VCS; let's try to distribute 3 SI's as follows:

1- Design column number 1 in share 1 and share 2 to give the required pixel color of column number 1 of SI_0.
2- Slide share 2 one column to the right and design column number 2 in share 1 such that with column number 1 in share 2 gives the required pixel color of column number 1 in SI_1.
3- Now perform one more slide of share 2 and design column number 3 of share one such that with column number 1 in share 2 gives the required pixel color of column number 1 in SI_2.
4- undo sliding of share 2 one column and design column number 2 of share 2 such that with column number 3 of share 1 gives the required pixel color of column number 2 in SI_1.

At the end of step 4 we have columns 1, 2 and 3 of share 1 designed and columns 1 and 2 of share 2 designed.

5- Undo sliding of share 2 one more column. We have to design column number 2 of share 2 and column number 2 of share 1 to give the required pixel color of column number 2 in SI_0 which is impossible since these columns were already designed.

Step 5 shows that it is impossible to distribute more than 2 SI's for the (2, 2)-VCS by applying our strategy, hence for the (*M*, 2, 2)-MVCS, $M_{\max}$ =2. Therefore, in order to visualize 3 SI's it is required to stack 3 shares to give the required degree of freedom for the design. In order to visualize $M_{\max}$ SI's it is required to stack $M_{\max}$ shares and hence $M_{\max}$ is upper bounded by the minimum number of shares required to visualize any of the distributed SI's which is *k*, this completes the proof.

7. Comparison and Evaluation

In order to share *s* secret images using the conventional (*k*, *n*)-VCS with $k \le n$ and without applying our strategy, it is required *sn* shares distributed among the *n* share-holders, each share-holder holds *s* shares, one for each secret image. Our strategy serves in reducing the number of shares required to be only *n* shares for $s \le k$. If $2k \ge s > k$, we can apply our strategy to encode *k* secret images using the (*k*, *n*)-VCS basis matrices, this requires *n* shares, then to encode the remaining *s-k* secret images, this requires another *n* shares. As a result, each share-holder holds 2 shares totaling 2*n* distributed shares instead of 2*kn* distributed shares. We conclude that for some integer *j* > 0 and using the (*k*, *n*)-VCS basis matrices it is possible to encode $s \leq jk$ secret images by distributing *jn* shares instead of *jkn* shares with the expense of a non-significant loss of several columns.

```
Input:
  • The k secret images to be shared.
  • The two basis matrices of the (k, n)-VCS.
Procedure:
  • For x = 1 to c do:
      ▪ All shifts are undone.
      ▪ Design column number x in all shares to give the required
        pixels colors of column number x in $SI_0$.
      ▪ For q = 1 to k-1 do:
          o For all $2 \leq i \leq g$ , Slide shares $q \ldots g_i$ in group i (in
            addition to all the shares in the groups above them) i-1
            more columns to the right where i is the group number.
          o Design column number x+1 of shares 1…q-1 in group g
            and all the columns coinciding with them to give the
            required pixels colors of column number x in $SI_q$. (Break
            this step for any completed $SI_q$).
Output: n shares, each player is delivered one share.
```

Fig.9 The strategy for the (k, k, n)-MVCS

8. Conclusions

Throughout this article we have introduced a new strategy by which, for the (*k*, *k*)-VCS and the (*k*, *n*)-VCS it is possible to encode *k* SI's simultaneously on the same shares, each secret image SI_j $(1 \leq j \leq k-1)$ is of size $(c-1)r$ pixels in the case of (*k*, *k*)-VCS and of size $(c-g.j)r$ pixels for some parameter *g* in the case of (*k*, *n*)-VCS where *c* is the number of columns and *r* is the number of rows of SI_0. Also, we have shown that, for some integer *j* > 0 and using the (*k*, *n*)-VCS basis matrices it is possible to encode $s \leq jk$ secret images by distributing *jn* shares instead of *jkn* shares with the cost of a non-significant reduction in the size of the original images.

References

[1] Naor M. and Shamir A., Visual cryptography, in Eurocrypt '94, Springer-Verlag LNCS Vol. 950, 1995, 1-12.

[2] M. Naor and A. Shamir, Visual Cryptography II: Improving the Contrast via the Cover Base. Theory of Cryptography Library, n., 1996.

[3] G. Ateniese, C. Blundo, A. De Santis, and D. R. Stinson, Visual Cryptography for General Access Structures, Information and Computation, Vol. 129, No. 2, (1996), pp. 86-106.

[4] G. Ateniese, C. Blundo, A. DeSantis, and D. R. Stinson, Extended Capabilities for Visual Cryptography, to appear in Theoretical Computer Science, 1999.

[5] Ingrid Biehl, Susanne Wetzel, "Traceable visual cryptography", ICIS 1997.

[6] S. S. Lee, J. C. Na, S. W. Sohn, C. Park, D. H. S. and S. J. Kim, Visual Cryptography based on an Interferometric Encryption Technique, ETRI Journal, Vol. 24, 5, 2002, 373-380.

[7] Anbarasi, L. J.; Vincent, M. J.; Mala, G. S. A., A novel visual secret sharing scheme for multiple secrets via error diffusion in halftone visual cryptography, International Conference on Recent Trends in Information Technology (ICRTIT), 2011, pp. 129 – 133.

[8] Chih-Ming Hu Wen-Guey Tzeng, Cheating Prevention in Visual Cryptography, IEEE transactions on image processing, vol. 16, issue 1, 2007, pp. 36 – 45.

[9] Jacques Machizaud, Pierre Chavel, and Thierry Fournel, Fourier-based automatic alignment for improved visual cryptography schemes, Opt. Express 19, 2011, pp. 22709-22722.

[10] Du-Shiau Tsai, Tzung-Her Chen, Gwoboa Horng, A cheating prevention scheme for binary visual cryptography with homogeneous secret images, Pattern Recognition, Volume 40, Issue 8, August 2007, pp. 2356-2366.

[11] Liu, F., Wu, C., Lin, X. Cheating immune visual cryptography scheme, IET Information Security 5 (1), 2011, pp. 51-59.

[12] Chang, C-C., Lin, C.-C., Tu, H.N., Safeguarding visual information using (t, n) verifiable secret shares, Journal of Computers 22 (2), 2011.

[13] Yang, C.-N., Wang, C.-C., Chen, T.-S, Visual cryptography schemes with reversing, Computer Journal 51 (6), 2008, pp. 710-722.

[14] YC Chen, G Horng, Share Authentication based Cheating Prevention in Naor-Shamir's Visual Cryptography, 電腦學刊 , 2011.

[15] Y. C. Hou, Visual cryptography for color images, Pattern Recognition, Vol. 36, 2003, pp. 1619-1629.

[16] A. Lazakidou, K. Siassiakos, (2, 3)-threshold visual cryptography for color images, ISCGAV'06 Proceedings of the 6th WSEAS International Conference on Signal Processing, Computational Geometry & Artificial Vision, 2006.

[17] M. N. kumar, D. S. Rao D. Sravanthi, A Novel Approach for Cheating Prevention through Visual Cryptographic Analysis, International Journal of Computer science and engineering Survey (IJCSES), 2(4), 2011, 123 – 131.

[18] Duo Jin, Wei-Qi Yan and Mohan S. Kankanhalli, "Progressive color visual cryptography", J. Electron. Imaging 14, 033019 (Aug 05, 2005)

[19] S. Kandar, A. Maiti, Visual Cryptography Scheme for Color Image Using Random Number with Enveloping by Digital Watermarking, IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 1, May 2011.

[20] Chin-Chen Chang, Chwei-Shyong Tsai, Tung-Shou Chen, Proceedings of Seventh International Conference on Parallel and Distributed Systems, 2000.

[21] Shyong Jian Shyu, Efficient visual secret sharing scheme for color images, Pattern Recognition, Volume 39, Issue 5, May 2006, pp. 866–880.

[22] Wei-Qi Yan, Duo Jin, Kankanhalli, M.S., Visual cryptography for print and scan applications, Proceedings of the International Symposium on Circuits and Systems, 2004

[23] Hegde, C.; Manu, S.; Shenoy, P. D.; Venugopal, K. R.; Patnaik, L. M.; Secure Authentication using Image Processing and Visual Cryptography for Banking Applications, 16th International Conference on Advanced Computing and Communications, ADCOM 2008.

[24] R. Hwang, A digital Image Copyright Protection Scheme Based on Visual Cryptography, Tambang Journal of science and Engineering, vol.3, No.2, pp. 97-106 (2000).

[25] Mahmoud A. Hassan, and Mohammed A. Khalili, Self Watermarking based on Visual Cryptography, World Academy of Science, Engineering and Technology (8) 2005.

[26] A. Sleit, and A. Abusitta, “A Watermark Technology Based on Visual Cryptography”, in the Proceedings of the 10th World Multi-Conference on Systemics, Cybernetics and Informatics, Vol. 1, 2006, pp. 227-238.

[27] Ghayab, Hadi Ratham (2010) Visual Cryptography for E-Voting by Using Fingerprint Technique. Master thesis, Universiti Utara Malaysia.

[28] Chaum, D.: Secret-Ballot Receipts and Transparent Integrity. Better and Less-costly Electronic Voting at Polling Places. http://www.vreceipt.com/article.pdf

Maged Hamada Ibrahim Received his BSc in communications and computers engineering from Helwan University, Cairo; Egypt. Received his MSc and PhD in Cryptography and Communications security systems from Helwan University in 2001 and 2005 respectively. Currently, working as an assistant professor and also joining several network security projects in Egypt. His main interest is Cryptography and network security. More specifically, working on the design of efficient and secure cryptographic algorithms. In particular, secure distributed multiparty computations.