Journal of Theoretical and Applied Information Technology
15th August 2017. Vol.95. No.15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

COLOR QR CODE RECOGNITION UTILIZING NEURAL NETWORK AND FUZZY LOGIC TECHNIQUES

[1]BAKRI BADAWI, [2]TEH NORANIS MOHD ARIS
[3] NORWATI MUSTAPHA, [4] NORIDAYU MANSHOR
[1,2,3,4] Department of Computer Science, Faculty of Computer Science and Information Technology, 43400 UPM Serdang, Selangor, malaysia
E-mail: [1]bakri.info@gmail.com, [2]nuranis@upm.edu.my (corresponding author)
[3]norwati@upm.edu.my, [4]ayu@upm.edu.my

ABSTRACT

Quick Response (QR) code is popular type of two dimensional barcode. The key feature of QR code is larger storage capacity and high damage resistance compared to the traditional barcodes. Color QR code is the future as it provides much higher encoding capacity, but it also brings tremendous challenges to the decoding because of color interference and illumination. This research paper presents a method for QR code recognition using the Neural Network (NN) and fuzzy logic techniques. We created a framework for image decoding. First, the color QR code is converted to black and white then the QR code is recognized using neural network. Next, the original colors are returned to the QR code. The colors are enhanced using fuzzy logic and then, the enhanced color QR code is split into three barcodes which are red, green and blue. Finally, each QR code is converted to black and white and sent to ZXing library for decoding and obtained the original data. ZXing library has been utilized for decoding and recognition purposes and has produced satisfactory results. This research proof that by, utilizing NN and fuzzy logic techniques has produced better QR code success rates of five percent.

Keywords: *QR Code, Artificial Intelligence (AI), Neural Network, Fuzzy*

1. INTRODUCTION

The two dimensional barcode, Quick Response Code (QR code) is focused with the goal of high speed reading and encoding capacity compared to traditional barcodes. QR code has gained popularity over classical barcode because of several advantages like high capacity, reduced size, 360 degrees of reading etc. As compared to traditional barcodes, QR codes can hold 10 times more data in the same amount of space [1]. The capacity of the QR code can be increased by encoding different layers of codes by different colors. The information encoded in the colored QR code is directly proportional to log2N [2], where N is number of colors used. Therefore, the larger number of colors used, the larger encoded information can be stored. Figure 1 shows an example of a color QR code.

Adding more color to QR code brings tremendous challenges to the decoding because of color interference and illumination. Many researchers tried to solve this issue by adding color interface [5], color multiplexing [7] and other image preprocessing technique. We proposed a different approach which is decoding algorithm based on NN and fuzzy logic to improve the accuracy in decoding the QR code. The detailed implementation is presented in this paper. The reason to use color QR code is because it can hold bigger data.

This paper is divided into seven sections. Section two explains about the QR code structure, Section three describes the current QR code and its limitation. Section four is the selected research works that are related to our work. Section five is the implementation of our proposed work. Section six is the result and discussions from our findings. Finally, section seven is the conclusion and future works.

Journal of Theoretical and Applied Information Technology
15th August 2017. Vol.95. No.15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 1: Color QR Code

2. QR-CODE STRUCTURE

QR code has various area patters [3] [4] as illustrated in Figure 2. Each of these has different functions as follows:

Finder patterns: are located in three of the four corners which allow 360 degree high-speed reading of the code.

Format information: It contains the error correction rate and mask pattern of QR code. The format information is read first when the code is decoded.

Data and error-correction code words: is applied to restore the data when a part of QR code is missing. The restoration rate varies on four different error correcting levels. For example, if a damaged area is up to 15% of the entire code size, its data can be restored by level M error correction.

Version Number: Span from version 1 (21 x 21 modules) to version 40 (177 x 177 modules).

Alignment pattern: they can appear in more complex QR codes and they will be located in the lower right hand corner. This pattern allows the QR reader to correct for distortion when the code is bent or curved. The number of alignment patterns used depends upon how much information is being encoded and appears on version 2 and higher.

Timing pattern: helps to detect the position of each cell in the QR code.

Figure 2: QR Code Structure

3. CURRENT QR CODE AND ITS LIMITATION

The symbol versions of QR code range from version 1 to version 40. Each version has a different module configuration. Module configuration refers to the number of modules contained in a symbol, commencing with version 1 (21 × 21 modules) up to version 40 (177 × 177 modules). Each higher version number comprises of four additional modules per side as shown in Figure 3.

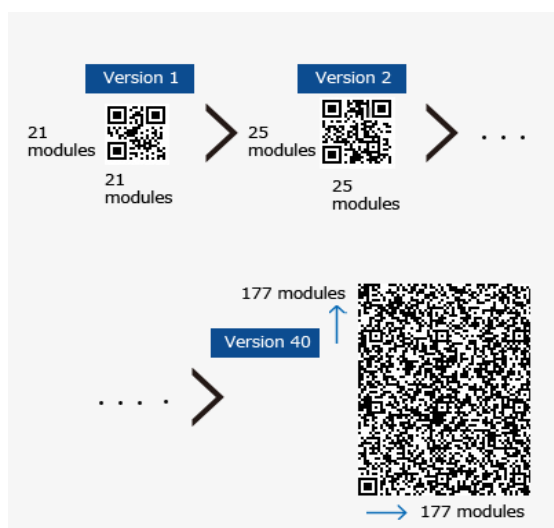


Figure 3: QR Code Versions

Each QR code symbol version has maximum data capacity according to the amount of data, character

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

type and error correction level. In other words, as the amount of data increases, more modules are required, resulting in larger QR code symbols. The maximum data capacity for each QR Code version is shown in Table 1.

Table 1: QR Code Data Size

Version	Modules	Error Correction	Data bits	Numeric	Alphanumeric	Binary
1	21x21	L	152	41	25	17
		M	128	34	20	14
		Q	104	27	16	11
		H	72	17	10	7
2	25x25	L	272	77	47	32
		M	224	63	38	26
		Q	176	48	29	20
		H	128	34	20	14
12	65x65	L	2.960	883	535	367
		M	2.320	691	419	287
		Q	1.648	489	296	203
		H	1.264	374	227	155
13	69x69	L	3,424	1,022	619	425
		M	2.672	796	483	331
		Q	1.952	580	352	241
		H	1.440	427	259	177
39	173x173	L	22.496	6.743	4.087	2.809
		M	17.728	5.313	3.220	2.213
		Q	12.656	3.791	2.298	1.579
		H	9.776	2,927	1.774	1.219
40	177x177	L	23.648	7.089	4.296	2.953
		M	18.672	5.596	3.391	2,331
		Q	13.328	3.993	2,420	1.663
		H	10.208	3,057	1.852	1.273

In Table 1, the maximum capacity for QR code is 23.648 bits with a low error correction [7][11]. Many researchers have tried to increase this limitation which will be explained in the next section.

4. RELATED RESEARCH WORKS

We will describe in detail four systems that are used as benchmarks in our research work:

4.1 [Rayana Boubezari, Hoa Le Minh, Zabih Ghassemlooy, and Ahmed Bouridane, 2016].

They proposed a camera-based receiver for a short range mobile-to-mobile communications consisting of the sender and receiver.
The sender converts any type of digital data into binary stream, in zeros and ones.
The binary stream is converted into image 0 that represents black pixel and 1 that represent white pixel as shown in Figure 4.

Figure 4: Rayana Boubezari, Hoa Le Minh, Zabih Ghassemlooy, and Ahmed Bouridane Barcode [4]

The receiver captures data from the screen of a transmitting smartphone and use the Speeded-Up Robust Features algorithm to detect it. Then, the receiver reconstructs the cell and obtains the original data.

From our review, this system provides good data capacity with acceptable success rate. However, the data capacity could be enhanced by adding color in the barcode.

4.2 [Sin Rong Toh, Weihan Goh, and Chai Kiat Yeo, 2016]

These researchers have suggested a new approach of exchanging data as images via mobile devices.
First, from the sender, the text file is split into red, green and blue color in almost equal chunks, each about a third of the file size (e.g. a text file of 900 bytes will have each chunk of 300 bytes). These chunks are then converted to QR code. The maximum size of each QR code is 500 bytes. Three QR codes in red QR, green QR and blue QR are multiplexed to represent one image. When they

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

overlap with each other, they produce secondary colors. Thus one image consists of a mixture of all primary colors and secondary colors. A total of eight colors are used including black and white.
As a start, from the receiver the image is extracted and split to Red, Green and Blue (RGB) channels using gray-scale conversion. The decoding results are in three black and white QR codes for one image which correspond to their original color QR code before the color inversion.

From our findings, this system provides a good data capacity. However, the success rate could be improved by enhancing the color in the QR code detection.

4.3 [*Thilo Fath, Falk Schubert,and Harald Haas, 2014*]

Authors have introduced data transmission within an aircraft cabin. The data is transferred by stream of QR code. The QR code is captured by a built-in camera in a passenger's mobile camera and is decoded to reconstruct the transmitted data.

The process is as follows: first, from the sender part, the file, for instance text documents or images, is compressed to reduce the amount of data which is to be transmitted. Second, the compressed file is Forward Error Correction (FEC) encoded to compensate for potential transmission errors. In the third step, the FEC encoded data is segmented into several packets. Each of these packets is visually encoded by a visual code resulting in a sequence of several visual codes. Fourth, this visual code sequence is displayed in a continuous loop on the In-Flight Entertainment (IFE) screen like a common video film.
From the receiver, the QR codes are captured using a mobile camera and the captured codes are visually decoded. The visually decoded data packets are reassembled in the correct order by means of additionally encoded meta-information providing the packet number. Then, the FEC encoded data is decoded. Due to the induced FEC, errors which arise during the transmission and capturing process can be corrected. Finally, the FEC decoded data is decrypted and uncompressed and the reconstructed file is stored on the user device. The sending and receiving process is shown in Figure 5.
From our review, this system provides good data capacity. However, the success rate can be enhanced by adding color in the QR code detection. In addition, the data size can also be increased.

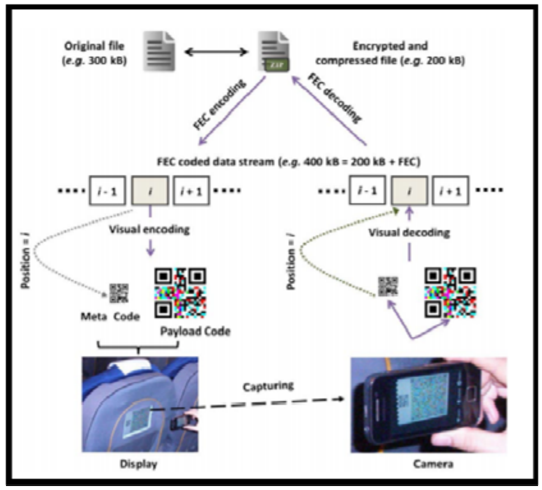


Figure 5: Thilo Fath, Falk Schubert,and Harald Haas, 2014 System [5]

4.4 [Tian Hao, Ruogu Zhou, Guoliang Xing 2012]

These researchers have proposed colored barcode to increase the capacity of QR code. First, for data encoding, the bit stream of data are obtained, then convert each of the bit into color for example red, green, blue and white colors which represent 00, 01, 10 and 11 respectively On a typical 4-inch phone screen with 800×480 resolution, a single color barcode with 6-pixel block size contains 18.8K bits. For decoding the receiver, the colors are identified and return it to its bit reference. Figure 6 show an example of their barcode.

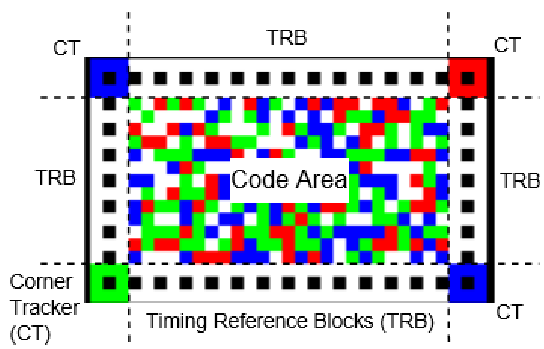


Figure 6: Thilo Fath, Falk Schubert,and Harald Haas, 2014[5]

We discover that this system provides a good data capacity. However, the success rate can be enhanced. In addition, the data capacity is only 2 bit per color while other researchers have enhanced the capacity to 3 bits per color.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4.4 Comparison Between Existing System

We compare the systems in four aspects: the maximum data that can hold each barcode, the camera use, success rate and the number of colors as shown in Table 2. The success rate is the median of the success reading for the proposed QR-Code (Total success reading/ Total captured QR-Code).

Table 2:[Thilo Fath, Falk Schubert,and Harald Haas, 2014 System]

	Max Data size (bit)	Camera	Success Rate	Color
(Sa Rayana Boubezari, Hoa Le Minh, Zabih Ghassemlooy, and Ahmed Bouridane,2016)	3750 bit	8 M 18 M	60% 80%	B/W
(Thilo Fath, Falk Schubert,and Harald Haas, 2014)	9264 bit 2240 bit	5M	50%	8 Color B/W
(Sin Rong Toh, Weihan Goh, and Chai Kiat Yeo ,2016)	3 times > original QR Code	5M	40%	8 Color
(Tian Hao, Ruogu Zhou, Guoliang Xing ,2012)	18800 bit	8 M	64%	5 Color

Summarization from Table 2, shows that the maximum data capacity that can be hold per barcode = 3*23684= 71052 bit with success rate of 40% only. The results produced in Table 2 based on previous research, is used as a bench mark to compare the results with our work.

5. IMPLEMENTATION

5.1 Data Encoding

Many researchers proposed to enhance the QR code by merging three QR codes (red, green, and blue) in one colored QR code [1,5]. This is done by first, splitting the file into almost equal 3 chunks, each about a third of the file size (e.g. a text file of 900 bytes will have each chunk of 300 bytes). These chunks are then converted to QR code. Three QR codes (Red QR, Green QR and Blue QR) are multiplexed to represent one image. When they overlap with each other, they produce secondary colors. Thus, one image consists of a mixture of all primary colors and secondary colors. A total of eight colors are used including black and white. This process is also adopted in our research work. Figure 7 illustrates the encoding process.

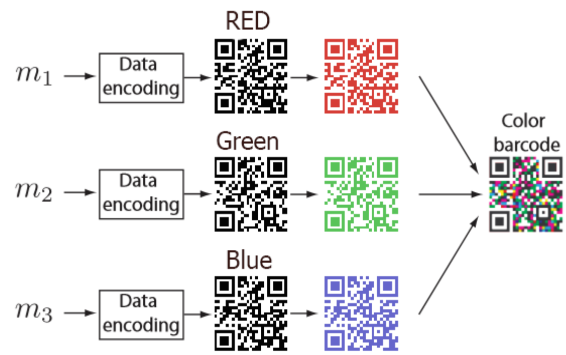


Figure 7: Data Encoding

5.1.1 Data Encoding Pseudo Code

The data encoding pseudo code is as follows:

```
encodedtext = Readtext();
chunklength = encodedtext.length / 3;
chunk1 = encodedtext.subString(0, chunklength);
chunk2= encodedtext.subString(chunklength,
chunklength);
chunk3= encodedtext.subString(chunklength*2);
redQr = ZXingLibrary.Generate(chunk1);
greenQr = ZXingLibrary.Generate(chunk2);
blueQr = ZXingLibrary.Generate(chunk3);
 for (i = 0; i < redQr.Width; i++)
    {
       for (int j = 0; j < redQr.Height; j++)
         {
              redColor = redQr.GetPixel(i, j);
              greenColor = greenQr.GetPixel(i, j);
              blueColor = blueQr.GetPixel(i, j);
              fullcolor =
                    Color.Fromrgb(rredColor,
              greenColor, blueColor);
             colorQr.SetPixel(i, j, fullcolor);
          }
       }
 Save(colorQr);
```

5.2 Information Decoding

The decoding process is shown in Figure 8. First we convert the image to black and white, then detecting the QR-code using text feature and neural network, retrieve the color to the detected barcode, enhance the color using fuzzy logic, split the barcode into 3 QR-code (red, green, blue), decode the QR-codes and get back the original file.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

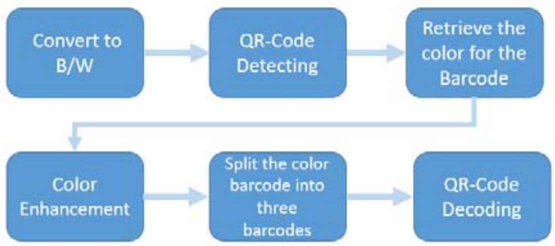


Figure 6: Decoding Process

5.2.1 Convert image into black and white

We use the lightness method to get the black and white image. The average of the most prominent and least prominent colors apply the following formula:

(max(R, G, B) + min(R, G, B)) / 2.

5.2.2 Detecting the QR code using text feature and neural network

Authors [2] have proposed a technique for recognition of Black and white QR code which based on neural network classifier detection. However we are using for color QR code. The QR code process is illustrated in Figure 9.

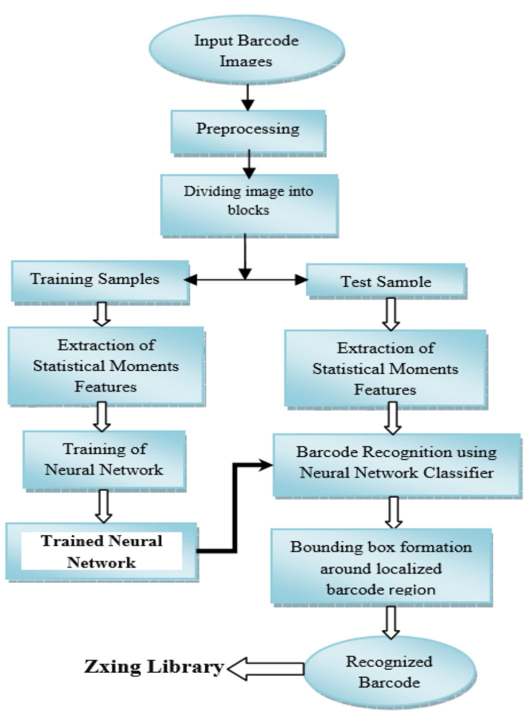


Figure 9: [Priyanka Gaur1, Shamik Tiwari, 2014 Detecting Technique]

Their work consists of three stages as follows:
Stage 1: Pre-processing all the input images used for feature extraction, training and testing of neural network should be of same size.
Stage 2: Dividing image into blocks. The whole process relies on both pixel level and block level processing. If required, all four edges of the input image will be trimmed so that the image size is divisible by the block size.
Stage 3: Feature extraction after converting input image into same size blocks, statistic features of each block is extracted using statistical functions. After obtaining output value for the whole image blocks from the trained neural network with statistical features, the image is converted into binary image.

5.2.3 Retrieve the color for the barcode

After we select the QR code location we select the QR code from the original colored image.

5.2.4 Color enhancement

[Cui Liu, Lianming Wang ,2015] proposed color fuzzy recognition method. The overall method shown in Figure 10.

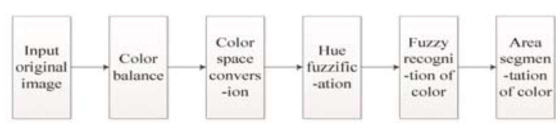


Figure 10 [Cui Liu, Lianming Wang ,2015] Color Enhancement Overall Method

Fuzzy processing includes two main steps: first, to reform the fuzzy membership and second, defuzzification. For the fuzzy algorithm, we proposed three fuzzy sets. First we calculate the pixel degree according to its color and then judge the type of pixel (red, green, or blue) and convert the color of the pixel point to the standard color of the color it belongs. Therefore, the color of the image can be divided into three fuzzy types.
Our fuzzy set consists of:

- If red component, for red color pixel value between 0 and 127, the color is black.
- If green component, for green color pixel value between 0 and 127, the color is black.
- If blue component, for blue color pixel value between 0 and 127, the color is black.

Journal of Theoretical and Applied Information Technology
15th August 2017. Vol.95. No.15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The membership function is defined as:
Ared = {*xred* | *xred* > 127}
Agreen = {*xgreen* | *xgreen*> 127}
Ablue = {*x*blue | *x*blue> 127}

Figure 11 illustrate the membership function:

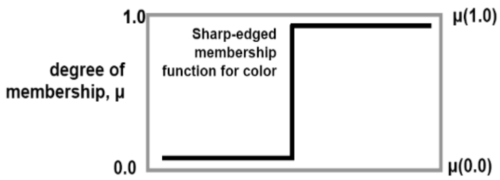


Figure 11: Membership Function

The captured image sample is shown in Figure 12.

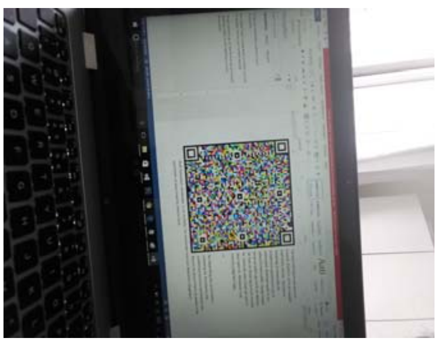


Figure 12: Captured image

The result after fuzzification is shown in Figures 13, 14, 15.

Figure 13: Enhanced Red QR-Code

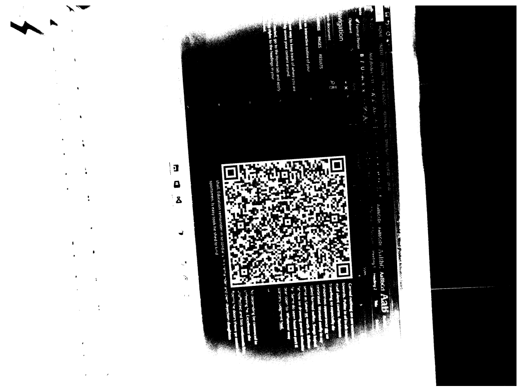


Figure 14: Enhanced Green QR-Code

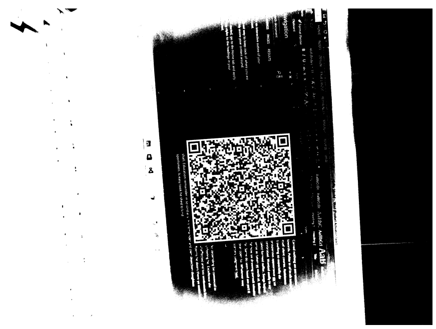


Figure 15: Enhanced Blue QR-Code

5.2.5 Split the color barcode into three barcodes and decode the QR code

First, we split the enhance color QR code into three main channels: red, green, and blue. Second, convert each channel to black and white image. So we produced three black and white QR codes. Third we send these QR codes to Zxing library for decoding. Forth, we obtain the original data, consisting of three separated result from Zxing. Finally, by combining the three results we get our original file.

5.2.6 Pseudo code for data decoding

The pseudo code for data decoding is as follows:

```
colorQr = ReadImage();
for (i = 0; i < colorQr.Width; i++){
        for (j = 0; j < colorQr.Height; j++){

colorPixcel = colorQr.GetPixel(i, j);
  if (colorPixcel.Red >= 128)
    {
        red = Color.FromArgb(0, 0, 0);
    }else {
```

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

```
        red = Color.FromArgb(255, 255, 255);
    }
  if (colorPixcel.Green >= 128)
    {
        green = Color.FromArgb(0, 0, 0);
    }else {
        green = Color.FromArgb(255, 255, 255);
     }
   if (colorPixcel.Blue >= 128)
      {
        blue = Color.FromArgb(0, 0, 0);
        }else{
         blue = Color.FromArgb(255, 255, 255);
       }
   redQr.SetPixel(i, j, rtempcolor);
   greenQr.SetPixel(i, j, gtempcolor);
   blueQr.SetPixel(i, j, btempcolor);
                 }
             }
chunk1 = ZXingLibrary.Decode(redQr);
chunk2 = ZXingLibrary.Decode(greenQr);
chunk3 = ZXingLibrary.Decode(blueQr);
decodedtext = chunk1 + chunk2 + chunk3;
```

6. RESULTS AND DISCUSSION

We test our system for ten colored QR code. We capture the QR code using Samsung a9, Lumia 950, Iphone6, and Lenovo Pahp. We have tested the success rate for each composite element of the color QR code (red, green, blue), then we calculate the total. Our result reached to 100% for Samsung front camera, 96.7% for Samsung rear camera and Iphone6, 90% for Lenovo, and 80% for Lumia. Table 3 illustrates our result.

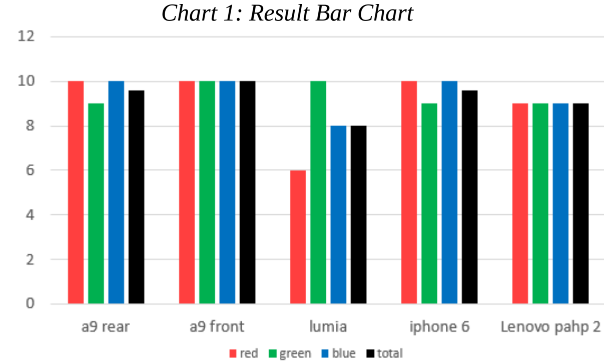
We notice that the success rate has reached 100% with 8 mega pixel camera while for 20 mega pixel camera obtained only 80% success rate. The error in decoding our QR code is due to various images processing algorithm used in different device.

Table 3: Results

	Camera	Red Success Rate	Blue Success Rate	Green Success Rate	Success Rate
Samsung a9 pro rear camera	16 MP	100%	90%	100%	96.67%
Samsung a9 pro front camera rear camera	8 MP	100%	100%	100%	100%
Lumia 950 rear camera	20 MP	60%	100%	80%	80%
I phone 6 rear camera	8 MP	100%	90%	100%	96.67%
Lenovo rear camera	13 MP	90%	90%	90%	90%

Chart 1 shows the result representation in a form of a bar chart.

Chart 1: Result Bar Chart



Using the same camera specification, we can compare our result with the related research works in section four. Table 4 shows the comparison between our work and previous works results. From Table 4, it is proved that our work produced better success rate of around twenty percent. Based on previous works, the best success rate was ninety five percent. Utilizing the NN and fuzzy logic techniques have produced a hundred percent success rate for the red, blue and green colors in some cameras.

Table 4: Results Comparison

	Max Data size (bit)	Camera	Success Rate	Our work Success rate using the same camera specification
(Sa Rayana Boubezari, Hoa Le Minh, Zabih Ghassemlooy, and Ahmed Bouridane,2016)	3750 bit	8 M 18 M	60% 80%	100% 96.67%
(Thilo Fath, Falk Schubert,and Harald Haas, 2014)	9264 bit 2240 bit	5M	95%	100%
(Sin Rong Toh, Weihan Goh, and Chai Kiat Yeo ,2016)	3 times > original QR Code	5M	90%	96.67%
(Tian Hao, Ruogu Zhou, Guoliang Xing ,2012)	18800	8 M	90%	96.67%

Although the success rate has been improved in our work, however there are still some limitations. In this work, we did not consider about the scanning speed and the distance between the screen and the mobile camera. Currently, the distance between the mobile camera and computer screen is twenty to fifty centimeters. In addition, the assumption made from this study is that the QR code is captured from computer screens because the color format in screens is red, blue and green (RGB) while printers is cyan, magenta, yellow and

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

black (CMYK) [15]. We will consider capturing the QR code images from printers.

7. CONCLUSION AND FUTURE WORK

In this paper we have improved the color QR code recognition utilizing neural network and fuzzy logic techniques. The experiment proved that utilizing these techniques have produced better success rate. Our proposed algorithm supports the detection of color QR code images for indoor and outdoor. The fuzzy logic algorithm is used for color enhancement which provides height detection accuracy. The neural network is applied for the detection part.

For future works, the system will be tested for printed QR code on paper, because currently the QR code is read from the computer screen. In addition, the scanning speed and the distance between the screen and the mobile camera will also be improved. It is expected that the developed application could be used for the authentication of consumer products and other business purposes.

REERENCES

[1] Nivedan Bhardwaj1, Ritesh Kumar2, Rupali Verma1, Alka Jindal1 and Amol P. Bhondekar2," Decoding Algorithm for color QR code: A Mobile Scanner Application", *International Conference on Recent Trends in Information Technology* (ICRTIT) (2016) 1-6.

[2] Priyanka Gaur1, Shamik Tiwari2 "2D QR Barcode Recognition Using Texture Features and Neural Network" *International Journal of Research in Advent Technology*, Vol.2, No.5, May 2014.

[3] Cui Liu, Lianming Wang, "Fuzzy Color Recognition and Segmentation of Robot Vision Scene", (2015) 8th *International Congress on Image and Signal Processing*.

[4] Sa Rayana Boubezari, Hoa Le Minh, Zabih Ghassemlooy, and Ahmed Bouridane, "Smartphone Camera Based Visible Light Communication" JOURNAL OF LIGHTWAVE TECHNOLOGY, VOL. 34, NO. 17, SEPTEMBER 1, (2016) 4120-4126.

[5] Thilo Fath, Falk Schubert,and Harald Haas, "Wireless data transmission using visual codes." *Photonics Research Volume* 2 Issue 5 Page 150 (2014).

[6] Samuel David Perli, Nabeel Ahmed, and Dina Katabi "PixNet: Interference-Free Wireless Links Using LCD-Camera pairs" *MobiCom* 10 (2010) 137-148.

[7] Sin Rong Toh, Weihan Goh, and Chai Kiat Yeo "Data Exchange via Multiplexed Color QR Codes on Mobile Devices" *MobiCom* Wireless Telecommunications Symposium (WTS), (2016).

[8] Tian Hao, Ruogu Zhou, Guoliang Xing "COBRA: Color Barcode Streaming for Smartphone Systems" *MobiSys* '12 (2012)

[9] R. Boubezari, H. Le Minh , Z. Ghassemlooy and A. Bouridane "Novel Detection Technique for Smartphone to Smartphone Visible Light Communications." 10th International Symposium on Communication Systems (2016).

[10] Bingsheng Zhang, Kui Ren, Senior Member, IEEE, Guoliang Xing, Senior Member, IEEE, Xinwen Fu, Senior Member, IEEE, and Cong Wang, Member, IEEE "SBVLC: Secure Barcode-Based Visible Light Communication for Smartphones" IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 15, NO. 2, FEBRUARY(2016).

[11] P.S. André and R.A.S. Ferreira"Colour multiplexing of quick-response (QR) codes"ELECTRONICS LETTERS 20th November (2014)Vol. 50 No. 24

[12] *Max E.Vizcarra Melgar, and Luz M. Santander "Channel Capacity Analysis of 2D Barcodes: QR Code and CQR Code-5". IEEE COLCOM (2016)*

[13] *Keng T. Tan, Douglas Chai, Hiroko Kato, and Siong Khai Ong " Designing a color Barcode for Mobile applications" ECU PUBLICATIONS (2012)*

[14]AshwinAshok,ShubhamJain,MarcoGruteser,NarayanMandayam,WenjiaYuan, Kristin Dana, "Capacity of screen–camera communication sunder perspective distortions" Pervasiveand Mobile Computing 16 (2015)

[15]https://en.m.wikipedia.org/wiki/CMYK_color_model