WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

QR IMAGES: OPTIMIZED IMAGE EMBEDDING IN QR CODES

[a] Dr. V Sangeetha, [b] A.Sriram

[a]Assistant Professor, Department of Computer Science, Karpagam Academy of Higher Education
[b]Department of Computer Science, Karpagam Academy of Higher Education

A B S T R A C T

This paper introduces the concept of QR images, an automatic method to embed QR codes into color images with bounded probability of detection error. These embeddings are compatible with standard decoding applications and can be applied to any color image with full area coverage. The QR information bits are encoded into the luminance values of the image, taking advantage of the immunity of QR readers against local luminance disturbances. To mitigate the visual distortion of the QR image, the algorithm utilizes half toning masks for the selection of modified pixels and nonlinear programming techniques to locally optimize luminance levels. A tractable model for the probability of error is developed and models of the human visual system are considered in the quality metric used to optimize the luminance levels of the QR image. To minimize the processing time, the optimization techniques proposed to consider the mechanics of a common Binarization method and are designed to be amenable for parallel implementations. Experimental results show the graceful degradation of the decoding rate and the perceptual quality as a function the embedding parameters. A visual comparison between the proposed and existing methods is presented.

Keywords: QR Codes, Encoding, Decoding, Binarization.

1. Introduction

The image below demonstrates all of the required sections of data modules, rendered in the QR Code. Each colorized section includes a description for its purpose in the final QR code rendering. Each one of these segments is obligatory, with the exception of "format information", which is introduced in version 7 and up. There are two types of such data. One is the "format data", which are modules such as the "alignment detection", as well as "position detection". These are important in allowing the decoder to know which version the code is, and what error correction level it has. The other type of data is the actual data being encoded. It is always complemented by the error correction bytes. These ensure that the QR code will be read successfully if some portion of it is damaged.

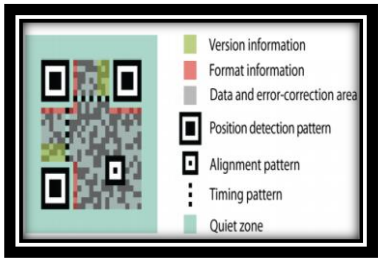

Fig- 1 structure of QR Code

In the above diagram the parts of the QR code pattern is represented with the corresponding colors. Using the color representation the important parts and information are highlighted so that this describes the full structure of the QR code.

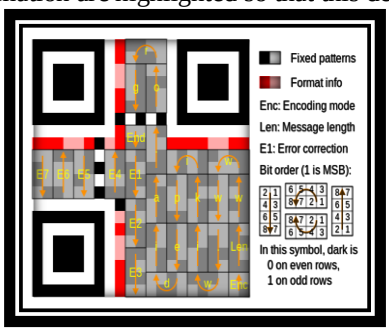

Fig- 2 message process of QR Code

The above diagram illustrates about how the message or data is placed in the QR Code. This Message placing process follows some pattern or order and the QR code has some fixed pattern for encoding the message into the code according to the version that is followed. In the diagram some of the data patterns are described with some color indication. The Most significant bit order is been followed to encode the data. As mentioned in the diagram Zeros are in the even rows and ones are in the odd row. MSB mechanism is explained with one example that describes about how the MSB is performed on the QR code. The Encoding Mode is represented as Enc. Length is represented as Len. The error correction codes are like E1, E2, and E3... And so on.

2. Methodology

1. The proposed system is based on the embedding operation of QR Image and hiding image.
2. The QR image is consists of the data pattern, retrieving pattern and so on.
3. When a secure QR code transmitted using web or internet then the QR code is not safe anybody can scan the image and then they can read the information.
4. Embedding: The embedding process involves the hiding procedure of the data content using the hiding image and hiding key.
5. Decoding:

Decoding process continues with three basic stages: Binarization, detection, and decoding of the bit stream. In the Binarization stage, the gray scale image captured by the camera is segmented into black and white pixels. This binary image is used to determine the QR modules centers and the sampling grid from which the code words are extracted

3. Results and Discussion

In our proposed system is effective and high robustness when compared to the Existing methods Based on proposed methods.

Fig- 4 Original Image

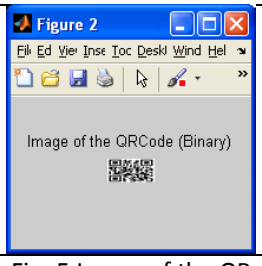

Fig- 5 Image of the QR Code

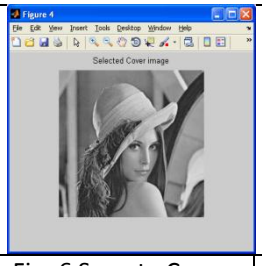

Fig- 6 Secrete Cover image

Fig- 7 Embedded image

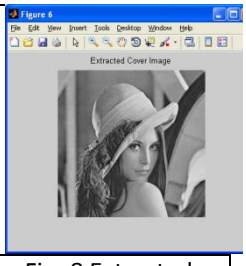

Fig- 8 Extracted cover image

4. Conclusion

Our work on attacking QR Codes explores a security issue that has not been within the scope of recent research projects and thus open new aspects in this area. We developed an implementation that attempts brute-forcing a QR Code. However, our results showed us that this task is not feasible in a short period. The execution time of our implementation indicates us that such an attack cannot be deployed in a real attack scenario. However, our second approach which targets on the binary representation of the encoded string was successful. We managed to produce the same result as a successful brute-forcing implementation would have and we showed that it is possible to alternate a QR Code so that it will lead to a new URL. To further explore possible uses of QR Codes involving malicious scenarios like a phishing attack, we deployed our empirical study. Our on-line survey was addressed to the users of four different European cities, allowing us in this way to work in an intercultural environment. Our research that is comparable to a social engineering experiment, achieved at least on a certain level its goal, which was to identify users level of security awareness concerning the security issues related to QR Codes.

REFERENCES

[1] The MathWorks Inc. MATLAB 7.0 (R14SP2). The MathWorks Inc., 2005.
[2] S. J. Chapman. MATLAB Programming for Engineers. Thomson, 2004.
[3] C. B. Moler. Numerical Computing with MATLAB. Siam, 2004.
[4] C. F. Van Loan. Introduction to Scientific Computing. Prentice Hall, 1997.
[5] D. J. Higham and N. J. Higham. MATLAB Guide. Siam, second edition edition, 2005.
[6] K. R. Coombes, B. R. Hunt, R. L. Lipsman, J. E. Osborn, and G. J. Stuck. Differential Equations with MATLAB. John Wiley and Sons, 2000.
[7] A. Gilat. MATLAB: An introduction with Applications. John Wiley and Sons, 2004.
[8] J. Cooper. A MATLAB Companion for Multivariable Calculus. Academic Press, 2001.
[9] J. C. Polking and D. Arnold. ODE using MATLAB. Prentice Hall, 2004.
[10] D. Kahaner, C. Moler, and S. Nash. Numerical Methods and Software. Prentice-Hall, 1989.
[11] J. W. Demmel. Applied Numerical Linear Algebra. Siam, 1997. [12] D. Houcque. Applications of MATLAB: Ordinary Differential Equations. Internal communication, Northwestern University, pages 1–12, 2005.
[12] Napoleon, D., and M. Praneesh. "Detection of Brain Tumor using Kernel Induced Possiblistic C-Means Clustering." *International Journal of Computer & Organization Trends, 3 (9)* (2013): 436-438