A Robust Secret Sharing QR Code via Texture Pattern Design

Xin Zhang, Jia Duan and Jiantao Zhou
Department of Computer and Information Science
University of Macau, Macau, China
Emails: mb65509@umac.mo, xuelandj@gmail.com, jtzhou@umac.mo

***Abstract*—The quick response (QR) codes have been widely used in a variety of applications, due to their fast readability and robustness against various distortions. In this work, we propose a robust secret sharing QR code scheme, which allows multiple users to share one secret through their QR codes. The QR code of each user, even in the printed/scanned form, is decodable by a standard QR decoder, and can be combined with other shared QR codes to recover the secret message through a correlation mechanism. A key factor leading to the success of our proposed scheme is to replace the modules of the standard QR codes with elaborately designed texture patterns. Extensive experimental results are provided to show the superior performance of our method.**

I. Introduction

The QR codes were initially designed for the Japanese automotive industry by Denso Wave corporation in 1994 [1]. They have now been used in a variety of scenarios including information storage, redirection to the website, identification *etc*. Due to their excellent performance in reading speed, robustness against various distortions, and relatively high capacity, QR codes have gained increasing popularity.

Considering the widespread use of QR codes, many methods took them as the carriers of secret information. Chuang *et al.* [2] first proposed a secret sharing technique using QR codes. The secret data was divided into several shadows by the secret sharing mechanism and then the shadows were embedded into QR codes. However, those QR codes stored the secret shadows without any protection measures. Lin *et al.* [3], [4] designed a secret QR sharing approach capitalizing on the error correction capability of the QR codes. Unfortunately, the secret storage capacity of these schemes was highly constrained by the number of error correction codewords. To let the QR code store one more level of information while not affecting the standard QR code decoding, Tkachenko I *et al.* [5] presented the so-called two-level QR code by intentionally changing the black modules of the standard QR code. As a result, the storage capacity was limited by the version size of the QR code. In addition, Cheng *et al.* [6] suggested a secret sharing QR code based on visual cryptography principle [7]. Essentially, they used the XOR operations for extracting the secret QR codes. Because of the high sensitivities of the XOR operations against even small disturbs, this method had high requirement on the display resolution in which the QR codes are presented.

In this work, we design a novel scheme for sharing secret messages among multiple users based on QR codes, where the black and white modules[1] are replaced by the sophisticatedly designed texture patterns. Here, the texture pattern represents the pure black or white modules with a small modification of white or black pixels. Our scheme can make the standard QR codes store one more level of information and have no effect on the modified QR codes being decoded by the standard QR decoder as well. Moreover, the proposed scheme is very robust and can readily be applied for both digital and analog version (printed or scanned). The contributions of our proposed work can be summarized as follows:

- The proposed scheme does not affect the readability of the public messages of the secret sharing QR codes. Namely, the public message can directly be decoded by the standard QR decoder.
- We elaborately design the texture pattern database to enhance the safety of the secret message. The texture patterns are randomly chosen from the database to replace the modules of cover QR codes.
- We propose a correlation-based decoding algorithm without involving any logical operations among the QR codes. This makes our scheme more robust to the display resolution comparing with those schemes with XOR operations.

The rest of this paper is organized as follows. In Section II, we briefly review the structure of the standard QR code. In Section III, the proposed method consisting of QR code generation and recovery processes are described. The experimental results are reported in Section IV. We finally conclude this work in Section V.

II. QR Code

The QR code is a type of two-dimensional barcode first designed for the automotive industry in Japan. A QR code uses four standardized encoding modes (*e.g.* numeric, alphanumeric, byte/binary, and kanji) to efficiently store data [8]. Each QR Code consists of an encoding region and function patterns

This work was supported in part by the Macau Science and Technology Development Fund under Grant FDCT/022/2017/A1, and in part by the Research Committee, University of Macau, under Grants MYRG2016-00137-FST and MYRG2018-00029-FST.

[1]According to the barcode terminology, the image block used to represent bit "0" or "1" is called a *module*. It corresponds to a smallest square unit in the binary black-and-white patterns of a QR code.

Fig. 1: The basic structure of the QR code

(*i.e.* finder, timing, and alignment patterns) [9]. The function patterns are designed to ensure the detection and decoding robustness of the QR code. Fig. 1 shows the basic structure of the QR code. Three finder patterns are used for QR code detection and orientation correction. The module coordinates are set by timing patterns. Alignment patterns are used for deformation adjustment. The format information areas contain error correction level and mask pattern. The code version and error correction bits are stored in the version information areas. QR code versions are referred to the form "Version V-E" where "V" identifies the version number (1-40) and "E" indicates the error correction level (L, M, Q, H). From Version 1 to Version 40, the size of the QR code increases from 21×21 to 177×177 modules gradually. Resorting to the error correction level, the QR code can be decoded when 7%(L) to 30%(H) codewords are corrupted [10].

To encode the data into QR code, the input data is encoded by the Reed-Solomon error correction code and then adds padding bits and remainder bits. The bit string is placed in the data region according to the zig-zag order. To make the black and white modules uniformly distributed in the data region, a mask pattern is chosen to be applied. Afterward, the standard QR code is generated.

The QR code recognition process includes the scanning process, image binarization, geometrical correction and decoding algorithm of error correction code. Due to the page limit, we refer the readers to [8] for more details.

III. Proposed Secret Sharing QR Code

In this section, we describe the proposed secret sharing QR code algorithm. We first build up a texture pattern database. Then we present the secret sharing QR code generation steps by exploiting the established texture pattern database. Afterwards, the secret QR code recovery process is given.

A. Texture Pattern Database Construction

The secret concealing and robustness of our proposed method are achieved by replacing the black and white modules by our designed texture patterns. Considering the fact that QR codes are usually in the printed/scanned forms, the texture patterns in our designed database are robust to the display resolution which can ensure the decoding accuracy.

Let $\{P_k\}_{k=1}^{Q}$ be texture pattern database, in which each pattern P_k is a binary image of size $a \times a$ pixels. All the texture patterns can be divided into two classes: black texture patterns and white texture ones. For each black (white) texture pattern, the percentage of white (black) pixels can be up to $r \approx 18\%$. According to the source code of ZXing library [10], $r \leq 18\%$ will not disrupt the standard QR code decoding, in this paper we call r the allowable modification rate. Moreover, the designed patterns should meet the following conditions.

1) The intra-correlation between texture patterns P_i and P_j should be very low, where $i, j \in \{1, \cdots, Q\}$ and $i \neq j$. Formally, it can be expressed as:

$$corr2(P_i, P_j) \ll 1, \forall i, j \in \{1, \cdots, Q\}, i \neq j. \tag{1}$$

where we utilize the 2-D correlation coefficient [5]

$$corr2(P_i, P_j) = \frac{\sum_w \sum_h (P_i^*(w,h))(P_j^*(w,h))}{\sqrt{\sum_w \sum_h (P_i^*(w,h))^2} \sqrt{\sum_w \sum_h (P_j^*(w,h))^2}} \tag{2}$$

Here, $P_i^*(w,h)$ are the central values of P_i defined by

$$P_i^*(w,h) = P_i(w,h) - \frac{1}{a^2} \sum_w \sum_h (P_i(w,h)). \tag{3}$$

2) The correlation between any two printed/scanned versions of the texture patterns A_i and A_j should be very low, where $i, j \in \{1, \cdots, Q\}$ and $i \neq j$. Formally, it can be expressed as:

$$corr2(A_i, A_j) \ll 1, \forall i, j \in \{1, \cdots, Q\}, i \neq j. \tag{4}$$

3) Each texture pattern P_i should have the highest correlation value with its printed/scanned version A_i, compared with all other printed/scanned versions A_j, where $i, j \in \{1, \cdots, Q\}$ and $i \neq j$ [5]. Namely, we have

$$corr2(P_i, A_i) = \max_{\forall j \in \{1, \cdots, Q\}} (corr2(P_i, A_j)). \tag{5}$$

4) Each A_i (printed/scanned version of P_i) should have the highest correlation value with P_i, compared with all other texture patterns P_j, where $i, j \in \{1, \cdots, Q\}$ and $i \neq j$ [5]. Namely, we have

$$corr2(A_i, P_i) = \max_{\forall j \in \{1, \cdots, Q\}} (corr2(A_i, P_j)). \tag{6}$$

We now describe how to construct the texture pattern database satisfying above conditions in practice. The size of the texture patterns is empirically set as $a \times a = 11 \times 11$. Referring to the allowable modification rate $r \approx 18\%$, we can choose at most 21 pixels to modify in each black or white module. We design plenty of alternative texture patterns by adding the shapes (*i.e.* triangle, rectangle) with intertwining black and white pixels to the white and black modules. The alternative texture patterns are different from each other as much as possible. By filtering the alternative texture patterns with aforementioned conditions, 47 texture patterns are selected (24 for black class, 23 for white class) in our experiment.

B. Secret Sharing QR Code Generation

The framework of the secret sharing QR code generation process is illustrated in Fig. 2. As can be seen, our scheme is

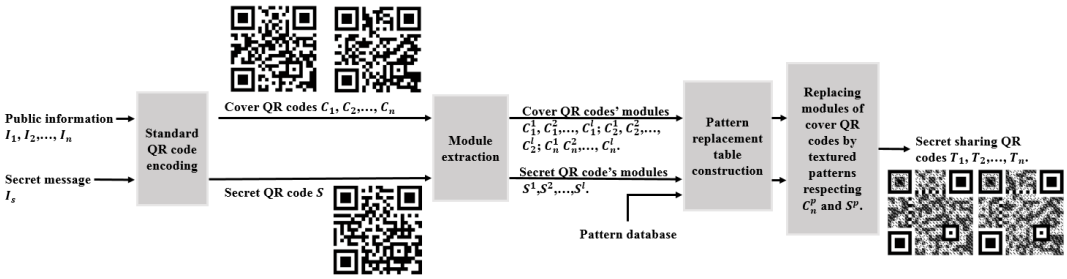


Fig. 2: The framework of n secret sharing QR codes generation process

composed of four steps: standard QR code encoding, modules extraction, pattern replacement table construction, and pattern replacement. For the ease of presentation, we take $n = 2$ as an example, where n is the number of shares. The discussion can be readily extended to the case of n ($n \gg 2$) shares. The inputs are two pieces of the public information (user-designed) I_1, I_2, and one secret message I_s. The outputs are two QR codes with texture patterns which are regarded as secret sharing QR codes T_1 and T_2.

Step 1 Standard QR code encoding: We encode the input information respectively by the standard QR code encoder. Here, the resulting QR codes of the public information I_1 and I_2 are named as cover QR codes C_1 and C_2. The secret QR code S is generated from the input secret message I_s in the same manner.

Step 2 Modules extraction: For each data region of QR code C_1, C_2, and S, we extract their modules according to the zig-zag order from bottom right to bottom left. Let C_1^p, C_2^p, and S^p represent the modules of C_1, C_2, and S. The superscript $p = 1, 2, ..., l$ (l is the number of modules of data region) denotes the index of the extracted module in the QR code.

Step 3 Pattern replacement table construction: We randomly choose q patterns from each texture pattern class (black and white) of our designed database. Since $C_1^p, C_2^p, S^p \in \{\text{“}black \quad module\text{”}, \text{“}white \quad module\text{”}\}$, there are totally eight different combinations of (C_1^p, C_2^p, S^p), which are shown in the first four columns of Table I. We take the second row of Table I as an example: C_1^p is black module, C_2^p is white module, and the S^p is black module. In this situation, T_1^p is chosen from the black class of the texture pattern according to the fact that the module of C_1^p is black. T_2^p is chosen from the white class due to the same reason. The selections of the texture patterns in other situations are in a similar way. Note that the combination of (T_1^p,T_2^p) should be unique in each situation. To meet the requirement that each pair of texture patterns is unique, the smallest value of q is two. The largest value of q is eight because each texture patterns can be different in the table.

Step 4 Pattern replacement: According to the combination of extracted modules C_1^p, C_2^p, and S^p, we replace the modules of cover QR codes C_1^p and C_2^p by T_1^p and T_2^p by referring to the Table I. The secret sharing QR codes are finally generated after all modules of data region are replaced by the texture patterns. Fixed patterns including finder patterns, alignment patterns, and timing patterns keep unchanged in order to avoid the secret QR code modules from being inferred from those patterns.

TABLE I: Pattern replacement table

NO.	C_1^p	C_2^p	S^p	T_1^p	T_2^p
1	■	■	■	▚	▚
2	■	□	■	▚	◥
3	□	■	■	◥	▚
4	□	□	■	◥	◢
5	□	□	□	◥	◥
6	□	■	□	◢	▚
7	■	□	□	▚	◢
8	■	■	□	▚	▚

(a) (b)

Fig. 3: (a) Original position tag. (b) Finder pattern after embedding.

Furthermore, we embed the selected texture patterns in the upper left finder pattern, which are called reference patterns. The illustration is given in Fig. 3. The black texture patterns replace the inner part of the position tag, the white patterns replace the white modules of position tags. As will be clear soon, the reference patterns play an important role in the secret recovery process.

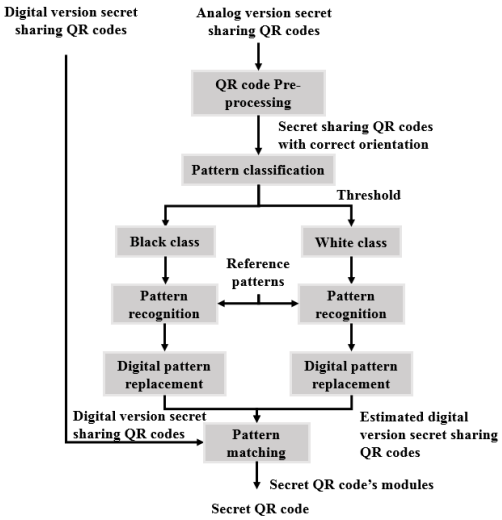


Fig. 4: Secret QR code recovery process

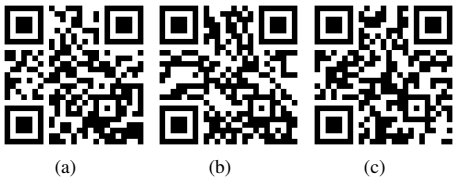


Fig. 5: QR codes generated by the standard QR code generator. (a) cover QR code C_1, (b) cover QR code C_2, (c) secret QR code S.

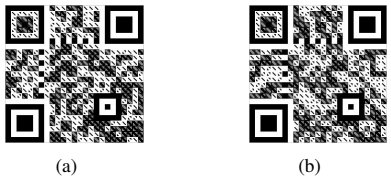


Fig. 6: Secret sharing QR codes generated by our proposed method. (a) secret sharing QR code T_1, (b) secret sharing QR code T_2.

C. Secret QR Code Recovery

The overview of the decoding process of secret sharing QR code is shown in Fig. 4. The inputs are the available secret sharing QR codes T_1 and T_2, either in their original digital version or in printed/scanned form. The output is the secret QR code S. Clearly, the public information of the secret sharing QR codes can be decoded by the standard QR code decoder. In the below, we assume that T_1 and T_2 are in the printed/scanned forms; the recover process when T_1 and T_2 are digital is simply a sub-process.

Step 1 Correcting the geometric distortion: The geometric distortion of the printed/scanned version of secret sharing QR code will be corrected by QR code preprocessing step. We get their original size $N \times N$ ($N = a \times m$, where $m \times m$ denotes the version size of the QR code) pixels by using fixed patterns to determine the position coordinates [9].

Step 2 Pattern classification: In this step, we extract the two classes of reference patterns (black and white classes of texture patterns) by thresholding method. The threshold here is calculated as a mean value of the whole texture patterns in the secret sharing QR code. If the mean value of the module is smaller than the threshold, the module will be categorized into the black class. Otherwise, the white class.

Step 3 Pattern recognition and Digital pattern replacement: We extract the reference patterns embedded in the upper left position tag. Then, we calculate the correlation values between the reference patterns and the texture patterns of the corresponding class. The texture pattern of the secret sharing QR code will be replaced by the digital version texture pattern with the largest correlation coefficient. After this step, the digital version of secret sharing QR code is produced.

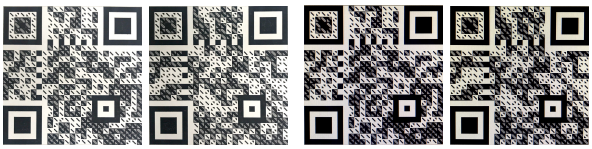
Step 4 Pattern matching: When all of the secret sharing QR codes are collected, according to different combinations of corresponding texture patterns of the secret sharing QR codes, the concealed secret modules can be recovered, which can also be considered as the inverse of the generated pattern replacement table in the secret QR code generation process.

IV. Experimental Result

In this section, we evaluate the applicability and performance of our proposed method. The open source library ZXing [10] with MATLAB is employed to generate and recover the multiformat QR codes.

We firstly conduct the experiments on readability of our proposed method. As described in Section III-B, the secret sharing QR code generation process consists of QR codes generation, modules extraction, pattern replacement table construction and patterns replacement steps. We use ZXing to generate cover QR codes C_1, C_2, and secret QR code S which stores the public information $I_1 = \{$The brand name: Starbucks$\}$, $I_2 = \{$Membership NO. 11111$\}$, and secret message $I_s = \{$Discount level: 30% off$\}$, respectively. Here, the "Version 2-L" with size of 25×25 is adopted. The generated QR codes are shown in Fig. 5. According to the texture pattern in Table I and secret QR code S, we replace the black and white modules of cover QR codes C_1 and C_2 to obtain the secret sharing QR codes T_1 and T_2, which are shown in Fig. 6.

Since the digital version has no distortion and noise, the secret QR code can be well recovered. In the following experiments, we conduct the recovery process on the printed/scanned version of secret sharing QR codes. To be consistent with real-world applications, we consider two kinds of printed/scanned scenarios: captured images from printed QR codes and captured images from screen-shown QR codes. After deformation adjustment, the printed/scanned versions of secret sharing QR codes T_1 and T_2 without geometrical distortion are shown in Fig. 7.



(a) (b) (c) (d)

Fig. 7: The printed/scanned versions of secret sharing QR codes. (a) captured image from printed T_1, (b) captured image from printed T_2, (c) captured image from screen shown T_1, (d) captured image from screen shown T_2.

To evaluate the recovery performance of our proposed method, we firstly choose 20 pairs of public information $\{I_1, I_2\}$ and secret message I_s. Then we utilize our proposed generation method to generate 20 pairs of secret sharing QR codes $\{T_1, T_2\}$. Also, we print or display 20 pairs of QR codes $\{T_1, T_2\}$ and obtain the printed/scanned version by capturing. At last, we employ the proposed recovery method to decode these secret sharing QR codes. Table II shows the accuracy of pattern classification, texture pattern recognition, pattern matching and secret message recovery. Specifically, for the digital version, the accuracy in each step is 100% and the secret message can be perfectly recovered. With respect to the printed/scanned version, the pattern can also be perfectly classified because small ratio ($r \approx 18\%$) of modification will not affect the black/white pattern classification. The accuracies of texture pattern recognition are 99.42% and 99.29% respectively, which is caused by the incorrect detection of white texture patterns. The incorrect texture pattern recognition also leads to the errors of pattern matching. However, the secret message can be recovered perfectly because of the correction capability of error correction code.

We also investigate the secret storage capacity of our proposed method and compare it with other methods [4], [6]. As shown in Fig. 8, the secret storage capacity of the method proposed in [4] is lower than 20% because this method stored the secret by capitalizing on the error correction capability. In other word, the length of the secret message is constrained by the number of error correction codewords. Additionally, the secret storage capacity of our proposed method is the same as that of the method in [6], because they both conceal the secret QR codes by utilizing the relationship among corresponding modules. Although the secret storage capacity is the same, our proposed method is robust to the printed/scanned QR codes, which is more practical than the method in [6].

TABLE II: The accuracies of key-steps in secret QR code recovery process

Accuracy	Digital version	Printed/scanned version	
		Scanned	Screen-shown
Pattern classification	100%	100%	100%
texture pattern recognition	100%	99.42%	99.29%
Pattern matching	100%	98.49%	98.37%
Secret message	100%	100%	100%

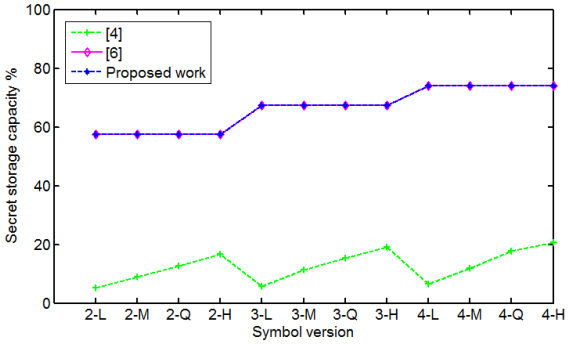


Fig. 8: The comparation of storage capacity

V. Conclusion

In this paper, we propose a novel secret sharing method based on QR code. By modifying the modules in QR codes, we conceal the secret into several QR codes to allow multiple users share one secret. The proposed scheme achieves the readability, robustness, and adjustable secret capacity. Extensive experimental results show the satisfactory of our method comparing to other methods. In the real world, the proposed method can be applied to some applications such as the e-coupon, e-ticket, airline luggage inspection and so on.

References

[1] A. Sun, Y. Sun, and C. Liu, "The QR-code reorganization in illegible snapshots taken by mobile phones," *Int. Conf. Computational Science and its Applications,* 2007, pp. 532-538.

[2] J. C. Chuang, Y. C. Hu, and J. H Ko, "A novel secret sharing technique using QR code," *Int. Journal of Image Processing,* vol. 4, no. 5, pp. 468-475, 2010.

[3] P. Y. Lin, "Distributed secret sharing approach with cheater prevention based on QR code," *IEEE Trans. Industrial Informatics,* vol. 12, no. 1, pp. 384-392, 2016.

[4] Y. J. Chiang, P, Y, Lin, R. Z. Wang, and Y. H. Chen, "Blind QR code steganographic approach based upon error correction capability," *KSII Trans. Internet Inf. Systems,* vol. 7, no. 10, pp. 2527-2543, 2013.

[5] I. Tkachenko, W. Puech, C. Destruel, O. Strauss, J.M. Gaudin, and C. Guichard, "Two-level QR code for private message sharing and document authentication," *IEEE Trans. Inf. Forensics Security,* vol. 11, no. 3, pp. 571-583, 2016.

[6] Y. Cheng, Z. Fu, B. Yu, and G. Shen, "A new two-level QR code with visual cryptography scheme," *Multimedia Tools and Applications,* pp. 1-21, 2017.

[7] Naor, Moni, Shamir, and Adi, "Visual cryptography," *Advances in Cryptology,* pp. 1-12, 1995.

[8] D. Wave, "QR Code features," Retrieved 3 October 2011.

[9] D. Wave, "Information technology automatic identification and data capture techniques QR code bar code symbology specification," *International Organization for Standardization,* ISO/IEC, 18004.

[10] Z. Team, "ZXing Zebra Crossing," 2015 [Online], Available: http://code.google.com/p/zxing/.