IMAGE RECOGNITION-BASED AUTOMATIC DECRYPTION METHOD FOR TEXT ENCRYPTED USING VISUAL CRYPTOGRAPHY

Naoyuki Awano

Department of Computer and Information Science, Seikei University, Tokyo, Japan

ABSTRACT

Using passwords only has rapidly become a security risk. Another approach to security is visual cryptography (VC), which divides paper documents into several encrypted papers managed by multiple people. Decryption occurs by stacking these papers, i.e., they cannot be decrypted individually. In our work, we consider a system for decrypting text encrypted by VC on digital devices. Furthermore, we propose a method for automatically recognizing encrypted portions using images captured by a digital device's camera. Our system has several advantages, including no actual text in communication and enabling users to use different passwords or secret questions at each use. Furthermore, our method is implementable on wearable glasses-like devices, thus enabling wearers to decrypt text simply by looking at encrypted portions. We conducted experiments regarding recognition accuracy and performance and obtained results showing that our proposed method was able to achieve a high recognition rate at a low cost.

KEYWORDS

Image Recognition, Decryption, Digital Device, Visual Cryptography

1. INTRODUCTION

In the last ten years, an increased number of services become available via the Internet; thus, Internet security techniques have rapidly gained significance. Many Web services require user authentication, with the most common method for personal authentication being the use of passwords due to their simple structure requiring users only retain a string of characters.

More recently, using only passwords has become increasingly risky in terms of security, because serious damage can be caused by password leaks. However, password authentication is very convenient; therefore, security improvements have typically assumed the continued use of passwords. Given the increase in security breaches, it has become crucial to combine multiple security techniques, including the use of secure sockets layer (SSL), the incorporation of secret questions and one-time passwords, and requiring passwords to contain a variety of character types (e.g., uppercase, lowercase, numerals, and punctuation).

Considering the scenario, when the Web was in its infancy, Naor and Shamir proposed a security technique for paper documents known as visual cryptography (VC) [1], which is illustrated in Figure 1. For encryption, the VC method divides a document into several encrypted documents, whereas for decryption, the original document can be extracted by stacking all or several original documents in any order. As a feature, the information can be only understood visually. The VC method does have some disadvantages, including the requirement of a dedicated device for printing to sufficiently transparent paper, the need for proper management of these separate papers, and the burden of exactly stacking these papers. Nonetheless, a key feature is that the original information cannot be decrypted by only one person, even if some encrypted documents are lost or leaked; in other words, all or several separate pieces are required to reconstruct the originally encrypted visual information.

DOI: 10.5121/ijcsea.2017.7401

As a means of taking advantage of these features, Kato and Imai proposed a VC method for personal authentication [2]. Here the system and one of its users preliminarily prepare a shared image in which random patterns are placed. This shared image, which the user retains, serves as a dedicated common-key image. First, the system creates a message that is converted to an image and then encrypts it using the common-key image. Next, the system sends this encrypted image to the user. Finally, the user stacks the encrypted image on the dedicated common-key image to and decrypts and reads the original message. Note that the above flow is merely conceptual and is yet to be attempted in practice.

Therefore, in this paper, we discuss the details of the above system for personal authentication using VC on digital devices; we consider its feasibility and system requirements. Since various digital devices (e.g., smartphones, tablets, and laptops) are used often in lieu of paper, our proposed method instead focuses on them, decrypting text by taking pictures of the encrypted images.

In particular, for the first time according to our knowledge, we consider a simple means of decrypting text, a means of representing large amounts of text in an image, a means of displaying that image, and a means of determining what the optimal resolution should be. Furthermore, as a simple means of decrypting text, we also propose a novel method for automatically recognizing encrypted portions using captured images from a digital device's camera. To recognize only encrypted portions with a high recognition rate, our proposed method combines rendering filters that have been proposed for three-dimensional (3D) point clouds in computer graphics. Furthermore, we also propose dedicated image correction to enhance the ability to understand the hidden text.

Using above methods, a different password or secret question can be used every time; moreover, there is no actual text included in the communication, except for user input. In addition, when the user might be inputting a password or answer to a secret question or both, nobody else is able to see what the user is responding. We expect our proposed method to be implemented on wearable glasses-like devices on which it will be able to decrypt text simply by looking at the set of encrypted images.

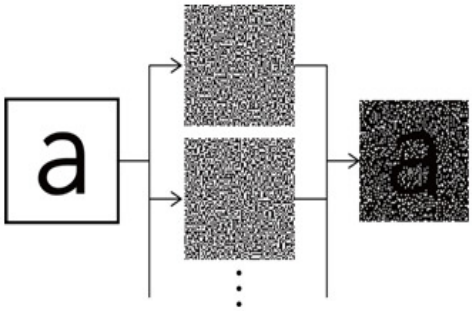


Figure 1. Illustrating visual cryptography (VC) in which information is encrypted by dividing the original into multiple encrypted pieces.

In addition to this introductory section, in Section 2, we describe conventional VC, including the basic method adopted in this paper. In Section 3, we introduce an implementable system and consider its system requirements. Next, in Section 4, we describe our automatic recognition method for encrypted images from captured images in actual system. In Section 5, we demonstrate recognition rates, the practicality of our system, and the processing times. Finally, we draw our conclusions in Section 6.

2. VISUAL CRYPTOGRAPHY (VC)

Protecting personal information and preventing it from being leaked or shared with unauthorized users have become increasingly important in this day and age; however, it is impossible to entirely prevent information leakage with 100% accuracy with current technology. Consequently, we require techniques that can prevent unauthorized users from reading protected information even if a portion of such information is leaked or stolen. One such technique, a secret-sharing scheme called the *k*-out-of-*n* threshold scheme, was proposed by Shamir [3]. In this decades-old scheme, information to be encrypted is divided into *n* encrypted pieces to be held by *n* people. The information can be decrypted from the full collection of *k* pieces, but it cannot be decrypted from only *k*-1 pieces.

By extending the above method, Naor and Shamir proposed VC, an approach that can visually decrypt encrypted information without the use of machines [1]. This method is intended for paper documents, such as the example shown in Figure 1. Since the method can be decrypted without the use of machines, it is available in such situations such as power failures.

The method [1], which is used in our paper, is explained below for the case of $k=n=2$. As shown in Figure 2, each pixel in an input binary image is first divided into two 2×2 blocks consisting of two white (i.e., transparent) pixels and two black pixels. When an input pixel is white, the blocks are arranged such that several transparent pixels remain when two blocks are stacked. Conversely, when an input pixel is black, the blocks are arranged such that all pixels are black when two blocks are stacked. Each block consists of several patterns, called "shares", as shown in Figure 3. Consequently, in the decryption step, we can read the information visually from the differences in the density of the black pixels; however, there are inherent constraints, such as the need to print on sufficiently transparent paper, to stack the papers exactly for accurate comprehension, and to retain many papers.

Many studies have since focused on VC, with nearly all such studies investigating binary images [4, 5], but several studies have extended the method to grayscale images through the use of halftoning [6, 7, 8, 9, 10]. There have also been studies that have extended its application to color images by applying the color model [11, 12, 13]. Furthermore, methods have been proposed in which the stacking of two different natural images, such as a landscape and an animal, results in a completely different image to be understood [14, 15]. Finally, there have also been proposals to expand VC to watermarking [16, 17] or personal authentication [2].

(a) (b)

Figure 2. Encryption and decryption schemes for (a) white pixels and (b) black pixels.

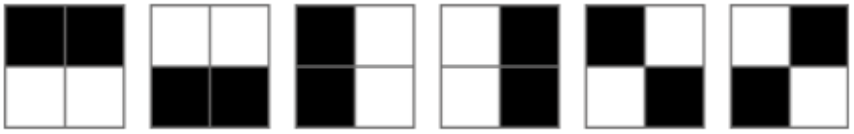


Figure 3. Shares.

3. SCHEME AND SYSTEM REQUIREMENTS

For systems that require passwords, it has become increasingly essential to use multiple security techniques, including SSL, secret questions, one-time passwords, and similar methods. In this section, we consider a system using VC on digital devices, describing our scheme and its system requirements. We make use of a two-out-of-two threshold scheme, which is the most basic of methods.

3.1. SYSTEM SUMMARY

The overall system flow of a general personal authentication system using digital devices, a concept proposed by Kato and Imai [2], is shown in Figure 4. This flow here is almost the same as that of a general common-key cryptosystem. The only prerequisite is to generate and share a common-key image composed of random shares in advance, with example shares shown in Figure 3 above. In the system, a user first sends a request, such as a user ID to the system. Next, the system generates text, for example a one-time password, as an image, and then encrypts this image using the pre-generated common-key image. The encrypted image is then sent to the user, and then the user stacks it exactly with the common-key image to decrypt the text. Finally, the user authenticates himself or herself using the given password or by following the given instructions. This approach enables the use of a different password or set of instructions every time. Further, there is no text in the communication save for the user input.

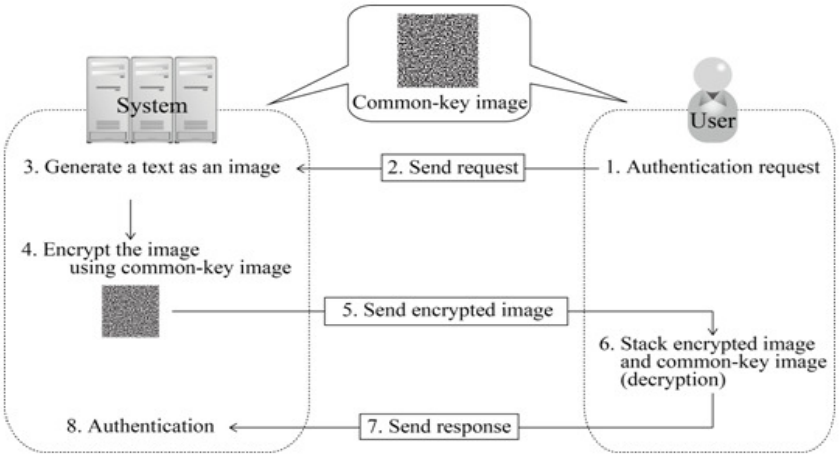


Figure 4. A system flow of a general-purpose personal authentication system [2].

Since the above system has only been proposed conceptually, detailed specifications have not yet been studied. For specific examples, we note the simple method of decrypting long text in an image, the method of displaying encrypted images, and the method for determining optimal resolutions. In this paper, we discuss the above three focus areas for achieving the specified requirements. In Section 3.2, we present a simple method of decryption that uses this system. Next, in Section 3.3, we describe a method for displaying text, and then in Section 3.4, we describe the necessary resolutions of the encrypted and common-key images.

3.2. DECRYPTION METHOD

For decryption, users are required to stack encrypted and common-key images exactly when using VC. Consequently, users would immediately benefit from the ability to decrypt images via automated means of stacking. One method that we propose in this paper is a method for decrypting text using a digital device's camera by taking a picture of the encrypted image, thus the method is expected to be usable on a smartphone or tablet, similar to Quick Response (QR)

codes, as shown in Figure 5. Furthermore, we expect our approach will be adaptable to decrypting text by simply looking at an encrypted image through wearable glasses-like devices.

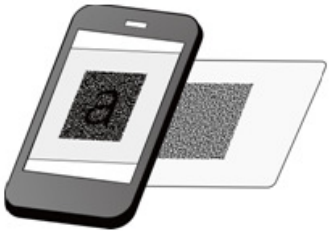


Figure 5. A potential example of decryption using a digital device.

3.3. ENCRYPTED IMAGE DISPLAY METHOD

As noted above, many VC methods have been proposed; however, since the purpose of our study is to read text from images, we note that using only binary images is sufficient. Therefore, in this paper, we use the method of Naor and Shamir for binary images [1].

Moreover, encrypted images cannot be displayed accurately if their resolution is higher than the given screen resolution. Consequently, the resolution of the encrypted image has certain limitations. In general, the longer and more complex a password is, the better, but it is difficult to represent such a password or text in a low-resolution image.

Therefore, an image represents each letter as a square, and the system displays one letter at regular time intervals. In addition, considering that the same letter is displayed in a row, blank images are displayed between the letters, as shown in Figure 6. This display method has no limitation and can use longer and more complex passwords or text.

Figure 6. Illustrating our method for displaying text (from left to right).

3.4. RESOLUTION OF COMMON-KEY AND ENCRYPTED IMAGES

To be automatically decrypted, the target region of the encrypted image captured by a camera must be automatically recognized. In general, a letter can be accurately represented in a display if the encrypted image has a high-enough resolution; however, if the resolution is excessively high, such as beyond that of the captured image, the encrypted image cannot be accurately represented in the captured image.

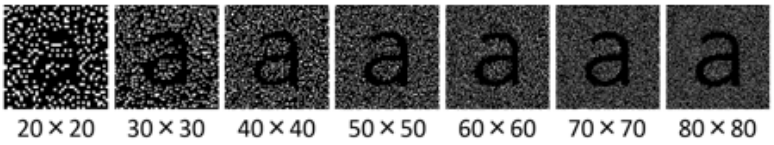
In contrast, it becomes straightforward to represent an encrypted image in a captured image if the encrypted image has relatively low resolution. In addition, we can expect to achieve high recognition even if the camera moves, but, a letter cannot be represented in an image with excessively low resolution. Accordingly, this suggests that we can achieve high recognition if we know the lowest resolution at which we can read a letter.

Typical characters used as parts of passwords or text are shown in Table 1. These 94 characters were tested for their readability. To simplify the problem, monospaced fonts were used in our experimentation. In addition, we selected two monospaced fonts, thus totaling 188 characters, namely Inconsolata and Courier New, for further experimentation. A character was placed in the center of a square image, with the top and bottom margins relative to the tops and bottoms of all characters of the font set to 5%.

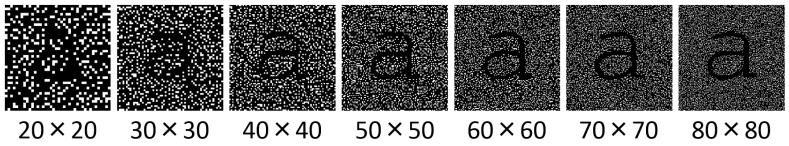
Table 1. Characters assumed to be used in passwords and text.

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z		
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z		
0	1	2	3	4	5	6	7	8	9																		
#	!	$	%	&	(	)	*	,	.	-	/	:	;	?	@	[	]	^	_	`	{	}	|	~	'	"	¥	+	<	>	=

Figure 7 shows examples of exactly stacked and decrypted results corresponding to each font. Since Courier New is thinner than Inconsolata, we observe that Courier New required a higher resolution to represent each character. From the results observed using all characters, we experimentally determined that all characters could be represented using resolution 100×100. Therefore, this resolution was used henceforth as the optimal resolution of all encrypted and common-key images.



(a) Inconsolata



(b) Courier New

Figure 7. Decryption results for various resolutions.

4. RECOGNITION OF ENCRYPTED IMAGE FOR DECRYPTION

Recognition based simply by taking a picture of an encrypted image enables automatic decryption. To simplify processing, an input captured image is converted into an 8-bit grayscale image in advance. Then, a dedicated binarization process is applied to the input image to detect the target region of the encrypted image. Next, a rectangle is extracted from the binary image with its corners extracted. After that, the encrypted image is extracted and corrected to improve decryption accuracy. Finally, superimposed onto the common-key image, the image is projected onto the input captured image.

4.1. BINARIZATION OF AN ENCRYPTED IMAGE

Since the encrypted image consists of shares shown in Figure 3, only that part of the input image appears as a mass of noise. Therefore, that part is recognized as the encrypted image in this study. For binarization, each pixel in the input image refers to the neighboring $N\times N$ pixels, thus we find maximum pixel value L_{max} and the minimum pixel value L_{min}. For our proposed method, we experimentally determined that N=10. Next, new pixel value $I'(p)$ is calculated as

$$I'(p) = C(128, L_{min}) + C(L_{max}, 128) \quad (1)$$

$$C(\alpha,\beta)=\begin{cases}255 & (\alpha \le \beta)\\ 0 & (\alpha > \beta)\end{cases} \quad (2)$$

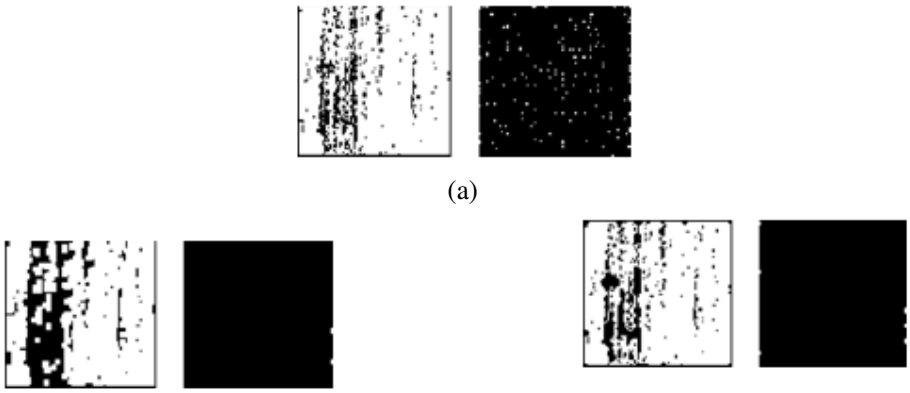
Widely varying parts of neighboring pixel values are converted to black using the above equations, as exemplified in Figure 8. Furthermore, a large black part is extracted as an encrypted image, as described in the next section.

(a) (b)

Figure 8. Binarization of various input images with (a) the input images and (b) the resulting binarizations.

When photographs are taken at close range, some white pixels might remain in a black region, as shown in Figure 9(a), because the encrypted image is represented using a high resolution. In such cases, morphological dilation and erosion are generally applied, and inner white pixels are filled; however, this also has the effect of merging with the noise, as shown on the left-hand side in Figure 9(b). Given this, recognition accuracy tends to drop if there is noise around any black regions.

Consequently, in our proposed method, we apply rendering filters for 3D point clouds, as proposed by Dobrev et al. [18]. More specifically, 3D point clouds are acquired by a 3D scanner and consist only of points. Because a point cloud has no surface, many background pixels remain in the object region when it is rendered. Dobrev et al. proposed conditional dilation filters that fill background pixels. Those filters can dilate object pixels while maintaining the silhouette of the shape. As the state of the image after our proposed binarization process is applied is similar, we apply our proposed method to the binary image assuming that black pixels are the object region and white pixels are the background. Applying this method can indeed yield good results, as shown in Figure 9(c).



(a)

(b) (c)

Figure 9. Dilation using filters showing (a) results of binarization, (b) dilation and erosion, and (c) results of the method by Dobrev et al. [18]

4.2. CORNER DETECTION OF AN ENCRYPTED IMAGE

In this section, we extract a large black region as an encrypted image under several conditions, and then the corner points on the large black region for projection transformation. In this paper, we apply a hierarchical contour detection [19] technique to the binary image created in the previous section. Subsequently, the encrypted image is a region satisfying all of the conditions below.

CONDITION 1. STRUCTURE OF CONTOURS

The encrypted image is a large black region. Since the target region has no white pixels, by the previous section, the regions that have no other contour on the inside are regarded as the encrypted image.

CONDITION 2. DEGREE OF SQUARE

The encrypted image is a black rectangle on a binary image. The degree of square Q is calculated as

$$Q = 16A / L^2 \tag{3}$$

where A is the area of the interior, and L is the length of the contour. In our proposed method, we set a threshold Tq such that rectangles whose Q is greater than Tq are regarded as the encrypted image. For our proposed method, we experimentally determined Tq=0.9.

CONDITION 3. RESOLUTION

If the interior resolution of the rectangle is insufficient, it cannot be decrypted even if it is superimposed onto the common-key image. For example, when the resolution of the common-key image is 100×100, regions with more than 400 pixels on the contour are regarded as the encrypted image.

If multiple regions are regarded as the encrypted image, then the region with maximum area A is determined as the encrypted image. Subsequently, the four most distant vertices on the contour are extracted as the corner vertices for projection transformation. Formally, pixels on the contour are denoted by $P = \{ p_i \mid i = 1,2,...,n \}$, and the four corner vertices $c_1 - c_4$ are calculated as follows:

$$c_1 = \underset{p_i}{argmax} \left\| p_i - \frac{1}{n} \sum_{j=1}^{n} p_j \right\| \tag{4}$$

$$c_2 = \underset{p_i}{argmax} \left\| p_i - c_1 \right\| \tag{5}$$

$$c_3 = \underset{p_i}{argmax} \sum_{j=1}^{2} \left\| p_i - c_j \right\| \tag{6}$$

$$c_4 = \underset{p_i}{argmax} \sum_{j=1}^{3} \left\| p_i - c_j \right\| \tag{7}$$

4.3. GENERAL FORMAT, PAGE LAYOUT AND MARGINS

Using extracted corner vertices, the common-key image is projected onto the captured image. Though it appears that decryption is complete, the accuracy of the decryption is often low and the character is difficult to discern in practice, because extracted vertices can be out of alignment in several pixels as a result, for example, of camera motion. Furthermore, when a binary image is superimposed onto a grayscale image, only the binary portion stands out as a result of the contrast. In our proposed method, we apply image correction to improve decryption accuracy, as depicted in Figure 10. First, the target region of the encrypted image is projected onto a square image using

bilinear interpolation. Second, the square image is downsampled to an image with a resolution matching the common-key image and simultaneously binarized. More specifically, the square image is a grayscale image, as shown on the left-hand side of Figure 11. Therefore, each pixel of the image is projected onto an image depicted as white lines in the figure. Since each 2×2 block has two black pixels and two white pixels, as shown in Figure 3, pixels inside the bold square line of Figure 11 should also become similar.

Accordingly, pixels inside the white lines are averaged, then downsampled to an image matching the resolution of the common-key image. Furthermore, with 2×2 pixels in a block, two darker pixels are converted to black, white the others are converted to white. Finally, the image is superimposed onto the common-key image, and then projected onto the input image.

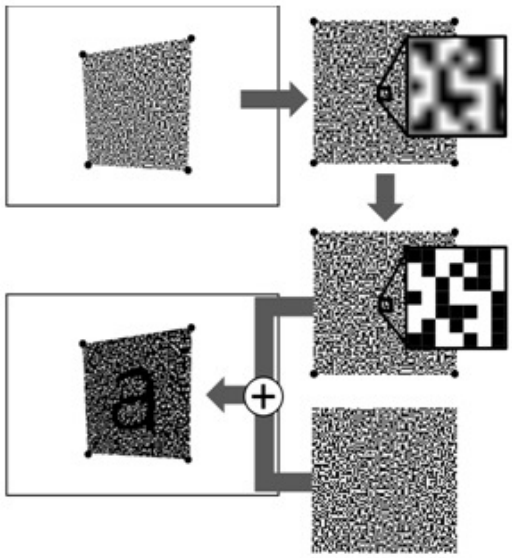


Figure 10. Decryption with image correction.

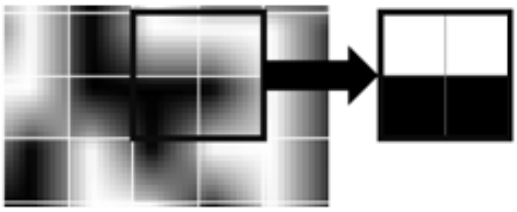


Figure 11. Downsampling and binarization.

5. VALIDATION OF METHODS

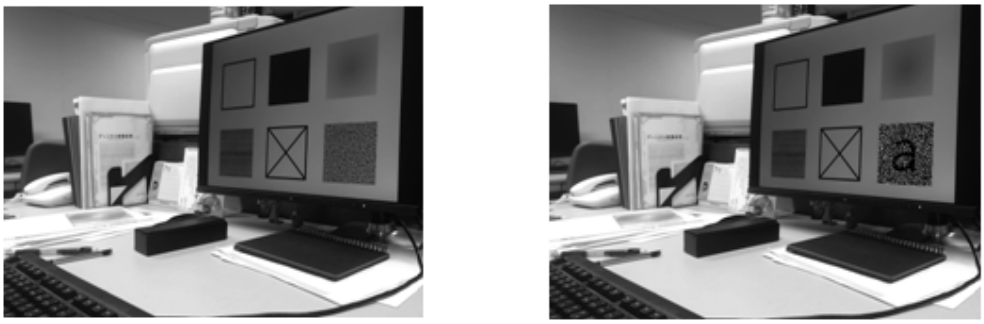
We implemented our recognition method to verify its effectiveness. In this section, we discuss the results of the decryption, the accuracy of the decryption, and the calculation costs. Images of all experiments were captured with 1080p resolution (i.e., 1920×1080).

5.1. RESULTS OF DECRYPTION

Figure 12 shows an example result of successful decryption. An encrypted image exists in the right-hand side of the captured image, as shown in Figure 12(a), and when the common-key

image is superimposed onto the captured image, the encrypted is recognized and decrypted, as shown in Figure 12(b). Not only does this confirm a successful decryption, it also confirms that our proposed method can achieve automatic decryption. As another experiment, we conducted the experiment in a place which there is no encrypted image; results here showed a false recognition rate of 0.2% per minute, which is not a problem, because our method cannot decrypt a message unless an encrypted image is actually provided.

Figure 13 shows sample results with and without image correction (i.e., Figure 13(b) and 13(a), respectively), thereby confirming that results of decryption are indeed improved by our image correction method to extent that image correction is required to detect and understand given characters.



(a) (b)

Figure 12. Example of actual image results using our method, showing (a) an input image with a hidden message and (b) the resulting image after automatic detection and decryption.

(a) (b)

Figure 13. Illustrating our image correction method both (a) without applying the correction method and (b) with the correction method applied.

5.2. DECRYPTION ACCURACY

Figure 14 illustrates our experimental setting in which an encrypted image was set at 10×10 cm with a camera set in front of the image. The following two conditions were then tested: (1) with the camera set to the same height as the center of the image; and (2) with the point of gaze always at the center of the image.

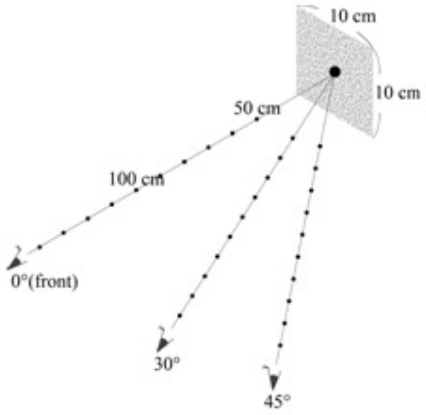


Figure 14. Experimental settings.

For our experimentation, the camera first captured an image from a distance of 50 cm. Then, accuracy was calculated using normalized cross correlation (NCC), which is a measure of the similarity of two images that is primarily used for pattern matching. It is calculated as

$$NCC = \frac{\sum_{j=0}^{M-1}\sum_{i=0}^{M-1} D(i,j)T(i,j)}{\sqrt{\sum_{j=0}^{M-1}\sum_{i=0}^{M-1} D(i,j)^2 \times \sum_{j=0}^{M-1}\sum_{i=0}^{M-1} T(i,j)^2}} \quad (8)$$

where D is an image resulting from decryption with image correction and T is the image from decryption with 100% accuracy. NCC produces a value between 0.0 and 1.0, with 1.0 representing the highest level of accuracy. In our study, we took photographs for a span of one minute, the measured accuracy being the mean NCC of all frames. We used this approach since the NCC of one frame might have a high value even if the NCC of the previous frame had a low value.

Next, we moved the camera 10 cm and captured the image again until the encrypted image could not be recognized. Similarly, mean NCC values were calculated. This same procedure was also conducted from 30 and 45 degree angles, as depicted in Figure 14.

Figure 15 shows the resulting graphs for all accuracies. Naturally, all the accuracies gradually decreased with greater photographing distances. The graphs break off since the encrypted image is not recognized under the conditions described in section 4.2. However, in particular, we do not know the criteria for inherently good NCC values for reading characters, thus to determine these criteria, Figure 16 shows average images of 100 decrypted images with the same NCC values. As can be seen from these images, we confirm that a character can be read if the NCC is greater than or equal to 0.7. Therefore, Figure 15 shows our proposed recognition method to be effective when the photographing distance is less than approximately 100 cm.

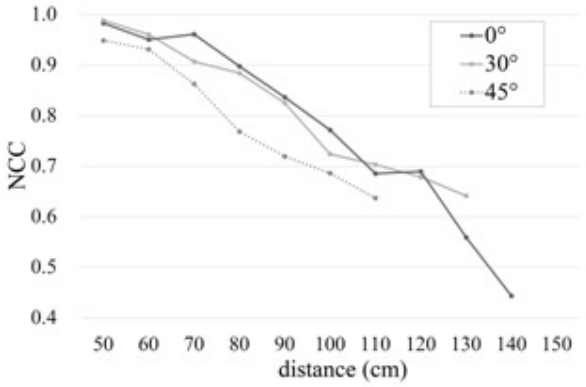


Figure 15. Normalized cross correlation (NCC) measures at each shooting distance.

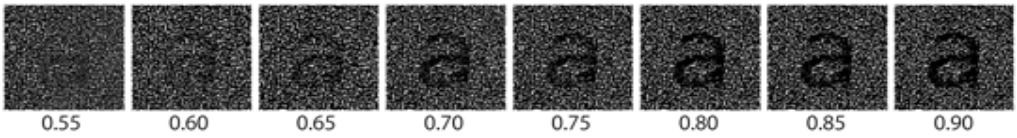


Figure 16. Average images of 100 decrypted images at varying NCC values.

5.3. COMPUTATIONAL COST

To determine computational costs, we implemented our proposed method on a Surface Pro 3 and measured computation time. Overall computation time per flow from image capture to decryption was 30.7 ms. In addition, the time required for image capture was as much as 50% of the overall processing time. In other words, our proposed method consumed only approximately 15 ms, thus we confirmed that computational cost of our proposed recognition method is very low.

5.4. DISCUSSION

The system described in section 3.1 above is a common-key cryptosystem. Such a system faces the problem of spoofing when the common key is stolen or leaked. However, since the common key is an image composed of a random pattern, updating the common key is easy when it is stolen or leaked. This has also been mentioned by Kato and Imai [2]. In addition, anyone around cannot see what the user is doing because the question itself is encrypted; that is, the user might be inputting a password or answering secret questions or both.

In practice, current systems employing a one-time password or secret question are very simple. For example, a one-time password is sent to a user in an e-mail, and the user inputs the password and is authenticated. In this way, it is possible to authenticate a user on only one device. In contrast, our proposed method requires two devices, a display, and a camera. Therefore, it can be used in such situations as online banking and other Web services on a personal computer. In addition, since the calculation cost is low, our proposed method can be implemented on various digital devices. Automatically recognizing an encrypted image, our proposed method can be easily used, similar to the ease of use experienced with a QR code. In the near future, we expect our technique to be implemented on wearable glasses-like devices, with text being decryptable simply by looking at the encrypted image.

For security, using our method, it is still possible to use a different password or a secret question every time, but with the added benefit of having no text in the given communication save for user input. In addition, we confirmed that users needed to take photographs at a distance of less than

100 cm to accurately read the hidden characters. In other words, there is no security threat from people who are more than 100 cm away from the display.

6. CONCLUSIONS

In this paper, we presented a decryption method for text encrypted with VC using digital devices. We explained our decryption system, discussed how to display text, and determined the optimal resolution of an image for our system. In addition, for user convenience, we proposed an automatic recognition method for the target region of an encrypted image. Our system displays only one character of text at regular time intervals. In addition, we determined that the minimal resolution for recognizing a character was 100×100; thus, systems can encrypt longer text, such as one-time passwords or secret questions.

Our proposed recognition method also included dedicated binarization. Using this, only the target region of the encrypted image can be recognized. In addition, to improve decryption accuracy, the extracted region was corrected using feature of shares of VC.

Our experimental results showed that our proposed method achieved a sufficiently high level of automatic recognition accuracy. In addition, we confirmed that users must take photographs within 100 cm to accurately recognize characters. Furthermore, as computational cost is low, we expect our system to be easily implemented on various devices, including wearable devices, which is part of our future work.

Overall, a common-key cryptosystem must be provided with a means of ensuring that the common-key image is not stolen. Further, in the future, we plan to build our proposed system, then develop and evaluate various real-world applications.

REFERENCES

[1] Naor, M., Shamir, A., (1994) “Visual cryptography”, Advances in Cryptology-EUROCRYPT'94, LNCS950, pp.1-12.

[2] Kato, T., Imai, H., (1996) “An Extended Construction Method of Visual Secret Sharing Scheme”, IEICE Transactions on Fundamentals, Vol.J79-A, No.8, pp.1344-1351. (in Japanese)

[3] Shamir, A., (1979) “How to share a secret”, Communications of the ACM, Vol.22, No.11, pp.612-613.

[4] Weng, IC., Chen, TH., (2017) “A Novel Weighted Visual Cryptography Scheme with High Visual Quality”, International Journal of Network Security, Vol.19, No.6, pp.922-928.

[5] Shaikh, R., Siddh, S., Ravekar, T., Sugaokar, S., (2016) “Visual Cryptography Survey”, International Journal of Computer Applications, Vol.134, No.2, pp.10-12.

[6] Blundo, C., Santis, A. D., Naor, M., (2000) “Visual cryptography for grey level images”, Information Processing Letters, Vol.75, No.6, pp. 255-259.

[7] Dharwadkar, N. V., Amberker, B. B., Joshi, S. R., (2009) “Visual Cryptography for Gray-Level Image using Adaptive Order Dither Technique”, Journal of Applied Computer Science & Mathematics, Vol.3, No.6, pp.60-65.

[8] Iwamoto, M., Yamamoto, H., (2002) “The optimal n-out-of-n visual secret sharing scheme for gray-scale images”, IEICE Transactions on Fundamentals, Vol.E85-A, No.10, pp.2223-2247.

[9] Lin, C. C., Tsai, W. H., (2003) “Visual cryptography for gray-level images by dithering techniques”, Pattern Recognition Letters, Vol.24, No.1-3, pp. 349-358.

[10] Su, PC., Tsai, TF., Chien, YC., (2017) “Visual secret sharing in halftone images by multi-scale error diffusion”, Multimedia Tools and Applications, pp.1-28.

[11] Hou, Y. C., (2003) “Visual Cryptography for Color Images”, Pattern Recognition, Vol. 36, No.7, pp.1619-1629.

[12] Koga, H., Yamamoto, H., (1998) "Proposal of a Lattice-Based Visual Secret Sharing Scheme for Color and Gray-Scale Images", IEICE Transactions on Fundamentals, Vol.E81-A, No.6, pp.1262-1269.

[13] Patil, S., Rao, J., (2012) "Extended Visual Cryptography for Color Shares using Random Number Generators", International Journal of Advanced Research in Computer and Communication Engineering, Vol.1, No.6, pp.399-410.

[14] Cimato, S., Yang, C. N., (2011) "Visual Cryptography and Secret Image Sharing", Digital Imaging and Computer Vision, CRC Press, pp.95-126.

[15] Nakajima, M., Yamaguchi, Y., (2002) "Extended Visual Cryptography for Natural Images", International Conferences in Central Europe on Computer Graphics, Visualization and Computer Vision, Vo.10, No.2, pp.303-310.

[16] Wang, C. C., Tai, S. C., Yu, C. S., (2000) "Repeating Image Watermarking Technique by the Visual Cryptography", IEICE Transactions on Fundamentals, Vol.E83-A, No.8, pp.1589-1598.

[17] Cimato, S., Yang, JC. N., Wu, C. C., (2014) "Visual Cryptography Based Watermarking", Transactions on Data Hiding and Multimedia Security, Vol.9, pp.91-109.

[18] Dobrev, P., Rosenthal, P., Linsen, L., (2010) "An Image-space Approach to Interactive Point Cloud Rendering Including Shadows and Transparency", Computer Graphics and Geometry, Vol.12, No.3, pp.2-25.

[19] Suzuki, S., (1985) "Topological Structural Analysis of Digitized Binary Images by Border Following", Computer Vision, Graphics, and Image Processing, Vol.30, No.1, pp.32-46.

AUTHOR

Naoyuki Awano Received The BIS, MIS, And DIS Degrees From Osaka Institute Of Technology, Osaka, Japan, In 2007, 2009, And 2012, Respectively. He Was A Research Associate At Osaka Institute Of Technology In 2009-2013. Since 2013, He Has Been An Assistant Professor At Seikei University. His Research Interests Are In The Areas Of Image Processing And Computer Graphics.