Pacific Graphics 2014
J. Keyser, Y. J. Kim, and P. Wonka
(Guest Editors)

Volume 33 (2014), Number 7

Structure Aware Visual Cryptography

Bin Liu[1] Ralph R. Martin[2] Shi-Min Hu[1]

[1]TNList, Tsinghua University, Beijing [2]Cardiff University

Abstract

Visual cryptography is an encryption technique that hides a secret image by distributing it between some shared images made up of seemingly random black-and-white pixels. Extended visual cryptography (EVC) goes further in that the shared images instead represent meaningful binary pictures. The original approach to EVC suffered from low contrast, so later papers considered how to improve the visual quality of the results by enhancing contrast of the shared images. This work further improves the appearance of the shared images by preserving edge structures within them using a framework of dithering followed by a detail recovery operation. We are also careful to suppress noise in smooth areas.

Categories and Subject Descriptors (according to ACM CCS): I.3.8 [Computer Graphics]: Applications—Cryptography

1. Introduction

Visual cryptography (VC) is a scheme for sharing secrets in the form of binary images. It encodes the target visual information into several carrier images which are printed on transparencies. On stacking these carriers together, and looking through them, the hidden image becomes visible. One advantage of VC is that decryption can be performed by the human visual system, without the need for computational assistance. VC can be applied to various areas such as copyright protection [HH05, Hwa00], watermarking [HC00], and secret message sharing [CY11].

In basic VC encryption, the input is a *binary* image to be transmitted while the output is a set of seemingly random binary images. If we denote the white (i.e. transparent) pixels by 1, and black (i.e. opaque) pixels by 0, then stacking the transparencies upon which the carriers are printed performs a pixel-wise **AND** operator to them, revealing the input image. At the same time, various additional black pixels are also present; these are necessary to disguise the input image. In the combined output image, the input image thus appears against a gray background, whose gray level is determined by the proportion of extra black pixels used. In general, the gray becomes darker and darker as more and more transparencies are superimposed. An illustration of VC is shown in Figure 1.

More advanced methods, referred to as extended VC, utilize *grayscale* natural secret and carrier images as input, but again produce encoded *black-and-white* carrier output [ZADC06, WADC09, LW11], using patterning dithering [Lim69] or an error-diffusion-like [FS75] algorithm to construct the result. It is well known that dithering tends to blur images and discards high-frequency details, such as sharp edges and textures, as Figure 1 shows.

In this paper, we show how to improve the appearance of the output carrier images by preserving edge structures within them. This is done using a framework of dithering followed by a series of operations to recover the fine details via optimization. While information leakage between carriers may arise in this process, it can be suppressed by adding an extra term to the objective function which is optimized.

A basic VC scheme uses just two carrier images. In a (k,n) VC scheme, n carrier images are used. Any superposition of k or more of them shows the secret image, while any superposition of $k-1$ or fewer does not. Such schemes are discussed in the related work section. In this paper, we focus purely on the graphical aspects of the problem, and use the same cryptographic approach used in earlier work. Thus, for simplicity, in the rest of the paper we usually take a $(2,2)$ VC scheme as an example. Our algorithm can be readily generalized to arbitrary (k,n) VC schemes using the method given in [LW11] as a pre-process. However, as discussed later, we adopt the encoding framework proposed by [NS95], which leads to loss of contrast in the output carriers and the stacked images. In general, the range of contrast is more restricted.

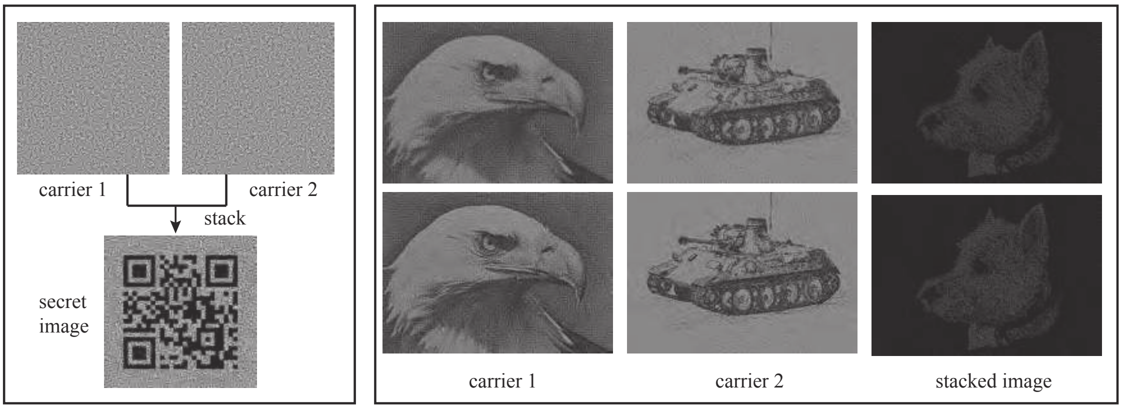


Figure 1: *Left: example of basic VC. Right: example of extended VC; top: result using the method in [LW11]: many details are lost. Bottom: our algorithm overcomes this problem and produces a structure aware result.*

The remainder of this paper is organized as follows: in Section 2 we briefly review related work. Section 3 describes our structure-aware VC algorithm in detail. Section 4 presents results and compares them to those obtained by previous methods. Section 5 draws some conclusions.

2. Related Work

The concept of visual cryptography was originally devised by Naor and Shamir [NS95], who also gave an algorithm to distribute the secret image amongst seemingly random binary patterns. Their approach, based on subpixel encoding, has been adopted in many subsequent papers. However, in the original approach, the shared carriers form random patterns lacking visual information, which can lead to suspicion of data hiding. Later papers have considered how to hide the secret image in natural looking carriers corresponding to further input images.

Ateniese et al. first discussed and implemented such an extended VC scheme, using hypergraph colourings to ensure that the carrier images are also meaningful after encryption [ABSS01]. In this case, the inputs are binary images. Nakajima and Yamaguchi improved upon this approach to handle grayscale input images [NY02]. Zhou et al. proposed a halftone visual cryptography scheme to achieve VC via halftoning techniques: a secret image is disguised into a pair of complementary images (i.e. ones in which black pixels in one image are white in the other and vice versa) derived from a single carrier image [ZADC06]. While this method does not suffer from contrast loss, the results are rather noisy. Wang et al. [WADC09] proposed methods for halftone visual cryptography. They use extra black pixels in each carrier to hide the opposite carrier image, a method we also adopt. They also consider how to reduce the number of extra black pixels to increase the contrast of the shares. However, the cost of doing so is that the visual content of each carrier image leaks into other output carrier images.

Liu and Wu proposed a series of methods that split the secret image into random shares and embeds them into meaningful carriers [LW11]. Complex permutations of carriers are considered in their paper, which again focuses on security and contrast. In their work, dithering is used to convert each image block to binary; this process ignores structures such as edges. Our work is an improvement in that we arrange the black and white pixels to reflect the structure of the original image.

Hou was the first to study VC encryption for color images. However, while the shared carriers are color images, they are meaningless [Hou03]. Kang et al. gave a color VC scheme based on error diffusion and visual information pixel synchronization to provided meaningful colored carriers [KAL11].

Several recently published books summarize the latest developments in VC [CY11, SLY14, LY14].

Various other graphics applications also aim to hide information in images. Mitra et al. considered 'emerging images', images of 3D objects that are difficult for a machine to detect and can only be recognized holistically by humans [MCL*09]. One application is as 'Captcha' images, to check that data is being entered onto a website by a human rather than a software robot. Chu et al. showed how to create 'camouflage images' using a texture synthesis method, in order to hide various objects in background images [CHM*10]. This has applications to puzzles and games; they could also again be used for web entry verification. A key difference of VC from such applications is that there must be no way to determine the hidden image from a *single* carrier. Recently, Chu et al. proposed an approach to generate a 'Halftone QR code', a kind of QR code

Secret image color (input)	white				black			
Carrier 1 color (input)	white	black	white	black	white	white	black	black
Carrier 1 block (output)	□■ □■	■□ ■■	□■ □■	□■ ■■	■□ □■	■□ □■	□■ ■■	□■ ■■
Carrier 2 color (input)	white	white	black	black	white	black	white	black
Carrier 2 block (output)	□■ ■□	■□ ■□	□■ ■■	□■ ■■	□■ ■□	□■ ■■	■□ ■□	■□ ■■
Stacked result	□■ ■■	■□ ■■	□■ ■■	□■ ■■	■■ ■■			

Figure 2: *Construction of a $(2,2)$ VC scheme. Each pixel is split into a 2×2 block of subpixels. The two subpixels in the top row represent the secret information as either a white-black, or black-white, pair. The bottom-left subpixel is the information pixel for carrier 1 and the mask pixel for carrier 2. The bottom-right subpixel is the mask pixel for carrier 1 and the information pixel for carrier 2.*

whose overall appearance is similar to a given image while a barcode reader can still decode the information embedded within it [CCLM13]. VC is different from this work in that decoding is done by a human, not a machine.

Structure-aware image manipulation aims to preserve edges and textures consisting of high-frequency patterns after image processing. It has many applications, and was applied to digital halftoning by Pang et al. [PQW*08]. They first randomly color pixels black and white to provide the correct global average gray level, and then swap pairs of black and white pixels to optimize structural similarity (measured in terms of SSIM [WBSS04]) and tone similarity (in terms of intensity difference of blurred inputs) between the halftone image and the grayscale input image. Their results preserve sharp edges and texture details, to which the human visual system is sensitive. Their method uses simulated annealing method, so is slow. Chang et al. solve the structure preservation problem differently [CAO09], using a standard error-diffusion pipeline in which uniform thresholding is replaced by adaptive threshold modulation based on an anisotropic Gabor filter. Fixed error diffusion coefficients are replaced by parameterized Gaussian filters. This approach achieves similar quality to Pang's approach but runs much faster. The contrast-aware halftoning method proposed by Li and Mould [LM10] modifies the error diffusion step. The basic idea is to place more black pixels in dark areas and more white pixels in light areas, helping to retain edges in the input images.

Previous VC work has mainly emphasised improving the quality of the shared carriers by increasing their contrast. Instead, our work exploits similar techniques to those in [PQW*08] to better preserve edges and textures.

3. Structure Aware Visual Cryptography Algorithm

We start this section by briefly summarising prior VC algorithms upon which our algorithm is based. We then describe our structure-aware dithering approach for VC, and how a coherence constraint can be used to reduce noise and prevent information leakage.

3.1. Review of Visual Cryptography

The basis of most VC algorithms is to map each pixel of the input carrier images into a block containing a certain number of pixels (referred to as *subpixels*) in the output carriers [NS95]. The input carrier and secret images are, in the simplest case, binary images. Figure 2 gives an example where each input pixel is mapped to a 2×2 block of subpixels in output; thus the output carrier images have twice the width and height of the input images.

Consider a particular carrier image. Each 2×2 block holds 3 kinds of information, stored in different subpixels. One subpixel, the *carrier information pixel* (CIP), holds the visual information for this carrier. Another, the *mask pixel* (MP), corresponds to the position in the block of the other carrier's CIP. It is always set to black. When stacking the carriers, in each 2×2 block, the CIP of carrier 1 is masked by the MP of carrier 2, and the CIP of carrier 2 is masked by the MP of carrier 1, so the CIPs always appear black in the stacked result.

The remaining two subpixels in each block, the *secret information pixels* (SIPs), hold the secret image's information; both carriers use the same two subpixel positions as SIPs. The secret information is represented in every block by a subpixel pair, black-white or white-black, as shown in Figure 2. The choice of pattern, black-white or white-black, used for the SIPs in carrier 1 is set at random for each block, ensuring secrecy of the secret image. The pattern for carrier 2 is determined by that for carrier 1 and the color of secret pixel. If the secret pixel is white, then the SIP patterns in the two carriers are the *same*, ensuring that in the stacked output carriers, a white subpixel appears within the block. If the secret pixel is black, then the patterns of the SIPs in the two carriers are *opposite*, so that the block in the stacked image is all black. Thus, the stacked block has either one or zero white subpixels, indicating a white or black secret pixel respectively. The secret message cannot be determined from a single carrier: the random choices ensure security.

Note that black and white pixels in the input *carriers* are mapped into subpixel blocks with average brightness $1/4$ and $2/4$, while black and white pixels in the *secret image* are mapped into subpixel blocks with average brightness $0/4$ and $1/4$ in the stacked result. In both cases, there is a loss of contrast, and previous work has concentrated on approaches to mitigate this problem.

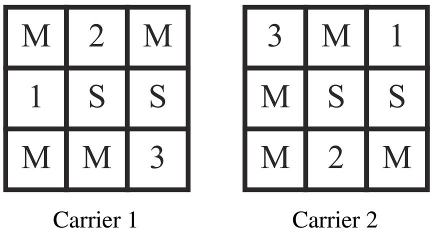
To extend the VC scheme to cope with grayscale images, halftoning and dithering techniques may be used to convert them into binary images [ZADC06, MTMT07, WADC09, LW11]. In partcular [LW11] adopts patterning dithering [Lim69] which works by replacing the grayscale pixel values by binary pixel blocks. Patterning dithering uses a dither matrix to determine the order in which the pixels are set to black to achieve the desired grayness for the blocks. In an analogous way, Liu [LW11] controls the gray level of a subpixel block in VC by setting a different number of CIPs to black.

If blocks of size other than 2×2 are used, there are constraints on the numbers SIP, CIP, and MP subpixels used in each block [LW11]. Suppose there are N_c CIPs and N_m MPs in a subpixel block. To ensure that carrier images are concealed by the MPs after stacking, we require that:

$$N_c \leq N_m. \tag{1}$$

Furthermore, for secrecy, the numbers of black and white SIPs (denoted as $N_{s_0} \geq 1$ and $N_{s_1} \geq 1$ respectively) should be the same for all blocks in a carrier, so that the SIPs have uniformly distributed brightness in the output carriers.

For example, consider a case in which each input carrier pixel is mapped to a 3×3 block [LW11]. The above constraints can be met by using 3 subpixels for CIPs, 4 for MPs, and 2 (1 pair) for the black and white SIPs. Figure 3 shows one possible arrangement for these subpixels. To determine the color of the CIPs, the gray scale input carriers are first quantized into 4 gray levels, as there are 3 CIPs in each block. As in a dither matrix, the numbers in the blocks indicate in which order the CIPs are set to white to achieve varying gray levels. To determine the color of the SIPs, the secret image is quantized into 2 gray levels: 2 SIPs can only represent 2 different colors in the stacked image. The blocks for carrier 1 and carrier 2 either have the same or complementary SIP patterns according to the corresponding quantized secret pixel. Again, for security, the SIP patterns should be randomly chosen.



M	2	M
1	S	S
M	M	3

Carrier 1

3	M	1
M	S	S
M	2	M

Carrier 2

Figure 3: *One possible arrangement of CIPs (numbered), SIPs (marked S) and MPs. Carriers are set to white in the numbered order given to achieve the desired gray level.*

Examples of output carriers generated by the algorithm of [LW11] can be seen in the top row of Figure 4. Close-ups are given in Figure 5. Details such as edges and textures are indistinct (e.g. the text on the aircraft is illegible). Each pixel in the source carrier becomes a 3×3 block in the output carrier, and the locations of the CIPs and SIPs are chosen for each block in a way which ignores edges and texture in the source input.

We show in this paper how to overcome this defect. In particular we show how to preserve details in the output carrier images, thus improving the visual quality of the results, with no significant effect on the output secret image. We next give a detailed description of our method.

3.2. Structure-Aware Visual Cryptography

The way in which we deal with the loss of detail in previous grayscale VC methods is to rearrange the positions of the subpixels in each output carrier block to form clear edges and more effectively preserve the structure information. Suppose the set of equal-sized source carrier images is $I = \{I_1, \cdots, I_n\}$ and the corresponding set of output carrier images is $C = \{C_1, \cdots, C_n\}$. Our goal is to maximise the similarity between each pair of I_k and C_k. Here, similarity is a pixel-wise measurement considering luminance, contrast and structure, as detailed later. Since rearranging the positions of the subpixels in a given block may increase the similarity for one source-output pair, but decrease it for others, there must be a trade off. We prefer to preserve structure in visually important places. An importance map for each image can be determined manually, or computed as the image saliency [CZM*11, PKPH12].

Formally, we can model the structure preservation problem as an optimization problem. Following [PQW*08], our objective function contains two main terms which account

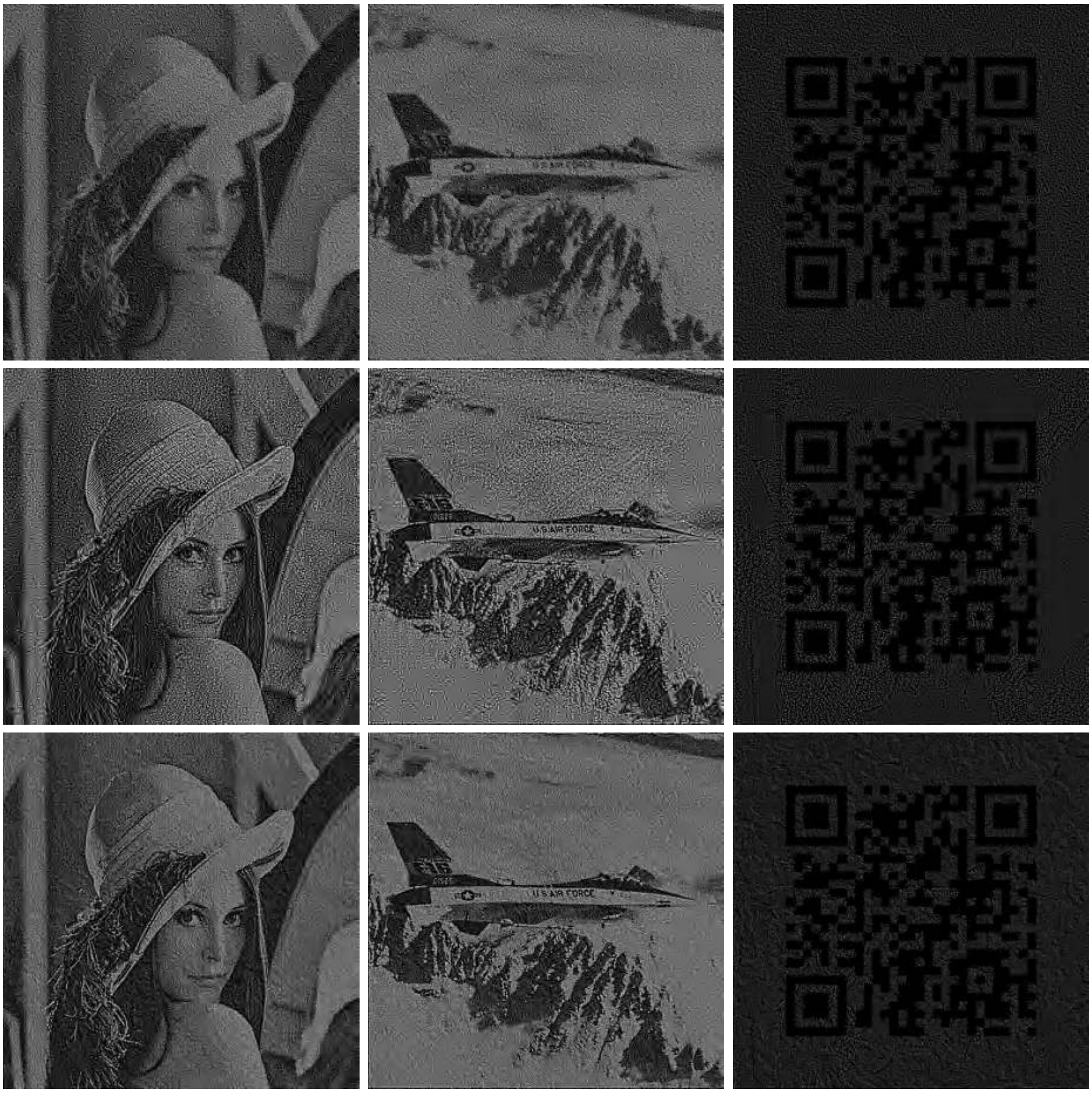


Figure 4: *Dithering VC scheme. Left, middle: output carriers. Right: decrypted hidden image. Top: method in [LW11]. Middle: minimizing the structure energy as in Eqn. 2. Bottom: optimizing the structure and coherence energy as in Eqn. 4. Note: these and other images in the paper* must *be viewed at high resolution / magnification for clarity, and to avoid aliasing.*

for structural similarity and tone similarity. Structural similarity is measured by SSIM [WBSS04] using pairs of corresponding pixels in the local neighborhoods of a given pixel. Statistical information including the weighted mean intensity and standard deviation around the given pixel in each image are compared to take into account similarity of luminance, contrast and structure. Tone similarity [PQW*08] computes the difference in luminance between each pair of corresponding pixels. The relative importance of these two terms is adjusted by saliency weights. The objective function measuring structure energy at each pixel position p is thus:

$$E_s^p(I,C) = \sum_{k=1}^{n} w_k^p (w_g G^p(I_k,C_k) + w_t(1 - SSIM^p(I_k,C_k))), \quad (2)$$

$$G^p(I_k,C_k) = (g^p(I_k) - g^p(C_k))^2, \quad (3)$$

where w_k is the pixel-wise importance map of I_k. G and

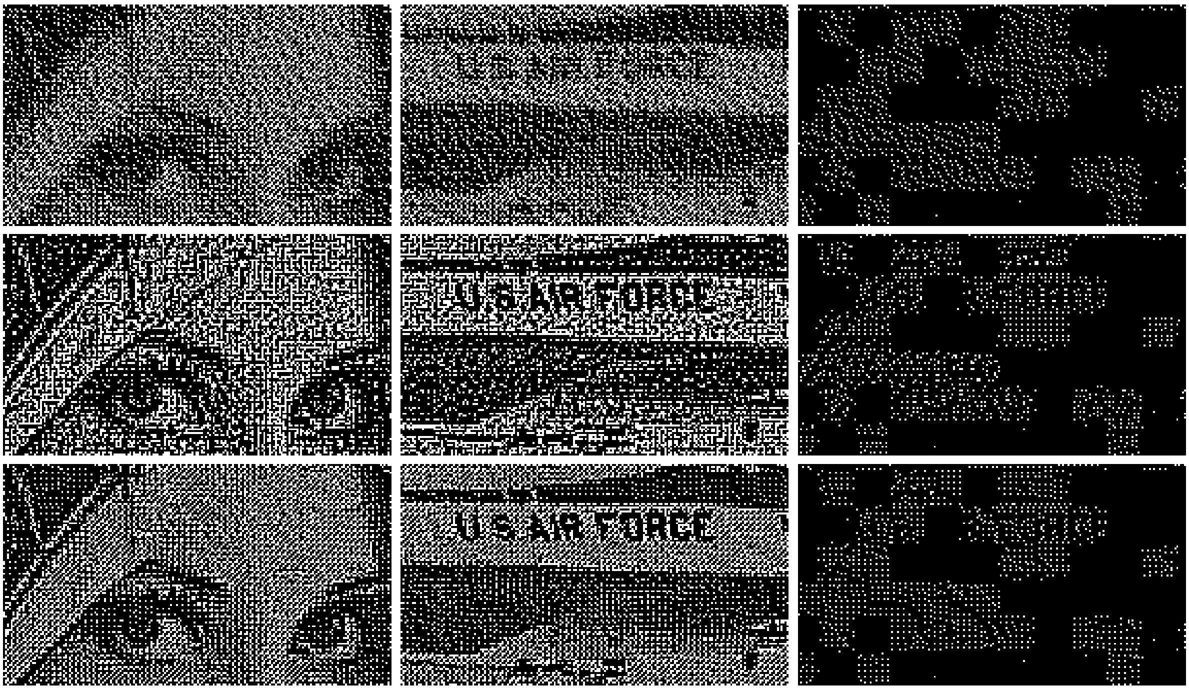


Figure 5: *Close ups of images in Figure 4. Top: method in [LW11]. Middle: minimizing structure energy. Bottom: optimizing structure and coherence energy.*

SSIM are pixel-wise similarity measurements.$g(\cdot)$ produces a blurred version of the given input image using a Gaussian kernel of size 11×11, to make the tone similarity computation G more robust. w_g and w_t are weights that control the relative importance of structural and tone similarities; both are set to 0.5 in our implementation. Because the arguments of *SSIM* and G should be two images of the same size, the I_k are enlarged to the size of C_k when computing these two terms.

We optimize the energy in Equation 2 using an iterative process. An initial output carrier set C is initialized using the algorithm of [LW11], and is then optimised using simulated annealing, following [PQW*08]. In each iteration, we consider in turn each output block for all carriers simultaneously, and make a random rearrangement of its subpixels, so that the color of the stacked block remains unchanged. Note that when we rearrange a block b, the energy values in a local area R within a radius of 6 pixels (half of the kernel size plus half of the block size) around the center of b are affected: pixels within b are used when computing the structure and tone similarity at positions in R. The total energy in R is re-calculated and if the new energy is smaller than the old one, we accept the rearrangement. Otherwise we accept it with a probability of $e^{-\Delta E_s/T}$, where T is the temperature in the annealing algorithm which is initialized to 0.2 and decreased by a factor of 0.95 in each iteration. Up to 9 random rearrangements are considered at each position before proceeding to the next block. By reducing the structure energy in this way, the subpixels in a block with same color tend to fall along the edges, preserving the structure.

Sample results of this optimization approach are shown in the second row of Figures 4 and 5. The edges are enhanced, but smooth areas become noisier in the carriers: on the one hand, exisiting noise and small details are enhanced along with edges, and on the other hand, some information from one carrier leaks to the other. For example, in carrier 2, in the second row of Figure 4 unnatural black lines appear above the plane, because the CIPs in carrier 1 (Lena) form strong white lines in the corresponding place. In order to conceal these CIPs, the mask pixels in carrier 2 must form lines of black pixels in the same place. As most other pixels in that area are white, the lines of MPs stand out.

In the next part we show how to modify the definition of the energy function to ameliorate the problem of noise, and suppress the leakage of information between carriers to some extent.

3.3. Noise and Leakage

The method above improves detail, but leads to noisy results whose visual quality is still low; information leakage between carriers also occurs.

To avoid these problems, we need to take into account that subpixel patterns should be coherent in textureless areas: they should be the same for a group of neighboring pixels having similar intensity values in the source image, to ensure that smooth areas remain smooth in the output and are not affected by other carriers, avoiding information leakage. This requirement is added as a further coherence term E_c to the previous energy function defined in Equation. 2:

$$E^p_{\text{total}}(I,C) = E^p_s(I,C) + \lambda E^p_c(I,C) \qquad (4)$$

$$E^p_c(I,C) = \sum_{k=1}^{N} \sum_{q \in N(p)} \left\| C^{q'}_k - C^{p'}_k \right\| e^{-\beta \cdot \| I^q_k - I^p_k \|} \qquad (5)$$

where E_s is the structure term in Equation 2. E_c is the coherence term. p,q are pixel locations in the input carrier images, and p', q' are corresponding subpixel blocks in the output carriers. $N(p)$ is the set of pixels neighboring p; we use 8-connected neighborhoods in our implementation. As we do not wish to constrain output carrier pixels $C^{q'}_k$ and $C^{p'}_k$ to have the same pattern if their corresponding input carrier pixels I^p_k and I^q_k differ, an exponential term is used to attenuate the energy as the difference between I^p_k and I^q_k increases. The attenuation rate is determined by β. λ controls the relative importance of structure energy and the coherence energy. We set $\lambda = 0.01$ and $\beta = 5$ respectively.

The bottom row of Figures 4 and 5 shows the VC output when noise reduction is used. Simultaneously optimizing structure and coherence energy as in Equation 4 produces better results than either of the other two methods shown. The results of patterning dithering lack detail, while only optimizing Equation 2 suffers from increased noise, and leakage which is noticeable in smooth areas of one carrier where there are strong edges in the corresponding place in the other carrier. Adding the coherence term encourages the smooth areas in the input carriers to remain smooth in the output, suppressing both noise and leakage. For example, see Lena's hat in the last two rows of Figure 4. The amount of detail preserved can be controlled by adjusting λ in Eqn. 4: smaller λ preserves more detail, but also leads to more noise and leakage. Figure 6 shows results for different λ. Eqn. 2 is the extreme case where $\lambda = 0$.

4. Results

To evaluate our method, we have tested it on various types of images. We first provide a visual comparison to previous methods, and then provide a quantitative evaluation of the algorithm. We also conducted a user study to evaluate it.

4.1. Visual Quality

We compare our structure aware VC encryption with the results of [LW11] in Figures 4 and 5. It shows that our methods can better preserve fine details and sharp edges after the random rearrangement phase. By also adding the coherence term, the rearrangement results in less noise and information leakage. Figures 1 and 7 show a further comparison. The hair and feather of the animals and the sketch strokes in the carriers are captured much better by our method. Although there are slight traces of the mixed carriers in the stacked result, they do not affect its comprehensibility. More importantly, it is very hard to detect any meaningful traces of one carrier in the other for these complex images. The other carrier is not visible at all for the cat and dog images. Even in the background of the tank image, which was originally uniform, while there is clearly a little structure, it is very hard to see and recognise details of the eagle.

4.2. Numerical Comparison

We next assess the visual quality of our methods for various images using two well-known metrics. We follow the same approach used in other work to measure how well structure is preserved [PQW*08, CAO09, LM10]: we compute the mean structure similarity measure (MSSIM) and the peak signal-to-noise ratio (PSNR), comparing the original input image, the carrier produced using the method in [LW11], and our structure aware carrier. The MSSIM is the mean value over the image of SSIM [WBSS04]. The PSNR is defined in terms of the mean squared error (MSE) between two images I_1 and I_2:

$$\text{PSNR}(I_1, I_2) = 10 \log_{10} \left(\frac{1}{\text{MSE}(I_1, I_2)} \right),$$

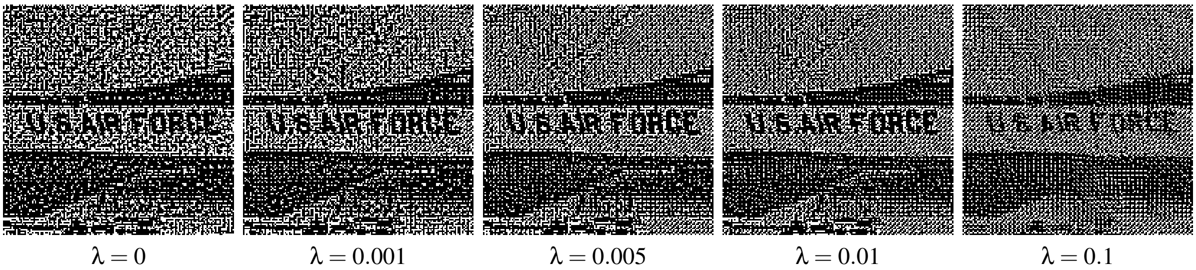
assuming pixel values in the range $[0,1]$. Note that the source carrier must be first scaled to the same size as the output; bilinear interpolation was used to antialias the scaling result.

Analysis results are given in Tables 1 and 2. Higher MSSIM and PSNR values mean that two images are more similar. Our methods generally achieve higher MSSIM and PSNR values than the method in [LW11]. This is because they optimize the structure and tone similarity between the source images and the output carriers. With the coherence term, the weak edges and texture are smoothed instead of enhanced, which causes the decrease in structure similarity.

Since all output images are monochrome, a better way to compare them is to convert them to grayscale images. To do this, we apply a 5×5 Gaussian blur kernel to each output carrier image, and recompute the assessment: see Tables 1 and 2. We can see in some cases using the energy function in Equation 2 performs similarly to the method in [LW11] due to the noise. Adding the coherence term improves the results by this measure.

4.3. User Study

We also conducted a user study to evaluate the our algorithm. Ten participants were asked to score results based on image quality and security; high security means that one can hardly notice information leakage between the carriers. The range of scores was 1 to 5. Table 3 summarizes the results. From



$\lambda = 0$ $\lambda = 0.001$ $\lambda = 0.005$ $\lambda = 0.01$ $\lambda = 0.1$

Figure 6: *Carriers produced by different choices of* λ *in Equation 4. The inputs are same as those in Figure 4.*

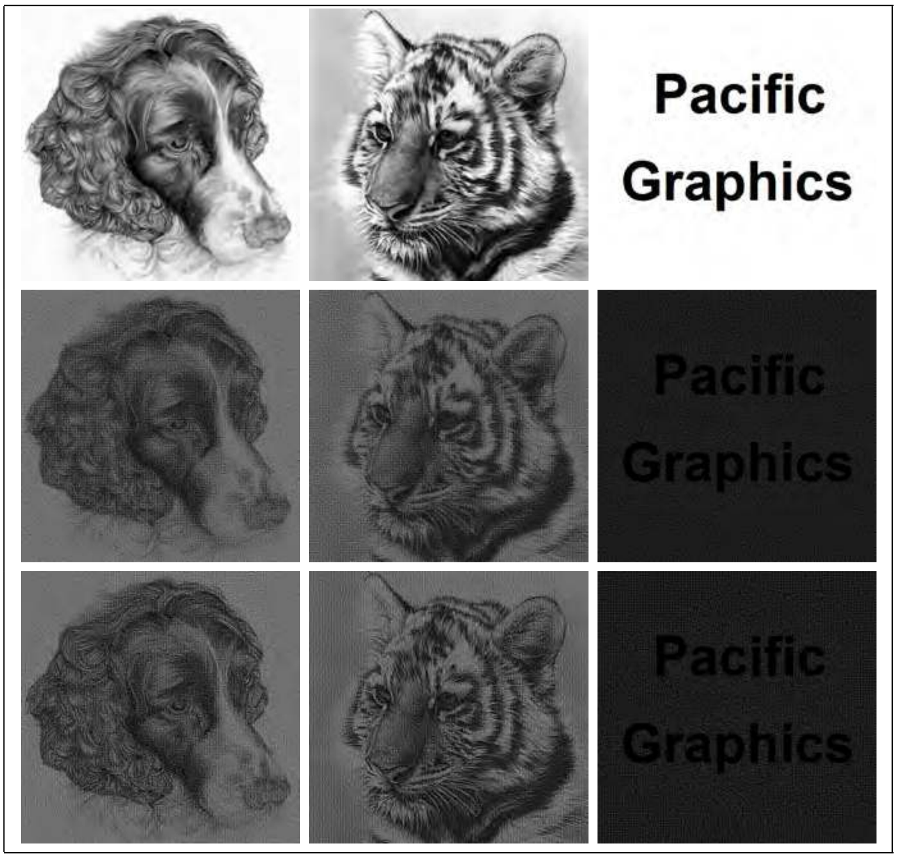


Figure 7: *Further VC results. Top: input carriers and secret image; middle: using the method in [LW11]; bottom: using our final method.*

Method	Carriers		Stacked image
	Quality	Security	Quality
Using [LW11]	3.23	4.33	3.35
Using Eqn. 2	3.39	3.58	3.13
Using Eqn. 4	3.81	3.80	3.28

Table 3: *Mean user study scores. Higher numbers indicate better perceived quality or security.*

the table, our algorithms based on Eqns. 2 and 4 achieve higher scores for image quality than [LW11]. By adding the coherence term, both quality and security of the carriers are improved. However, our results achieve a lower score for the quality of the decrypted image. This may be because there are some undesired artifacts in the stacked images, e.g. where one input carrier has strong black edges while the other is bright and smooth. Avoiding such artifacts is a topic for future work.

Method	lena	airplane	dog	tiger	eagle	tank
Using [LW11]	0.013	0.016	0.009	0.013	0.011	0.025
Using Eqn. 2	0.042	0.044	0.051	0.063	0.054	0.102
Using Eqn. 4	0.027	0.032	0.028	0.034	0.028	0.095
Using [LW11], blurred	0.335	0.327	0.381	0.352	0.356	0.371
Using Eqn. 2, blurred	0.402	0.378	0.394	0.424	0.402	0.252
Using Eqn. 4, blurred	0.480	0.461	0.441	0.422	0.405	0.411

Table 1: *Mean SSIM between the input and output carriers for different methods. Above: using images directly. Below: output carriers are first blurred by a* 5×5 *Gaussian filter to convert them to grayscale.*

Method	lena	airplane	dog	tiger	eagle	tank
Using [LW11]	5.63	4.54	4.48	4.94	5.69	3.31
Using Eqn. 2	5.81	4.71	4.61	5.13	5.87	3.54
Using Eqn. 4	5.72	4.65	4.54	5.02	5.77	3.52
Using [LW11], blurred	10.53	8.55	8.33	9.27	10.86	6.05
Using Eqn. 2, blurred	10.62	8.64	8.33	9.31	10.86	6.08
Using Eqn. 4, blurred	10.69	8.70	8.36	9.33	10.91	6.10

Table 2: *PSNR between the input and output carriers for different methods. Above: using images directly. Below: output carriers are first blurred by a* 5×5 *Gaussian filter to convert them to grayscale.*

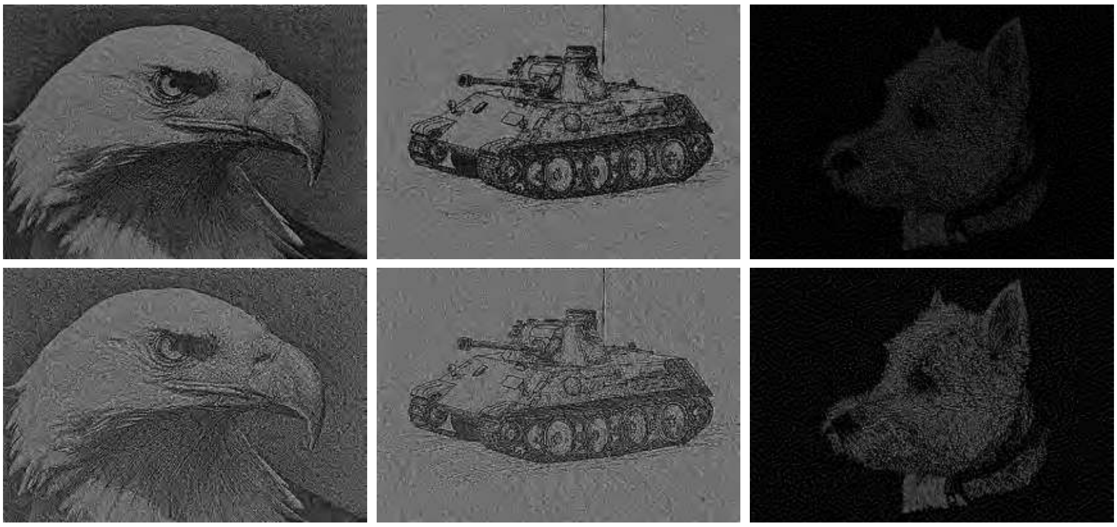


Figure 8: *Using different schemes for subpixel assignment in each block. Above: 3 CIPs, 4 MPs and 2 SIPs. Below: 2 CIPs, 3 MPs and 4 SIPs.*

4.4. Discussion and Limitations

The main limitation of our method is the time required. As our algorithm is iterative, it takes longer than previous methods. We have implemented it in C++. On a PC with an Intel Core i7 3.4GHz CPU it takes about 50 seconds to encrypt a set of two carriers and a secret image, each of size 512×512.

Adding the extra coherence term can reduce noise and information leakage between carriers in smooth regions, and while they remain in textured regions, they are typically less noticeable there.

SSIM, used in Equation. 2, is probably not a good way to measure structural similarity between a monochrome image and a grayscale image.

So far we have discussed the case where 2 SIPs and 3 CIPs are used in each block. If we increase the number of SIPs, the contrast of the decrypted secret image will be increased, but the price is that the contrast of the carriers is decreased. Figure 8 shows such a result.

5. Conclusions

We have presented a structure preserving VC scheme that maintains edges and other structure detail in the output carriers. Our approach uses a simulated annealing algorithm, and takes into consideration the structure and tone similarity between the source images and the carriers as well as the coherence between each subpixel block and its neighbors in the output carriers. Compared to state-of-art VC algorithms, our method better preserves the fine details such as sharp edges and textures.

For future work, methods like those in [CAO09] may be used to accelerate the algorithm. Finding a better similarity measurement which can relate monochrome and grayscale images may lead to more accurate evaluation. Another direction is to exploit higher order image features to better describe image structure.

References

[ABSS01] Ateniese G., Blundo C., Santis A. D., Stinson D. R.: Extended capabilities for visual cryptography. *Theoretical Computer Science 250*, 1 (2001), 143–161. 2

[CAO09] Chang J., Alain B., Ostromoukhov V.: Structure-aware error diffusion. In *ACM Transactions on Graphics (TOG)* (2009), vol. 28, ACM, p. 162. 3, 7, 10

[CCLM13] Chu H.-K., Chang C.-S., Lee R.-R., Mitra N. J.: Halftone qr codes. *ACM Trans. Graph. 32*, 6 (2013), 217. 3

[CHM*10] Chu H.-K., Hsu W.-H., Mitra N. J., Cohen-Or D., Wong T.-T., Lee T.-Y.: Camouflage images. *ACM Transactions on Graphics 29*, 3 (2010), 51. 2

[CY11] Cimato S., Yang C.-N. (Eds.): *Visual cryptography and secret image sharing*. CRC Press, 2011. 1, 2

[CZM*11] Cheng M.-M., Zhang G.-X., Mitra N. J., Huang X., Hu S.-M.: Global contrast based salient region detection. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on* (2011), IEEE, pp. 409–416. 4

[FS75] Floyd R., Steinberg L.: An adaptive algorithm for spatial gray scale. 1

[HC00] Hou Y.-C., Chen P.-M.: An asymmetric watermarking scheme based on visual cryptography. In *Signal Processing Proceedings, 2000. WCCC-ICSP 2000. 5th International Conference on* (2000), vol. 2, IEEE, pp. 992–995. 1

[HH05] Hsu C.-S., Hou Y.-C.: Copyright protection scheme for digital images using visual cryptography and sampling methods. *Optical Engineering 44*, 7 (2005), 077003–077003. 1

[Hou03] Hou Y.-C.: Visual cryptography for color images. *Pattern Recognition 36*, 7 (2003), 1619–1629. 2

[Hwa00] Hwang R.-J.: A digital image copyright protection scheme based on visual cryptography. *Tamkang Journal of science and Engineering 3*, 2 (2000), 97–106. 1

[KAL11] Kang I., Arce G. R., Lee H.-K.: Color extended visual cryptography using error diffusion. *Image Processing, IEEE Transactions on 20*, 1 (2011), 132–145. 2

[Lim69] Limb J.: Design of dither waveforms for quantized visual signals. *Bell System Technical Journal 48*, 7 (1969), 2555–2582. 1, 4

[LM10] Li H., Mould D.: Contrast-aware halftoning. In *Computer Graphics Forum* (2010), vol. 29, Wiley Online Library, pp. 273–280. 3, 7

[LW11] Liu F., Wu C.: Embedded extended visual cryptography schemes. *Information Forensics and Security, IEEE Transactions on 6*, 2 (2011), 307–322. 1, 2, 4, 5, 6, 7, 8, 9

[LY14] Liu F., Yan W. Q.: *Visual Cryptography for Image Processing and Security*. Springer, 2014. 2

[MCL*09] Mitra N. J., Chu H.-K., Lee T.-Y., Wolf L., Yeshurun H., Cohen-Or D.: Emerging images. In *ACM Transactions on Graphics (TOG)* (2009), vol. 28, ACM, p. 163. 2

[MTMT07] Myodo E., Takagi K., Miyaji S., Takishima Y.: Halftone visual cryptography embedding a natural grayscale image based on error diffusion technique. In *Multimedia and Expo, 2007 IEEE International Conference on* (2007), IEEE, pp. 2114–2117. 4

[NS95] Naor M., Shamir A.: Visual cryptography. In *Advances in Cryptology - EUROCRYPT'94* (1995), Springer, pp. 1–12. 1, 2, 3

[NY02] Nakajima M., Yamaguchi Y.: Extended Visual Cryptography for Natural Images. In *International Conference in Central Europe on Computer Graphics and Visualization* (2002), pp. 303–310. 2

[PKPH12] Perazzi F., Krahenbuhl P., Pritch Y., Hornung A.: Saliency filters: Contrast based filtering for salient region detection. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (2012), IEEE, pp. 733–740. 4

[PQW*08] Pang W.-M., Qu Y., Wong T.-T., Cohen-Or D., Heng P.-A.: Structure-aware halftoning. In *ACM Transactions on Graphics (TOG)* (2008), vol. 27, ACM, p. 89. 3, 4, 5, 6, 7

[SLY14] Shi Y. Q., Liu F., Yan W. (Eds.):. *Transactions on Data Hiding and Multimedia Security IX* (2014), vol. 8363 of *Lecture Notes in Computer Science*, Springer. 2

[WADC09] Wang Z., Arce G. R., Di Crescenzo G.: Halftone visual cryptography via error diffusion. *Information Forensics and Security, IEEE Transactions on 4*, 3 (2009), 383–396. 1, 2, 4

[WBSS04] Wang Z., Bovik A. C., Sheikh H. R., Simoncelli E. P.: Image quality assessment: from error visibility to structural similarity. *Image Processing, IEEE Transactions on 13*, 4 (2004), 600–612. 3, 5, 7

[ZADC06] Zhou Z., Arce G. R., Di Crescenzo G.: Halftone visual cryptography. *Image Processing, IEEE Transactions on 15*, 8 (2006), 2441–2453. 1, 2, 4