# Enhancing the Data Capacity of QR Codes by Compressing the Data before Generation

Nancy Victor
Centre for Information Technology and Engineering
M.S. University

## ABSTRACT

Quick Response (QR) Codes helps us in encoding the data in an efficient manner. The data capacity is limited according to the various data formats used. For increasing the data capacity, data to be encoded can first be compressed using any of the data compression techniques. Then, the data can be encoded. This paper suggests a technique for data compression which in turn helped to increase the data capacity of QR Codes. Results are compared with the normal QR Codes to find the efficiency of the new technique of encoding followed by compression.

## Keywords

2D barcodes, Data Capacity, Data Compression, Lossless Compression, QR Code

## 1. INTRODUCTION

Bar codes have become widely popular because of their reading speed, accuracy, and superior functionality characteristics. Barcodes can be divided as 1D and 2D. 1D barcodes can express information in horizontal direction only. Also, the data capacity is limited. 2D barcodes can hold data both in horizontal and vertical direction. As a result, the data capacity is 100 times more than the 1D barcode [1].

As bar codes became popular and their convenience universally recognized, the market began to call for codes capable of storing more information, more character types, and that could be printed in a smaller space. However, these improvements also caused problems such as enlarging the bar code area, complicating reading operations, and increasing printing cost. 2D Code emerged in response to these needs and problems [2].

QR Code is a kind of 2-D (two-dimensional) symbology developed by Denso Wave and released in 1994 with the primary aim of being a symbol that is interpreted by scanning equipment [3]. 2D bar codes can act like identifier (like in 1D) but takes less space. Alternatively, it can function as a database itself.

QR Code holds a considerably greater volume of information than a bar code. These can be numeric, alphanumeric or binary data – of which up to 2953 bytes can be stored. Only a part of each QR bar code contains actual data, including error correction information. A large area of the QR code is used for defining the data format and version as well as for positioning, alignment and timing purposes. The smallest square dot or pixel element of a QR code is called a module. QR Codes have an empty area around the graphic. This quiet area is ideally 4 modules wide. Examination certificates can also use the QR Encoding techniques [4].

This paper proposes a method in which data capacity can be increased by first compressing the data and then encoding it. A lossless compression technique is proposed to increase the data capacity. For decoding the data, two steps will have to be followed. The first step focuses on de-compressing the data using the techniques which are just the reverse of compression technique used here. The second step focuses on decoding the decompressed data. For this, the reverse technique used for encoding the data can be used.

## 2. LITERATURE SURVEY

QR Codes have already overtaken the conventional bar codes because of the main fact that the capacity of data that can be stored by a conventional bar code is very much less when compared to the data that can be stored by a 2-D barcode, the QR Code. QR Code contains data both in horizontal and vertical positions.

QR Codes have already overtaken the classical barcode in popularity in some areas. This stems in many cases from the fact that a typical barcode can only hold a maximum of 20 digits, whereas as QR Code can hold up to 7,089 characters [3]. QR Codes are capable of encoding the same amount of data in approximately one tenth the space of a traditional bar code. A great feature of QR Codes is that they do not need to be scanned from one particular angle, as QR Codes can be read regardless of their positioning. QR Codes can be easily decoded with a mobile phone with appropriate software (Kaywa Reader) [5]. Secure communication can also be established using QR Encoding techniques [6].

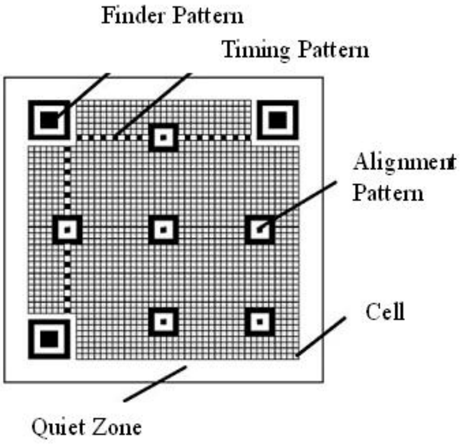


**Fig. 1 QR Code Structure**

## 2.1 Structure of QR Codes

QR Codes are actually black modules in square patterns on white background. QR Codes consists of many areas that have specific importance.

- Finder Pattern
- Alignment Pattern
- Timing Pattern
- Quiet Zone
- Data Area

Fig.1 shows the structure of QR Code. The importance of each area is as described as follows

Each QR Code symbol consists of mainly two regions: an encoding region and function patterns. Function patterns consist of finder, timing and alignment patterns which does not encode any data. The symbol is surrounded on all the four sides by a quiet zone border [7]. A QR Code can be read even if it is tilted or distorted. The size of a QR Code can vary from 21 x 21 cells to 177 x 177 cells by four cell increments in both horizontal and vertical direction.

### *2.1.1 Finder Pattern*

This pattern can be used for detecting the position of QR Code. The position, size and angle of the QR Code can be determined with the help of the three position detection patterns (Finder Patterns) which are arranged at the upper left, upper right and lower left corners of the symbol. The patterns can be easily detected in all directions.

### *2.1.2 Alignment Pattern*

The alignment pattern consists of dark 5x5 modules, light 3x3 modules and a single central dark module. This pattern is actually used for correcting the distortion of the symbol [8]. The central coordinate of the alignment pattern will be identified to correct the distortion of the symbol.

### *2.1.3 Timing Pattern*

The timing patterns are arranged both in horizontal and vertical directions. These are actually having size similar to one module of the QR Code symbol. This pattern is actually used for identifying the central co-ordinate of each cell with black and white patterns arranged alternately.

### *2.1.4 Quiet Zone*

This region is actually free of all the markings. The margin space is necessary for reading the bar code accurately. This zone is mainly meant for keeping the QR Code symbol separated from the external area [9]. This area is usually 4 modules wide.

### *2.1.5 Data Area*

The data area consists of both data and error correction code words. According to the encoding rule, the data will be converted into 0's and 1's. These binary numbers will be then converted into black and white cells and will be arranged. Reed-Solomon error correction is also employed here [10].

## 2.2 Data Capacity

The capacity of QR Code is very large when compared to the data that can be stored in a linear barcode. The number of characters that can be encoded as QR Code varies according to the type of information that is to be encoded. The various information types and the volume that the QR Code can hold are explained in Table 1.

**Table 1. Information Types and Volume of Data**

| Information Type | Volume of Data |
|---|---|
| Numeric Characters | 7089 |
| Alphabets, Symbols | 4296 |
| Binary(8 bit) | 2953 |
| Kanji Characters | 1817 |

## 2.3 Data Compression

In computer science and information theory, data compression, source coding,[1] or bit-rate reduction involves encoding information using fewer bits than the original representation. Compression can be either lossy or lossless. Lossless compression reduces bits by identifying and eliminating statistical redundancy. No information is lost in lossless compression. Lossy compression reduces bits by identifying marginally important information and removing it.

Compression is useful because it helps reduce the consumption of resources such as data space or transmission capacity. Because compressed data must be decompressed to be used, this extra processing imposes computational or other costs through decompression. The design of data compression schemes involve trade-offs among various factors, including the degree of compression, the amount of distortion introduced and the computational resources required to compress and uncompress the data [11].

Lossless data compression algorithms usually exploit statistical redundancy to represent data more concisely without losing information. Lossless compression is possible because most real-world data has statistical redundancy. The Lempel–Ziv (LZ) compression methods are among the most popular algorithms for lossless storage. DEFLATE is a variation on LZ which is optimized for decompression speed and compression ratio, but compression can be slow.

# 3. EXPERIMENTAL RESULTS

The high capacity QR Codes which this paper focuses on consists of mainly two steps for encoding. First step compresses the data to be encoded. Second step encodes the compressed data into a QR Code.

## 3.1 Compression Technique

The compression technique followed here allows the user to compress the data so that a large quantity of data can be stored in a small sized bar code. More than 4 kb of data can be stored in this small barcode [11]. The compression which is employed here actually uses two steps: converting the text data into binary form and then generating the hash map data from this binary data.

## 3.2 QR Encoding

The normal encoding of data is done through various steps such as [1]:

*1)* Analyse the data to be encoded. Convert the data to symbol characters. Find out the error correction and detection level.
*2)* Encode the data.
*3)* Error Correction Coding
*4)* Add reminder bits and data masking patterns.
*5)* Generate the format information and version information [12].

The entire process can be made clear with the help of the simple flowchart given below.

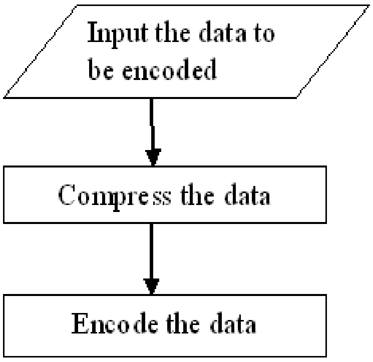


**Fig. 2 Steps for generating High-Capacity QR Code**

## 3.3 QR Decoding

Normally, QR decoding is done with the help of camera equipped mobile phones. Decoding process is just the reverse of the encoding procedure applied. We need to identify the quiet zone in order to decode the correct data. Alignment patterns help the decoding procedure by correcting the distortion of the symbol.

Kaywa reader is the most commonly used software to decode the original text. Image processing with J2ME is found to be more powerful. J2ME is designed to work on low-end devices in terms of processing power and memory capabilities. The various pre processing steps include gray scaling of the captured colour image, histogram stretching of the image, local adaptive thresholding, noise filtering, cropping, rotation correction and tilt correction. After pre processing, the finder patterns are detected and then the original data is decoded [5]. Another technique for decoding involves "edge to similar edge" estimation method is employed to check whether the detected bar-space pattern is correct [12].

## 3.4 Steps for Compression and Encoding

Various steps are to be followed to generate a 2D barcode for a large amount of data.

1) Save the data to be encoded as a text document.
2) Convert the entire data in this text document into a series of 0's and 1's.
3) Generate a hash map data by converting this series of 0's and 1's into hexadecimal numbers, like 0000 for A, 0001 for B and so on.
4) Generate the 2D barcode for this hash map data using the normal encoding procedures.

Save the alphanumeric data which we want to encode as a text file, say "abc.txt". For comparison purposes, include the text data which is more than 4 kb. All the other steps make use of this text data to generate a 2 dimensional bar code. The next major step is the conversion of this text data into a series of 0's and 1's. First of all, find out the length of the entire text data which you want to encode. After converting each letter into its corresponding binary form, the length needs to be updated as to make the procedure easier.

The technique by which we convert the text data into binary form is quite simple to understand. Every letter will be converted into an 8 bit sequence of 0's and 1's. All the letters from 'a' to 'o' will have the first four binary numbers same as '0110'.The letters from 'p' to 'z' will have the first four binary numbers as '0111'. The remaining four numbers depends upon the position of the letter. That means, if the letter is 'a', the next four numbers will be '0001'. If the text data contains a space, then the binary equivalent is 0010 0000.

The next step actually focuses on the compression of the binary data that we have generated from the actual text data. It will group each of the four numbers as one alphabet. This is actually the generation of the hash map data.

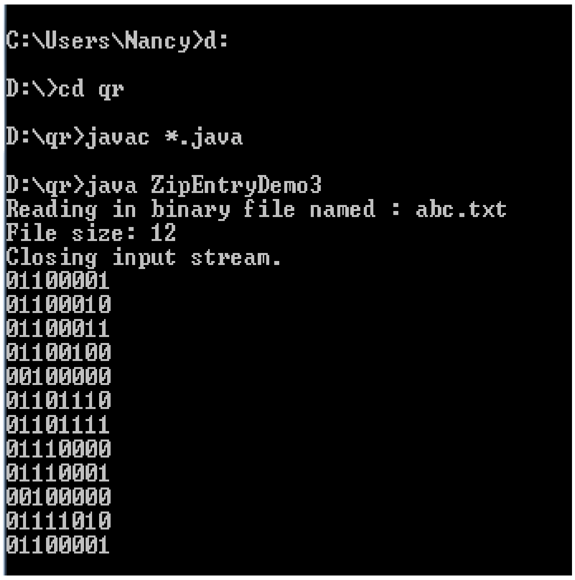
The procedure for generating hash map data is as follows. Every 4 bit data is converted into one of the characters from 'A' to 'P'. Whenever a '0000' is encountered, the hash map data will be 'A', '0001' will be mapped as 'B' and so on.

After generating the hash map data, the actual encoding process starts. It will encode these data into a barcode which is square in nature and which will be like black squares in white back ground.

Colour QR Codes can be generated from the hash map data. RGB colour coding technique can be used. Every letter from 'A' to 'P' will be assigned different colours. Whenever such a character is encountered, the corresponding colour will be placed on the QR Code. Finder patterns and timing patterns are also arranged in the QR Code at appropriate places. After filling the QR Code with the data, finder pattern and timing pattern, rest of the QR Code is filled with filler patterns.

Consider that the data we want to encode is "abcd nopq za". First convert it into a series of 0's and 1's.

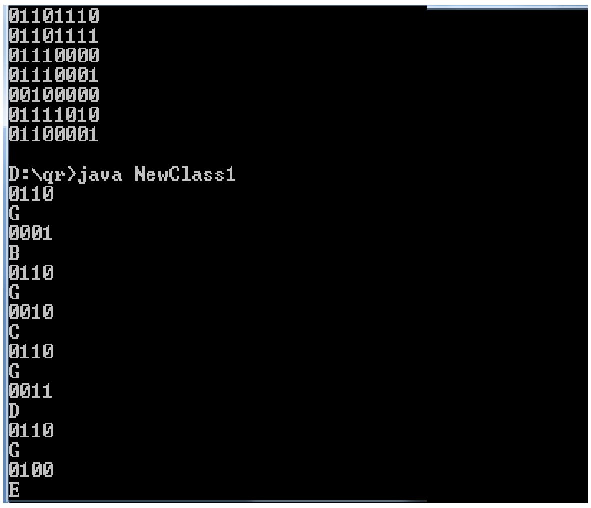
Figure 2-4 well explains the various steps which are used to generate a 2D bar code. Figure 2 shows how the original text data "abcd nopq za" is converted into its binary form. Figure 3 clearly shows how this binary data is converted into its hash map representation. The complete hash map representation is not shown here. Figure 4 shows how the hash map data can be encoded into a 2D barcode.



```
C:\Users\Nancy>d:

D:\>cd qr

D:\qr>javac *.java

D:\qr>java ZipEntryDemo3
Reading in binary file named : abc.txt
File size: 12
Closing input stream.
01100001
01100010
01100011
01100100
00100000
01101110
01101111
01110000
01110001
00100000
01111010
01100001
```

**Fig. 3 Binary data**

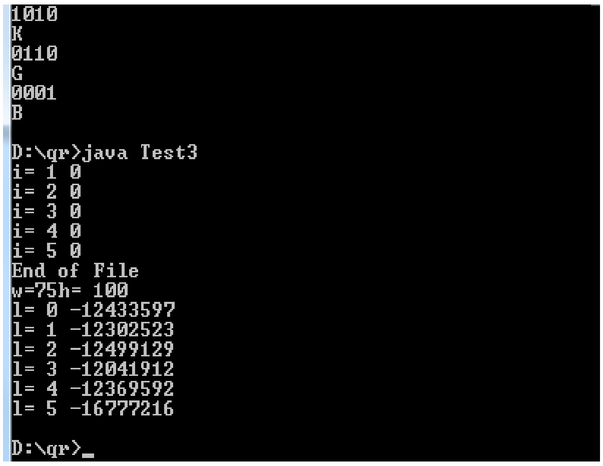
Generate the hash map data for the above binary data. The hash map data will have characters only from 'A' to 'P' because the binary data will be only from '0000' to '1111'.



```
01101110
01101111
01110000
01110001
00100000
01111010
01100001

D:\qr>java NewClass1
0110
G
0001
B
0110
G
0010
C
0110
G
0011
D
0110
G
0100
E
```

**Fig. 4 Hash map data**

This takes place for the entire series of 0’s and 1’s that we have created. The last step creates the actual encoded barcode image from the hash map data.



```
1010
K
0110
G
0001
B

D:\qr>java Test3
i= 1 0
i= 2 0
i= 3 0
i= 4 0
i= 5 0
End of File
w=75h= 100
l= 0 -12433597
l= 1 -12302523
l= 2 -12499129
l= 3 -12041912
l= 4 -12369592
l= 5 -16777216

D:\qr>_
```

**Fig. 5 Encoding**

If a colour QR Code is needed, then the last step will change accordingly. Here, figure 4 is not for a colour QR Code. It just creates a normal black-and-white QR Code. Colour QR Code is actually an expanded version of the normal QR Codes. The last step focuses on the QR Encoding of images. If the compression technique used is novel, the data capacity can also be increased.

QR Code 2005 employs Reed-Solomon error control coding to detect and correct errors. Reed Solomon codes can be used to correct errors in many systems including storage devices, satellite communication etc. Errors occur during data storage or transmission for a number of reasons. Reed Solomon Codes are a subset of BCH Codes and are linear block codes. A series of error correction code words is generated, which are added to the data code word sequence in order to enable the symbol to withstand damage without loss of data. The error correction code words can correct two types of erroneous code words, erasures (erroneous code words at known locations) and errors (erroneous code words at unknown locations). An erasure is an un-scanned or un-decodable symbol character. An error is a mis-decoded symbol character.

## 4. RESULTS

Using this new technique of compressing the data before generating QR Code, efficiency can be improved. This technique suggests simple ways to accomplish this task. All the text data will be converted into a series of 0’s and 1’s. Then, the hash map data will be generated and then only the original QR Code will be generated.

Normal QR Codes can compress only up to 4 kb of data. Using the techniques followed here, the data capacity can be increased drastically. As compared to the normal QR Codes, the data capacity of the QR Code after following technique was found to be more than 4 kb. Efficient data compression techniques can be used to store more than 4 Mb of data inside a QR Code. A variety of data compression techniques can be used to obtain more data storage capacity. Comparing with the existing technologies used to generate bar codes, QR Codes were found to be of great advantage to the manufacturer because of its great data storage capacity, reading speed and accuracy. The data capacity was further improved by combining the most distinguishing features of compression and bar code generation. Using this novel technique of data compression followed by data encoding, the data storage capacity of QR Codes were increased drastically.

## 5. CONCLUSIONS

QR Codes are being increasingly noticed on products and in advertisements, and many are still unaware of what the black modules on a white background in a square box stand for. QR code has high data density. The data density was increased even more using the technique of compression of data followed by encoding the data. They can be used to link to any URL [2]. Disadvantage regarding QR code is that users must be equipped with a camera phone and the correct reader software that can scan the image of the QR Code. Currently only Smartphone’s are technically equipped to do this. Many users that have mobile phones that have cameras are unable to get QR reading software for their phones. Future enhancements focus on QR Encoding of images which is more than 4 kb of size. Secure QR Coding can also be implemented using encryption techniques. Also, more advanced data compression techniques can be used to add more to the data capacity of the normal QR Codes.

## 6. REFERENCES

[1] Xiaofei Feng, Herong Zheng, “Design and Realization of 2D Color Barcode with High Compression Ratio” 201O International Conference On Computer Design And Appliations (ICCDA 2010), 978-1-4244-7164-51$26.00 © 2010 IEEE, 978-1-4244-7164-51$26.00 © 2010 IEEE, Volume 1

[2] Antonio Grillo, Alessandro Lentini, Marco Querini and Giuseppe F. Italiano, “High Capacity Colored Two Dimensional Codes”, Proceedings of the International Multiconference on Computer Science and Information Technology pp. 709–716, ISBN 978-83-60810-27-9, ISSN 1896-7094

[3] Peter Kieseberg, Manuel Leithner, Martin Mulazzani, Lindsay Munroe, Sebastian Schrittwieser, Mayank Sinha, Edgar WeipplT. J., “QR Code Security”

[4] Chun-lei XIA, “Examination Certificate Based on Two-Dimensional Bar Code Technology”, 2008 International Symposium on Computer Science and Computational Technology, 978-0-7695-3498-5/08 $25.00 © 2008 IEEE DOI 10.1109/ISCSCT.2008.102

[5] Tasos Falas, Hossein Kashani, “Two-Dimensional Barcode Decoding with Camera-Equipped Mobile Phones”, Proceedings of the Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops(PerComW'07) 0-7695-2788-4/07 $20.00 © 2007

[6] William Claycomb, Dongwan Shin, “Using A Two Dimensional Colorized Barcode Solution for Authentication in Pervasive Computing”, 1-4244-0237-9/06/$20.00 ©2006 IEEE

[7] Guenther Starnberger, Lorenz Froihofer and Karl M. Goeschka, “QR-TAN: Secure Mobile Transaction Authentication”, 2009 International Conference on Availability, Reliability and Security, 978-0-7695-3564-7/09 $25.00 © 2009 IEEE DOI 10.1109/ARES.2009.96

[8] ISO/IEC 18004:2000 Information Technology - Automatic Identification and Data Capture Techniques – Barcode Symbology- QR Code (MOD), June 2000.

[9] Sarah Lyons and Frank R. Kschischang, “Two-Dimensional Barcodes for Mobile Phones”, 25th Biennial Symposium on Communications, 978-1-4244-5711-3/10/$26.00 ©2010

[10] R. Bose and D. Ray-Chaudhuri. On a class of errorcorrecting binary group codes*. Information and control, 3(1):68{79, 1960.

[11] David L. Donoho, Martin Vetterli, Fellow, IEEE, R. A. DeVore, and Ingrid Daubechies, Senior Member, IEEE,” Data Compression and Harmonic Analysis”, IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 44, NO. 6, OCTOBER 1998, 0018–9448/98$10.00 ã 1998 IEEE

[12] Hee I1 Hahn and Joung Koo Joung, “Implementation of Algorithm to Decode Two-Dimensional Barcode PDF-417”, ICSP’O2 Proceedings, 0-7803-7488-6/02/$17.00 Q 2002 IEEE.