High Capacity Embedding Methods of QR Code Error Correction

Song Wan(✉), Yuliang Lu, Xuehu Yan, Wanmeng Ding, and Hanlin Liu

Hefei Electronic Engineering Institute, Hefei 230037, China
wsong1031@163.com

Abstract. In this paper, two methods about how to embed message into QR code are investigated. According to different application scenarios, two different embedding ways are given. The first proposed embedding way is to modify a continuous region based on the arrangement of codewords in QR code and the mechanism of QR code error correction which can reach the maximum error correction capability as well as scan the QR code altered by a QR code reader. The second embedding way is designed to modify each column separately in coding regions which can be decoded correctly as well. Although the second embedding way couldn't reach high capacity, it can be applied in many occasions while the first embedding way couldn't. Based on the proposed two embedding methods and the analysis of the error correction mechanism, we conclude the general rules about how to embed message into QR code. The experiment results show the effectiveness of our methods.

Keywords: QR code · Error correction · Embedding capacity

1 Introduction

QR code [1] which are widely used as a means of conveying textual information, such as hyperlinks, emails, or phone numbers, through images that are interpreted using a smartphone camera are a popularly used two-dimensional barcode recently with the advantages of larger QR content and error correction capability. Even if it is dirty, we may be able to read it since it has error correcting capability. Additional to the characteristics for QR code such as high-speed reading, high-density recording (approx. 100 times higher in density than linear symbols), and large volume data (7,089 numerical characters at maximum), QR code has other superiority in both functionalities and performance aspects [2].

Recently, many researchers have proposed some schemes [3] exploiting the error correction mechanism inherent in the QR code structure to implement different applications [4]. For example, a scheme [5] which deeply integrate the error correction mechanism of QR code and the theory of Visual Secret Sharing (VSS). The scheme generates the bits corresponding to shares by VSS from a secret bit in the processing of encoding QR. Each share is a valid QR code that can be scanned and decoded by a QR code reader. The shares may not be suspected if distributed via public channels and will reduce the likelihood of attracting the

M. Huang et al. (Eds.): WICON 2016, LNICST 214, pp. 70–79, 2018.
https://doi.org/10.1007/978-3-319-72998-5_8

attention of potential attackers. Based on this, this kind of schemes has great application value. So, the applications are valuable that exploiting the error correction mechanism inherent in the QR code structure. If we want to complete the kind of schemes, the mechanism of QR codes error correction need to be understand. In addition, it is the most important that how to embed message into QR code which can reach the maximum error correction capability within the error correction level. However, there is no method to all QR code versions about how to embed message into QR code which can reach the maximum error correction capability satisfying that the QR code altered can still be decoded.

In this article, we propose two methods about how to embed message into QR code. Due to different application scenarios, two different embedding ways are given. The first embedding way modifies a continuous region of QR code which can reach the maximum error correction capability within the error correction level for all QR code versions. In addition, the output QR code can be so as to scanned correctly. The second embedding way is to modify the modules through separating columns in coding regions that the QR code altered can still be decoded for all QR code versions as well. Although this embedding way couldn't reach high capacity, it can be applied in many occasions while the first embedding way couldn't.

The remainder of the paper is organized as follows. The introduction to the QR codes are presented in Sect. 2. The proposed process described in Sect. 3. Section 4 demonstrates the simulation results and analyses. Finally, Sect. 5 concludes this paper.

2 Background

2.1 QR Code

QR code which was invented by the Denso Wave [6] Incorporated in 1994 is defined as a two-dimensional barcode. The standard [1] defines forty sizes of QR code symbol versions which range from version 1 to version 40. A QR code is divided into modules and each QR code symbol version is comprised of a different number of modules. Each version has four modules more than the previous one. For example, Version 1 is made up of 21×21 modules while version 2 is made up of 25×25 modules. The QR code structure consists of function patterns and encoding regions. The encoding region consists of error correction and data codewords, version information and format information while function patterns consist of the alignment patterns, timing, separators patterns and finder patterns. The structure of a QR code version 7 is illustrated in Fig. 1. Each QR code has three Finder Patterns which are located in the lower left, upper left and upper right corner. They are used to recognize the QR code and detect the position of the symbol. Alignment Patterns that only occur from version two up to forty permit QR code readers to compensate for image distortion and the higher the version is, the more Alignment Patterns exist. A quiet zone which is the blank area around QR code is necessary for reading the QR code. It should have the same reflectance value as the light modules, because the QR code readers could not distinguish between the Finder Patterns and the dark background. Timing patterns are used to determine the module coordinates and the separators which are one module wide are used to separate the finder patterns from the encoding region.

The data in QR code is encoded into the binary numbers of "1" and "0" based on Reed-Solomon codes that used for error detection and correction. The bit stream which is generated by message data encoded is divided into a sequence of codewords that are 8 bits length. There are four different error correction levels (L ∼ 7%, M ∼ 15%, Q ∼ 25%, H ∼ 30%). The error correction [7–9] is used for recovering the QR code in the event that parts of the symbol are dirty or destroyed. So, the QR code can also be recognized when embed other information in it. The recovery capacity of QR code will be improved by using the higher error correction levels, but it will increase the amount of data to be encoded. It means that a larger QR code version may be required when using a higher error correction level to encode the same message.

To minimize the possibility that localized damage will cause the QR code become undecodable, the codewords from the blocks are encoded in an interleaved manner, with the error correction codewords appended to the end of the data codewords sequence. The data codewords and error correction codewords arrangement for QR code version 7, with an error correction level of H, is shown in Fig. 1.

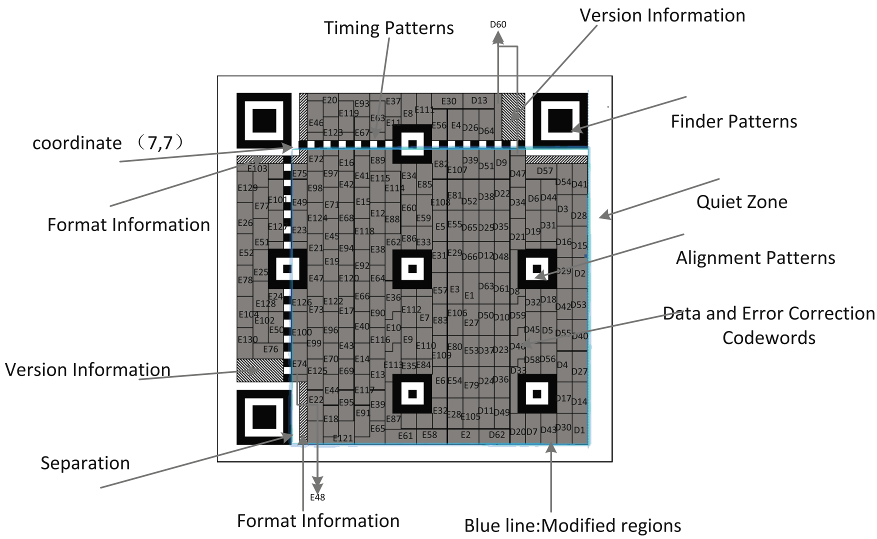


Fig. 1. The structure and codewords arrangement for QR code version 7 with error correction level H

3 Proposed High Capacity Embedding Methods

In this section, two methods about how to embed message into QR codes which can reach the maximum error correction capability are proposed. According to different application scenarios, two different embedding ways are given.

The first way about how to embed is a continuous region altered. The region altered can be a rectangle from the coordinate of $(7, p-j)$ to the lower right corner while the value of j can be determined by the error correction capacity of the QR code and the codewords altered can reach the maximum error correction capacity. The theoretical derivation is as follow.

The QR codes can be correctly decoded while the codewords altered reach the maximum that the error correction level allowed. In this paper, the methods proposed are to use the highest level of QR codes error correction.

Given a cover QR code with error correction level of H, and the embed message S with n bits, the length of n is determined by the error correction capacity of the QR code as

$$n = \left\lfloor \frac{m-e}{2} \right\rfloor \times 8 \leq \left\lfloor \frac{m}{2} \right\rfloor \times 8 \tag{1}$$

Here, m is the number of error correction codewords, e is the number of error detection codewords, and a codeword equals to eight modules in the QR codes structure. The maximum number of codewords that can be altered in QR code are referred to as l, where $l = n/8$.

The encoded data before data mask is stored in an array module of QR codes from left to right. The coordinate of $(0, 0)$ represents the top left corner of the array module. The three identical finder patterns which are made up of 7×7 modules are used to recognize the QR code and to determine the rotational orientation of the symbol. In order to make sure that the three identical finder patterns of QR code could not be altered so that it does not affect the appearance, we define the data which is blue region as shown in Fig. 1 that could be modified is from the coordinate of $(7, 7)$ to the last. Although some bits of the format information may be altered, It couldn't affect the format information [9]. So, the region we define is possible.

Due to the characteristics and basic principles of the design of QR code, it can be seen that the arrangement of the codewords are in two modules wide starting from the symbol of the lower right corner to decorate, from right to left, and alternately from the bottom up or down. In QR code with an error correction level H, the first codeword of each block is arranged, then the second codeword, until the last one. The codewords from the blocks are encoded in an interleaved manner, with the error correction codewords appended to the end of the data codewords sequence, as shown in Fig. 1. So, the left region of QR code is error correction codewords while the right region is data codewords. It is possible that localized damage would not make the QR code undecodable.

The error correction codewords for each block is given as (c, k, r), Here, c is the total number of codewords, k is the number of data codewords and r is the error correction capacity which represents the maximum number of codewords that can be altered per block. It means that if more than r codewords per block contain errors, the QR code would not be decoded. So, we must make sure that the codewords altered per block can't be more than r and the total codewords altered in QR code need to be equal to or less than l.

When we alter the codewords including data codewords and error correction codewords, it is hard to locate the position of each block so that can't make sure that the number of codewords altered per block is smaller than r or not. In this case, although we can control the total codewords altered in QR code less than l, the number of codewords altered per block couldn't guarantee. As the characteristics of QR code, it can be concluded that if the codewords altered all in data region or the whole in the error correction region, we can make measurers to guarantee the codewords altered per block wouldn't more than r. So, we can ensure that the number of codewords that can be altered would be infinitely close to the maximum. In conclusion, we will altered codewords all in data region or the whole in error correction region, which can reach the maximum number of codewords that can be altered. As the blue region chosen is from the coordinate of $(7, 7)$ to the last, there are some error correction codewords and data codewords that do not be included here and that the number of error correction codewords are much more than data codewords. Based on this, we will modify the codewords all in data region.

The problem now is how to determine the coordinates of the region altered that can reach the maximum and can be scanned correctly. Due to the arrangement of data codewords, the region altered can be a rectangle from the coordinate of $(7, p - j)$ to the lower right corner. The total codewords that can be altered must be equal to or less than l, so that the value of j can be determined. As can be seen that when meeting alignment patterns, the arrangement of codewords need to leave space for them. In other words, the same number of codewords will need bigger regions if containing alignment patterns with size of 5×5. So, when containing alignment patterns, the number of codewords of the region altered with size of n is less than l. Considering the relationship between the total number of alignment patterns contained in the region altered and the value of $(p-7)$, the arrangement of the data codewords, other possible errors that affect the QR code with high version scanned correctly, the range of j will be as follow:

$$\left\lfloor \frac{n}{p-7} \right\rfloor - 1 \leq j \leq \left\lfloor \frac{n}{p-7} \right\rfloor + 1 \tag{2}$$

Here, the coordinate of the lower right corner is (p, p), $(p - 7)$ represents the number of lines in the blue region. In this way, the region altered of QR code with version from two up to forty would reach the maximum that match the capacity of QR code with error correction H.

As alignment Patterns only occur from version two up to forty, if QR code version 1 with error correction H using the method above, the region can be altered wouldn't reach maximum. After analysis, we find the region altered that from the coordinate of $(7, 13)$ to $(17, 21)$ is maximum and the QR code can be decoded correctly.

The QR code error correction level of H $\sim$ 30%, is merely approximate value. The total number of codewords in each QR code version can be referred to as N, where $N = m + t$, t is the total number of data codewords. To analyze the error correction blocks of 40 versions, we can find that only version 1 of QR code need one error correction codeword for error detection and other versions do not. In QR code version 1 with error correction level H, there are 17 error correction codewords which can correct 8 data codewords. AS the total codewords are 26, the error correction rate is 30.7%. From version 2 to 40 of QR code, two error correction codewords can correct one data codeword, so the error correction rate of QR code H, will be referred to as

$$H = \frac{m}{2N} \tag{3}$$

Hence, the relationship between the total number of error correction codewords and data codewords can be defined as

$$\frac{m}{t} = \frac{2H}{1 - 2H} \tag{4}$$

The value of m/t range from 1.7 to 1.96 with the research of data codewords and error correction codewords from version 2 to 40 of QR code with error correction level H. Based on the value of m/t, the error correction rate of QR code with error correction level H is range from 31.5% to 33.2%. Theoretically, the error correction rate will range from 30.7% to 33.2%. As some alignment patterns in the region altered and the arrangement of the data codewords, sometimes the error correction rate may be larger than 33.2% or smaller than the rate of 30.7% as the method of the region altered and other possible errors. Although the rate is change, it is very small. The first embedding way would be applied in all versions of QR code and it can reach the maximum capacity.

The second embedding way is to separate columns in blue region which is easy to alter and wouldn't affect the appearance of QR codes, as shown in Fig. 1. In this case, we can control the total codewords altered in QR code less than l and guarantee the number of codewords altered per block wouldn't more than r. Although this embedding way couldn't reach high capacity, it can be applied in many occasions while the first embedding way couldn't.

4 Experimental Results and Analyses

In this section, the effectiveness of methods proposed are evaluated by our experiments. A large number of QR code versions with error correction level H are used to test the efficiency of the proposed methods.

4.1 Image Illustration

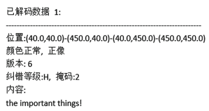
In our experiments, the simulation environment of the proposed process is python language. In the first embedding way, Fig. 2 shows the simulation results for the QR code version 6 with 41×41 modules and error correction H. Figure 2(a) is the cover QR code with the content "the important things", S. The region altered of QR code, SC_1, from the coordinate of $(7, 29)$ to $(41, 41)$ is shown in Fig. 2(b) while the region altered from the coordinate of $(7, 28)$ to $(41, 41)$ is shown in Fig. 2(c), SC_2. Figure 2(d)–(f) show the decoding information for S, SC_1, SC_2. It can be seen that the altered QR code shown in Fig. 2(c) can't be decoded correctly while the altered QR code shown in Fig. 2(b) can be scanned. So, the altered QR code shown in Fig. 2(b) reaches the maximum error correction capability.

Figure 3 demonstrates the results of the QR code version 40 with 177×177 modules and error correction level H. Figure 3(a) is the cover QR code, S, and the region altered of QR code, SC_1, from the coordinate of $(7, 121)$ to $(177, 177)$ is shown in Fig. 3(b). Figure 3(c) shows the region altered of QR code, SC_2, from the coordinate of $(7, 58)$ to $(85, 85)$. Figure 3(d)–(f) show the decoding information for SC_1, SC_2. The altered QR code shown in Fig. 3(c) can't be decoded correctly while the altered QR code shown in Fig. 3(b) can be scanned. So, the altered QR code shown in Fig. 3(b) reaches the maximum error correction capability.

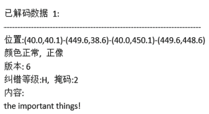
(a) Original QR S

(b) SC1

(c) SC2



(d) the decoding information for S



(e) the decoding information for SC1

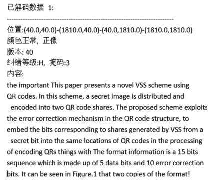
(f) the decoding information for SC2

Fig. 2. The results of QR code version 6 with error correction level H by the first embedding way proposed.

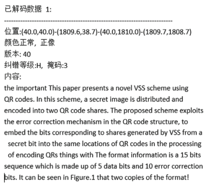
(a) Original QR S

(b) SC1

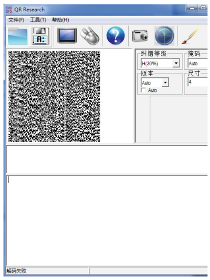
(c) SC2



(d) the decoding information for S



(e) the decoding information for SC1



(f) the decoding information for SC2

Fig. 3. The results of QR code version 40 with error correction level H by the first embedding way proposed.

Table 1. The payload and rate of region altered of the proposed first embedding way

Version	Size	N	m	l	j	payload	rate
1-H	441	26	17	8		80	38.46%
2-H	625	44	28	14	7	126	35.79%
3-H	841	70	44	22	8	176	31.42%
4-H	1089	100	64	32	10	260	32.5%
5-H	1369	134	88	44	12	360	33.58%
6-H	1681	172	112	56	12	408	29.65%
7-H	2025	196	130	65	14	532	33.92%
10-H	3249	346	224	112	18	900	32.5%
17-H	7225	815	532	266	26	2028	31.1%
20-H	9409	1085	700	350	30	2700	31.1%
27-H	15625	1828	1200	600	40	4720	32.27%
30-H	18769	2185	1440	720	44	5720	32.72%
35-H	24649	2876	1890	945	50	7500	32.59%
40-H	31329	3706	2430	1215	56	9520	32.11%

Table 2. The results of the second embedding way

The number of columns separated version-error correction level	4	5	6	7	8	9
1-H	No	Yes				
2-H,3-H,4-H,5-H,6-H,7-H,8-H,12-H,18-H	No	No	Yes			
9-H,10-H,11-H,13-H,15-H,17-H,19-H,20-H,21-H,23-H,24-H,25-H,26-H,27-H	No	No	No	Yes		
14-H,16-H,22-H,28-H,29-H,30-H,32-H,33-H,35-H,36-H,37-H,38-H,39-H	No	No	No	No	Yes	
31-H,34-H,40-H	No	No	No	No	No	Yes

Table 1 lists the payload of region altered of the proposed first embedding way under different QR versions with error correction H. As the QR code version 1 contains no alignment pattern is analyzed before, for the sake of brevity, we list several versions with different alignment patterns between the Version 2 (25×25 modules = 625 modules) and the largest Version 40 (177×177 modules = 31,329 modules). The designed mechanism exploits the characteristics of the error correction capability and the arrangement of codewords of the QR code to achieve the purpose.

Table 2 lists the results of the second embedding way under different QR code versions with error correction H. The operation of separating columns is in the blue region of all QR code versions as shown in Fig. 1. The value of columns separated in blue region is listed in table. 'Yes' represents the QR code altered can be decoded correctly while 'No' represent the QR code altered couldn't.

4.2 Analysis

From the statistical data, we can conclude that the region altered of the proposed first embedding way can reach the error correction capability. In this way, the number of modules that can be modified would reach almost the maximum with the error correction level H. From the second embedding way, it can be seen that the value of the columns separated is between five and night. In general, the value of the columns separated in the blue region of most QR code versions is range from six to eight and with the QR code version higher, the value of the columns separated would be bigger. Although the second embedding way couldn't reach high capacity, it can be applied in many occasions which may contain image information hiding and so on, when the first embedding way couldn't.

5 Conclusion and Future Work

This paper presents two methods about how to embed message into QR code. Due to different application scenarios, two different embedding ways are given. The first embedding way can reach the maximum error correction capability within the error correction level H for all QR code versions. In addition, the output QR code can be so as to scanned correctly. The second embedding way is to modify each column separately in coding regions that the QR code altered can still be decoded and would be applied in all QR code versions. In general, the value of the columns separated in the blue region of most QR code versions is range from six to eight. Although the second embedding way couldn't reach high capacity, it can be applied in many occasions while the first embedding way couldn't. The future work will be implemented some applications which exploiting the high capacity embedding methods of QR code error correction.

Acknowledgement. The authors would like to thank the anonymous reviewers for their valuable comments. This work is supported by the National Natural Science Foundation of China (Grant Number 61602491) and Hefei Electronic Engineering Institute Foundation (KY16A654).

References

1. Jtc1/Sc, I.: Information technology - automatic identification and data capture techniques - qr code 2005 bar code symbology specification (2006)
2. Várallyai, L.: From barcode to QR code applications. Agrárinformatika Folyóirat **3**(2) (2013)
3. Wang, G., Liu, F., Yan, W.Q.: 2D barcodes for visual cryptography. Multimedia Tools Appl. **75**(2), 1223–1241 (2016)
4. Kan, T.W., Teng, C.H., Chen, M.Y.: QR code based augmented reality applications. In: Furht, B. (ed.) Handbook of Augmented Reality. Springer, New York (2011). https://doi.org/10.1007/978-1-4614-0064-6_16
5. Wan, S., Lu, Y., Yan, X., Liu, L.: Visual secret sharing scheme with (k, n) threshold based on QR codes. In: 12th International Conference on Mobile Ad-hoc and Sensor Networks (2016)
6. Denso Wave Inc. (2002). http://www.qrcode.com
7. Yan, X., Guan, S., Niu, X.: Research on the capacity of error-correcting codes-based information hiding, pp. 1158–1161. IEEE Computer Society (2008)
8. Saito, K., Morii, M.: Efficient decoding of QR code using error correcting capability: Decoding method using erasure error correction and the ability. Technical Report of IEICE ISEC, vol. 111, pp. 79–84 (2011)
9. Chow, Y.-W., Susilo, W., Yang, G., Phillips, J.G., Pranata, I., Barmawi, A.M.: Exploiting the error correction mechanism in QR codes for secret sharing. In: Liu, J.K.K., Steinfeld, R. (eds.) ACISP 2016. LNCS, vol. 9722, pp. 409–425. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-40253-6_25