Journal of Advanced Research 66 (2024) 15–30

Contents lists available at ScienceDirect

Journal of Advanced Research

journal homepage: www.elsevier.com/locate/jare

Original Manuscript

4SQR-Code: A 4-state QR code generation model for increasing data storing capacity in the Digital Twin framework

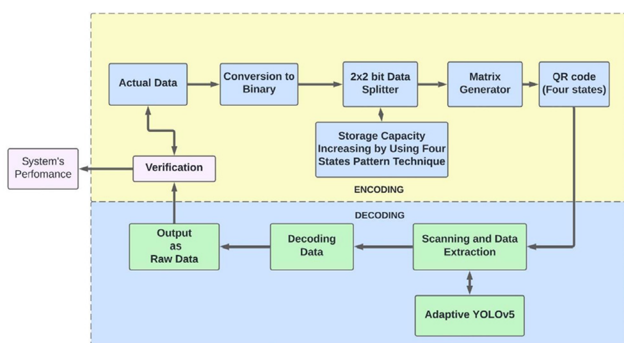
Ababil Islam Udoy [a], Muhammad Aminur Rahaman [a,*], Md. Jahidul Islam [a], Anichur Rahman [b,*], Zulfiqar Ali [c], Ghulam Muhammad [d,*]

[a] Department of CSE, Green University of Bangladesh, Purbachal American City, Kanchan, Rupganj, Narayanganj-1461, Dhaka, Bangladesh
[b] Department of CSE, National Institute of Textile Engineering and Research (NITER), Constituent Institute of the University of Dhaka, Savar, Dhaka 1350, Bangladesh
[c] School of Computer Science and Electronic Engineering, University of Essex, United Kingdom
[d] Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia

HIGHLIGHTS

- A novel 4SQR code is proposed based on traditional QR codes.
- We implement the 4SQR to generate and detect the readability of the system.
- The proposed 4SQR can increase data storing capacity with security.
- It is tested in a digital twin framework using randomly generated samples.

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:
Received 27 May 2023
Revised 13 September 2023
Accepted 6 October 2023
Available online 18 October 2023

Keywords:
Quick response (QR) code
4-State QR (4SQR) Code
Digital Ttwin
Encoding
Decoding

ABSTRACT

***Introduction**:* The usage of Quick Response (QR) Codes has become widely popular in recent years, primarily for immense electronic transactions and industry uses. The structural flexibility of QR Code architecture opens many more possibilities for researchers in the domain of the Industrial Internet of Things (IIoT). However, the limited storage capacity of the traditional QR Codes still fails to stretch the data capacity limits. The researchers of this domain have already introduced different kinds of techniques, including data hiding, multiplexing, data compression, color QR Codes, and so on. However, the research on increasing the data storage capacity of the QR Codes is very limited and still operational.
***Objectives**:* The main objective of this work is to increase the data storage capacity of QR Codes in the IIoT domain.
***Methods**:* In the first part, we have introduced a 4-State-Pattern-based encoding technique to generate the proposed 4-State QR (4SQR) Code where actual data are encoded into a 4SQR Code image which increases the data storage capacity more than the traditional 2-State QR Code. The proposed 4SQR Code consists of four types of patterns, including Black Square Box (BSB), White Square Box (WSB), Triangle, and Circle, whereas the traditional 2-State QR Codes consist of BSB and WSB. In the second part, the 4SQR Code decoding module has been introduced using the adaptive YOLO V5 algorithm where the proposed 4SQR Code image is decoded into the actual data.

☆ Peer review under responsibility of Cairo University.
* Corresponding authors.
E-mail addresses: aminur@cse.green.edu.bd (M.A. Rahaman), arahman@niter.edu.bd (A. Rahman), ghulam@ksu.edu.sa (G. Muhammad).

https://doi.org/10.1016/j.jare.2023.10.006
2090-1232/

Results: The proposed model is tested in a Digital Twin (DT) framework using randomly generated 3000 testing samples for the encoding module that converts into 4SQR Code images successfully and similarly for the decoding module that decodes the 4SQR Code images into the actual data.
Conclusion: Experimental results show that this proposed technique offers increased data storage capacity two times than traditional 2-State QR Codes.

Introduction

The Industrial Internet of Things (IIoT) and other industry sectors are increasingly employing QR Codes in this fourth industrial revolution due to their easy use and wide range of application areas [1]. Humanity needs a better way to use the existing technology since we are mostly reliant on it, thus researchers are seeking a straightforward method of information sharing or storage. During this procedure, we store our data in a quad image, also referred to as a QR Code. Nearly 6.648 billion people, or 83.04% of the world's population (those who possess a smartphone), can utilize a QR Code, a sort of matrix bar Code that is used to store data quickly. This code can be useful in many different fields in modern days such as banking systems, e-commerce, transferring information through integration in images, industrial facilities, shopping malls, shipyards, food shops, billboards, magazines, and electronics product [2].

On the demand of the era, another recent technique is the digital twin (DT), which emerges with the vast possibility of solving a complex problem. A DT is a virtual replica of a physical object or system that can be used to test different issues with ease [3]. The advance of modern technologies like cloud computing, Artificial Intelligence (AI), Big data, blockchain, and IoT integration with DT has allowed us to create a physical twin of current status [4,5]. The research aims to use DT technology to optimize the QR Code system and bring new 4SQR. The DT can be used to improve the accuracy and efficiency of the 4SQR Codes, such as improving generation time, reducing the decoding time, and enhancing the accuracy, efficiency, and reliability of the 4SQR. However, Fig. 1 depicted excellent collaboration scenarios between a real QR Code system with the DT techniques that are currently used in the modern world [6].

On the other hand, traditional QR Fig. 2(a) depicts the structure of the QR Code [7]. There are several aspects of a QR Code. These include the code design, the fixed pattern, format information, encode mode, length of the data (Len), timing pattern, and finally, the data pattern. Furthermore, a QR Code can only store a limited number of bits in the image of the existing version of the QR Code. We come up with a way to improve the data storage capacity of the QR Code by double(2x) the current standard QR Code by introducing a new technique to the industries by proposing a new system.

We get to a new horizon of data storing in QR Codes. In the field of QR Codes, there has been a lot of work done by researchers in the last two decades [8], but there are lots of things that are still lacking. such as limited data storage, limited error-correcting system, limited usability, etc., The Two-State Pattern Technique (BSB & WSB) is adopted in the industry where we have introduced the "4-State Pattern Technique". The existing two-state has a masking part, but if the masking part gets distorted, it makes the whole QR Code unreadable. This is why we introduce the 4-State Pattern Technique QR Code which removes the masking pattern. Industrial costs can be minimized by the proposed technique. Moreover, QR Codes have some essential uses in the industry, but existing QR Codes have less data storage capacity which we can increase through our proposed "4-State Pattern Technique based Method".

Therefore, we aim to create a QR Code for collecting information with minimum time and computational cost and, increase the accuracy of the existing system. In the proposed system, several test cases are generated to test the data storage capacity, time complexity, and memory complexity. Our main aim is to reduce the limitations of convolution when increasing the data storage capacity and prepare a simple intellectual way to ensure all the techniques that have been followed for increasing capacity.

Research questions & contributions of the study

Q1. How can we increase the storage capacity of the proposed 4SQR-Code system?
Q2. How can we easily use an 4SQR-Code system and what are the test cases?
Q3. How much accuracy does an 4SQR-Code system provide?
Q4. How to improve accuracy and minimize computational cost and what are those different approaches?

The contributions of this paper are as follows:

- We have designed a structure for increasing existing system data storing capacity.
- We have implemented the 4SQR-Code to generate and detect the readability of the system.
- Also, a 4SQR-Code algorithm is proposed for increasing data storing capacity.
- In addition, we evaluate the data storing capacity and performance of the proposed model.

Organization

The rest of the paper has been formed as follows: We have studied and discussed the literature review in Related Works section. After that, Proposed 4-State QR Code Pattern section presents the structure for 4SQR; architecture for 4SQR; procedure for 4SQR. Moreover, an evaluation of the results and analysis of the proposed model are provided in **Experimental Results And Discussion** section. In sum, the authors concluded the paper in **Conclusion** section.

Related works

In this section, we have gone through some recent research works such as data hiding techniques [9–11], multiplexing techniques [12–14], use of color code techniques [15,16], use of data compression techniques [17,18], etc., but most of them are focused only on the outer level of the QR Code system instead of a more root level approach. QR Codes are used in several types of domain applications, particularly when transferring various types of data between electronic devices is needed. In essence, the fourth industrial revolution is the trend towards automation and data exchange [19] in manufacturing technologies and processes which include Internet of Healthcare Things [20], IoT [21], industrial Internet of things [22], cloud computing [23–25], cognitive computing, and artificial intelligence. Following automation and data exchange technologies, the QR Code might be a fantastic communication channel between many electronic gadgets. It means that the QR

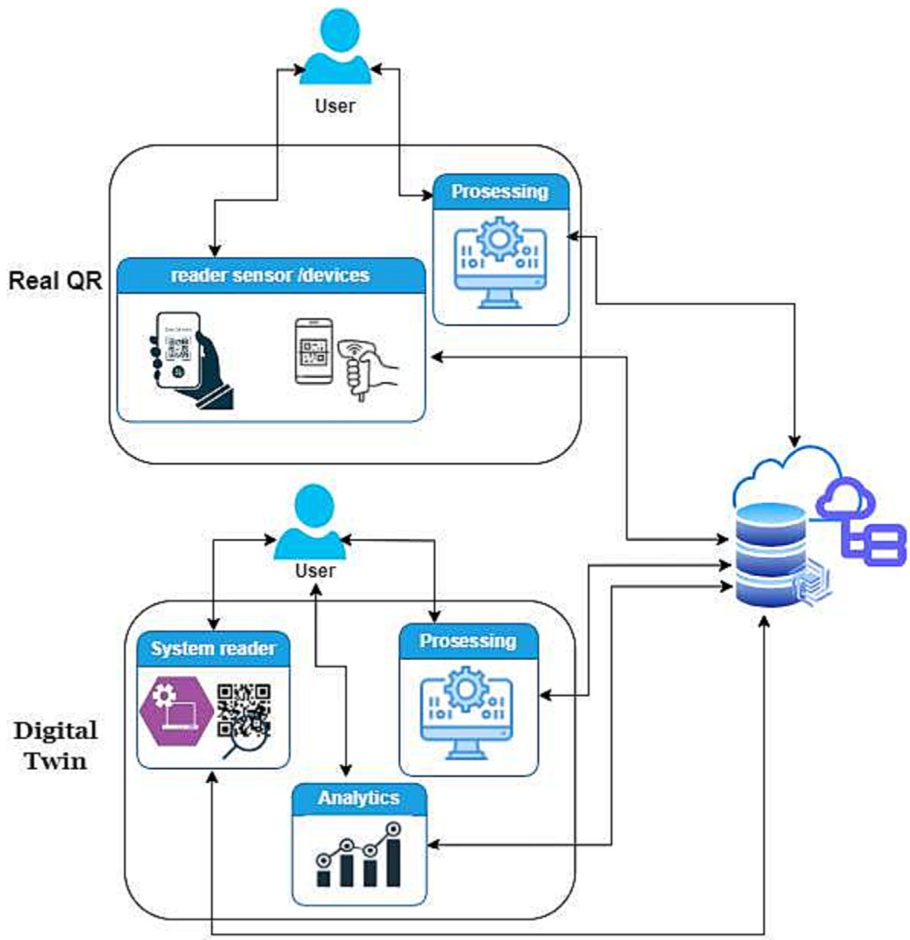


Fig. 1. Scenario of DT technology with real QR Code System.

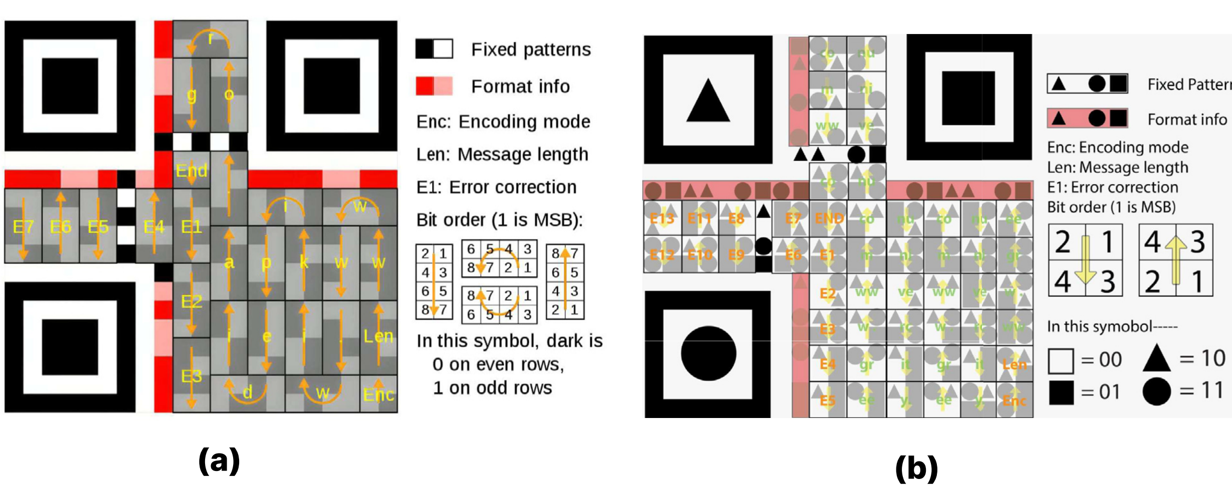


Fig. 2. (a) Traditional QR Code and (b) Proposed 4-State QR Code Structure.

Code requires a large amount of data storage capacity in order to exchange a large amount of data, which is possible if we increase the QR Code's data storage capacity. The rising data storage capacity may aid in meeting the needs of current technology. The author's objective in our suggested system is to expand the data storage capacity of the present QR Code technology. As a result, we should familiarize ourselves with the conventional QR Code system. There is also a need to learn about various strategies used in this sector, such as data concealing techniques, multiplexing techniques, color coding techniques, data compression techniques, and so on. We analyzed their approach in order to improve the work progression of concept implementation in our system.

Most systems have primarily concentrated on various methods (compression techniques, color code techniques, zip encoding) to tackle the limited storage issue, and they are not nearly there; as a consequence, we cannot consider their methodology to be the QR Code's successor.

Digital twin

The concept of DT originates from Michael Grieves in the year 2002 at a conference [3]. As the conceptual model guiding product life-cycle management, Grieves offered the DT. The actual product, its virtual representation, and a bidirectional data link between them make up DT, in accordance with his notion. In the context of product life cycles, such as simulation, integration, testing, monitoring, and maintenance, this concept focuses on DT. The DT method offers a crucial basis for linking physical items to their virtual counterparts, enabling greater learning and interaction over time and space. By incorporating third-party services, it does more than only assist in improving design and operations; it also establishes a new industrial ecosystem. For businesses like manufacturing, the change from a purely product-focused perspective to a more industrial one is crucial [26–31].

Compression techniques

We have analyzed a few compression techniques that have been used for the QR Code system, which has been focused on increasing the capacity of QR Code by applying different compression techniques [32,17,33]. After investigating numerous techniques, we have identified that there are some different parameters that are mainly focused on various models and algorithms like Huffman encoding [34], zip encoding, etc.; these types of techniques are the most common in the area which makes a great impact on visualizing the importance and areas of increasing capacity of QR Code. Ali et al. [17] have developed a system that enhances the QR Code capacity using a lossless compression technique, and they recognized the verification of secure e-documents. Their system mainly focused on using different hash values for integrity ensuring and compression using Huffman encoding, which is a lossless data compression algorithm. The main focus of this system was enhancing the QR Code capacity, which was satisfied by the feature of satisfying the security requirements. Moreover, using compression technique for QR Codes focusing the increasing capacity has been developed by Arora et al. [18], which system mainly focused on using zip encoding for data compression algorithm and then color QR Code for applying modified multicolor QR encoding technique. The main focus of this system was increasing the storage capacity, which resulted in a maximum of 29% increment with a difference of 14% to the minimum based on different levels of QR Code. We have noticed that the increasing storage capacity is very much convoluted between BSB & WSB QR Code and multi-colour QR Code. We have identified different parameters by analyzing some different compression techniques focusing on increasing capacity [35]. We have distinguished the convolution of storage capacity when using color instead of black and white. We also spot that there are some great techniques that enhance the capacity of QR Codes, which is very important. For these reasons, we have to find a better way to consider those two in a way that the techniques can fulfill their requirement and also remove the convoluted limitations.

Encoding–decoding techniques

There are various types of work that have been done focusing on encoding–decoding techniques. We have analyzed a few encoding–decoding related papers [14,36,17]. After investigating several methodologies, we discovered multiple parameters associated with various models and algorithms, such as Huffman encoding, multiplexing, and lossless compression techniques. Umaria et al. [32] Enhancing the data storage Capacity in QR code using Compression Algorithm and achieving security and Further data storage capacity improvement using Multiplexing. This paper proposes a technique for increasing data capacity by zip compressing and multiplexing and retrieving data by reversing. By using zip compressing and multiplexing, the system creates a QR Code with increased data capacity and provides data security. Another system developed by Sijia Liu et al. [37] Rich QR codes with three-layer information using Hamming code. This paper introduces a (n,n) secret sharing scheme, where $n = 2^p$. In this setup, there are three main roles: a secret distributor, a secret compositor, and n participants. The secret distributor encodes the second and third-layer information into multiple QR code shares, all of which can be accurately decoded using a standard QR code reader. During secret recovery, the second layer information is obtained through XOR operation, followed by the extraction of the third layer information.

Color code techniques

There are various types of work that have been done focusing on using the color QR Code technique. We have analyzed a few color QR Code that has been used for the different type of QR Code [38–42]. After investigating those systems, we have identified that there are different models and approaches for different QR Codes. Those different systems have various working processes. The system developed by Taveerad et al. [38] for the development of color QR Codes for increasing capacity. Moreover, they try to convert the bit stream into code ward hexadecimal [43] to use a different color than that hexadecimal code converts into binary. Those binary values are placed into the QR matrix and generate images. Galiyawala et al. [39] to increase the data capacity of QR Code using multiplexing with color-coding: An example of embedding speech signals in QR Code. This paper proposes a technique for increasing data capacity by multiplexing more QR Codes in the same version using color coding. Moreover, the proposed method has several black-and-white QR Codes that are multiplexed together. For every distinct binary pattern, a distinct combination of RGB (Black, Blue, Green, Red, Cyan, Magenta, Yellow, and White) weights is assigned to its new QR Code. This will generate a multiplexed single-color QR Code with increased capacity. By analyzing different color QR systems [40–42,44], we have identified that there are very few common patterns. Most of the patterns are different from one another and have distinct processes for encoding and decoding. For those reasons, we select a root-level pattern technique to ensure efficiency.

Standard technique

In the field of QR Code [7] ISO/IEC 18004:2015, "Information technology–Automatic identification and data capture techniques, 2015(E)". There are not many common patterns, but a few common patterns exist. Those common patterns used by most QR Codes, those common patterns are converting characters to binary, Solomon encoding, and converting a matrix to an image. All of the systems above use those common patterns.

Comparison with existing work

There are different types of techniques that are used by various types of systems like zip encoding, multiplexing, data embedding, Huffman coding, Solomon encoding, etc. For the need to increase the capacity of QR Codes. We identified that there are different type of convolution that happens while increasing data storage capacity. We also observe that there is a decrease in performance between the BSB and WSB or standard QR Code and Multi-Color QR Code [38,45] which indeed resulted in a focusing point for the proposed technique. There are very few common patterns where most of the patterns are different from one another but

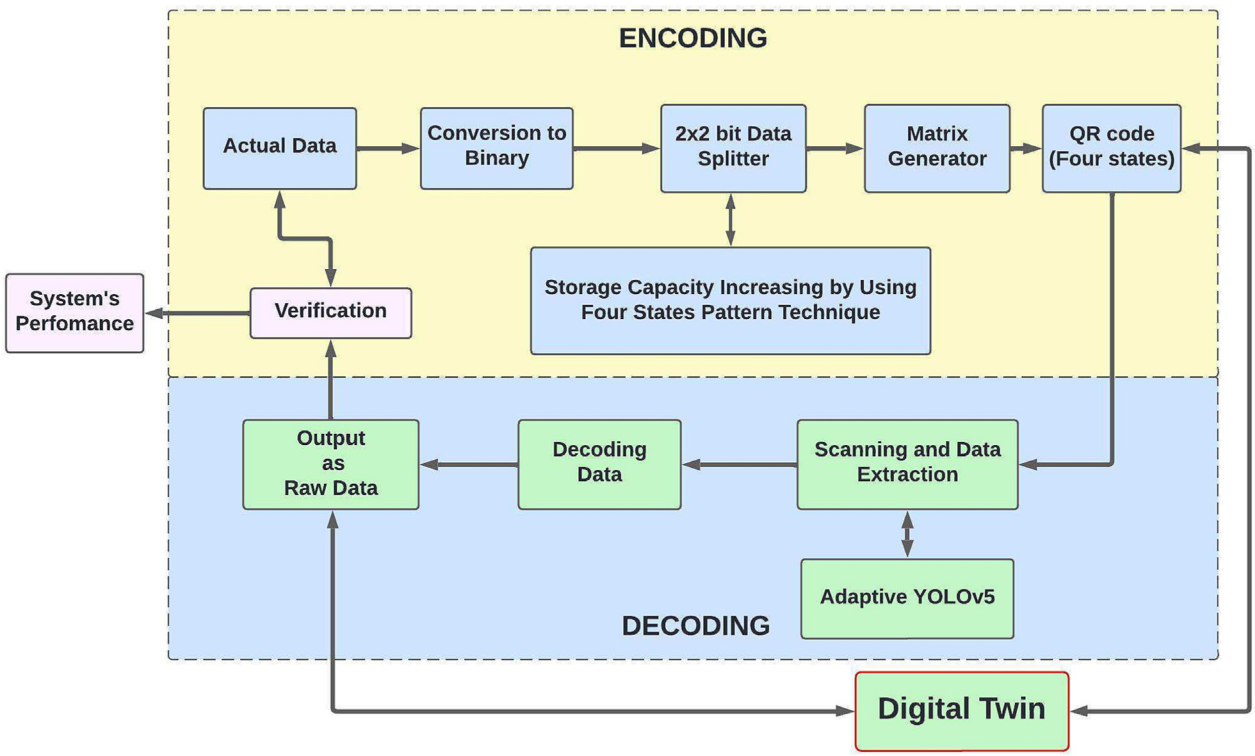


Fig. 3. The system architecture of the proposed 4-State QR Code pattern generation (encoding) and recognition (decoding) based on DT which is presented by Fig. 4 in detail.

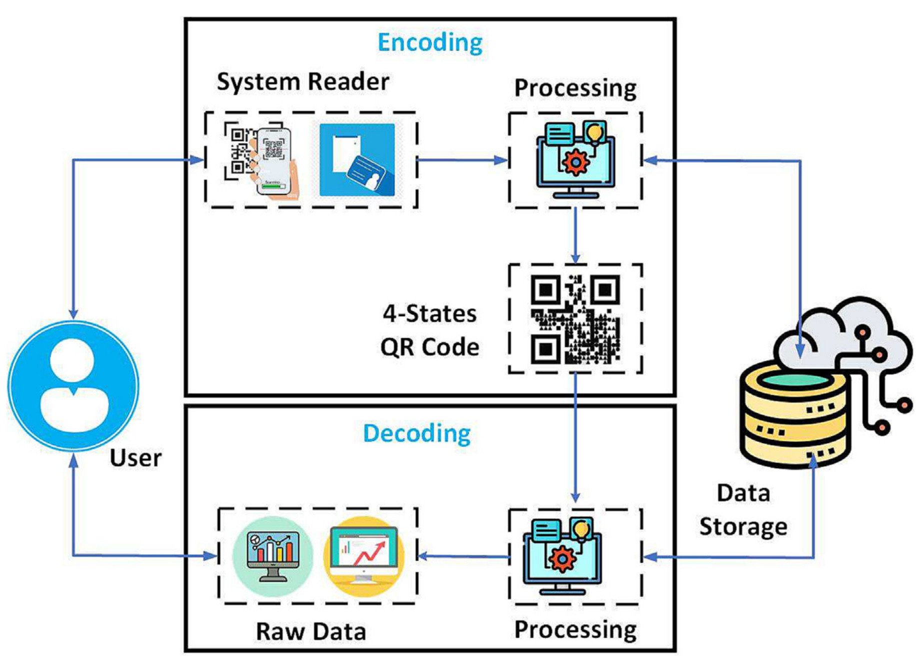


Fig. 4. DT Model for the proposed 4-State QR Code pattern.

many techniques of color QR Code increase the importance of these techniques [38–40,44] which indeed is resulted as a selecting key to going on a root level pattern technique for the proposed technique.

However, different literature reviews with a focus on increasing the capacity of QR Codes with techniques like compression and color code are mainly discussed. Those techniques have used various parameters and algorithms. Their proposed algorithms were

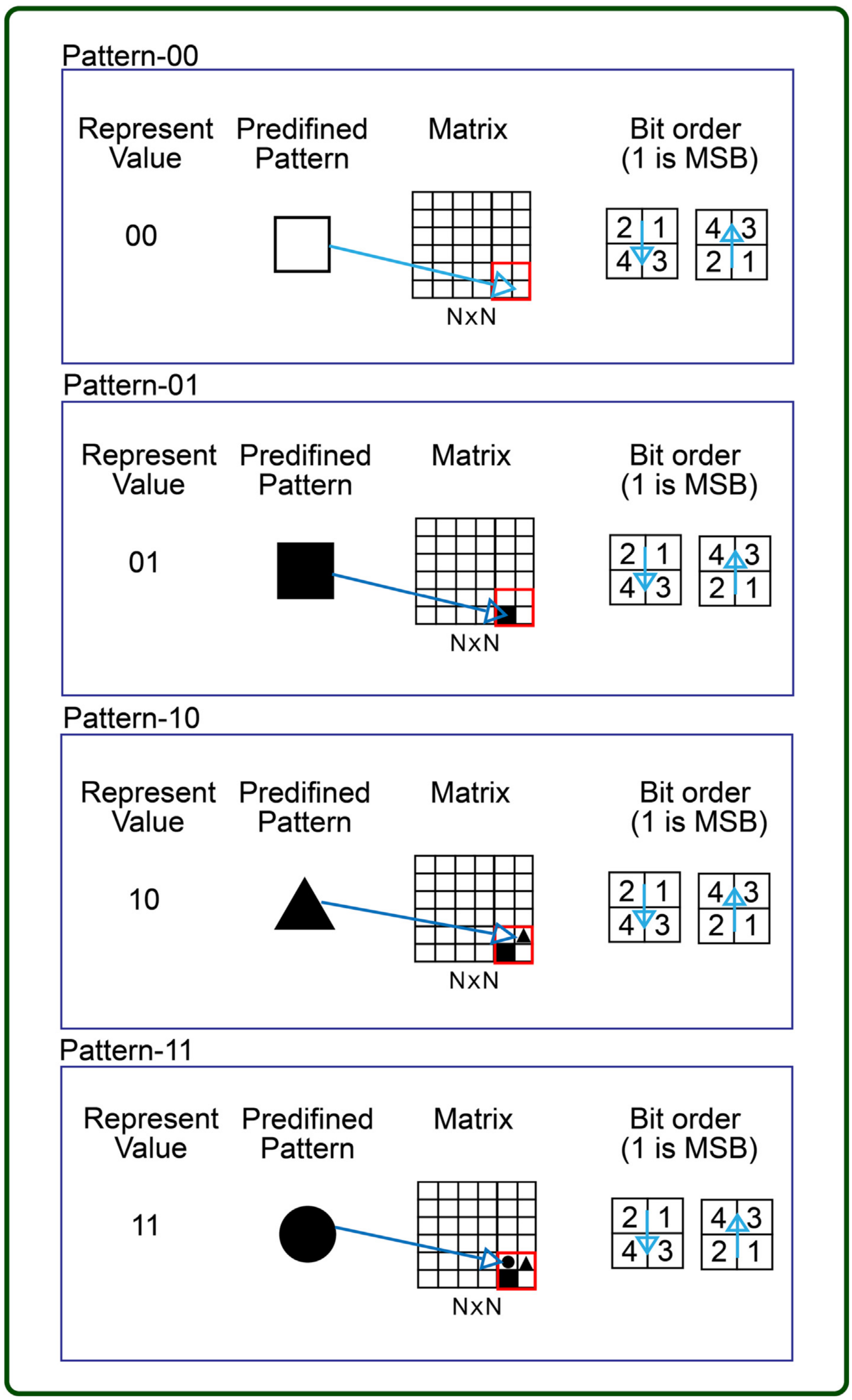


Fig. 5. 4-State QR Code image generation from NxN matrix where four patterns are represented by 00, 01, 10, and 11.

discussed in the schemes of several aspects, such as secrecy[46–52], durability, complexity, and storage capacity [7]. They have different strategies used to expand the capacity limit according to the business request. All of the papers shows various approach to improving the capacity of QR Code, but we can see carefully that most of the system process is nearly similar in type of working procedure but different in the process. When there are numerous types of processes going on in various models, we can see the difference between the complexity of the data and the capacity of data storage. On the other hand, that system's better performance, usability, and efficiency depend on the different system's working processes. When we thought about working with them and giving them an extra opportunity so that they could have some more areas to work with, we thought about the proposed system.

Proposed 4-state QR code pattern

System optimization is a process where the system gets optimized. For the process of optimization, creating a new system is very important. If we propose a new technique for QR Codes, then we also need to introduce a special method of testing. This paper proposed the "4-State Pattern Technique" to ensure the optimization of data storing capacity and getting output as decoded data.

The proposed method works for the QR Code generation system and also for the QR Code decoding system. In this system, there are 2 (two) main parameters, one is encoding, and another is decoding. where the encoding module creates a "4-State QR Code" and the decode module decodes the QR Code image. We implement the procedure from standard "ISO/IEC 18004:2015 of QR Code" [7].

Fig. 2(b) shows the structure of the proposed system, which follows the proposed four-state pattern technique. In our proposed system, there are mainly two parts which are: encoding and decoding where there are several processes that need to be followed according to the architecture. Fig. 3 in the encoding section main idea is to use two bits by two bits-by-bits binary data-splitter to split the data. Then we apply it to the matrix value placer and by doing so we get different patterns with respect to given data. Matrix to image generator places every two bits to blocks where, WSB (as binary number = 00), BSB (as binary number = 01),

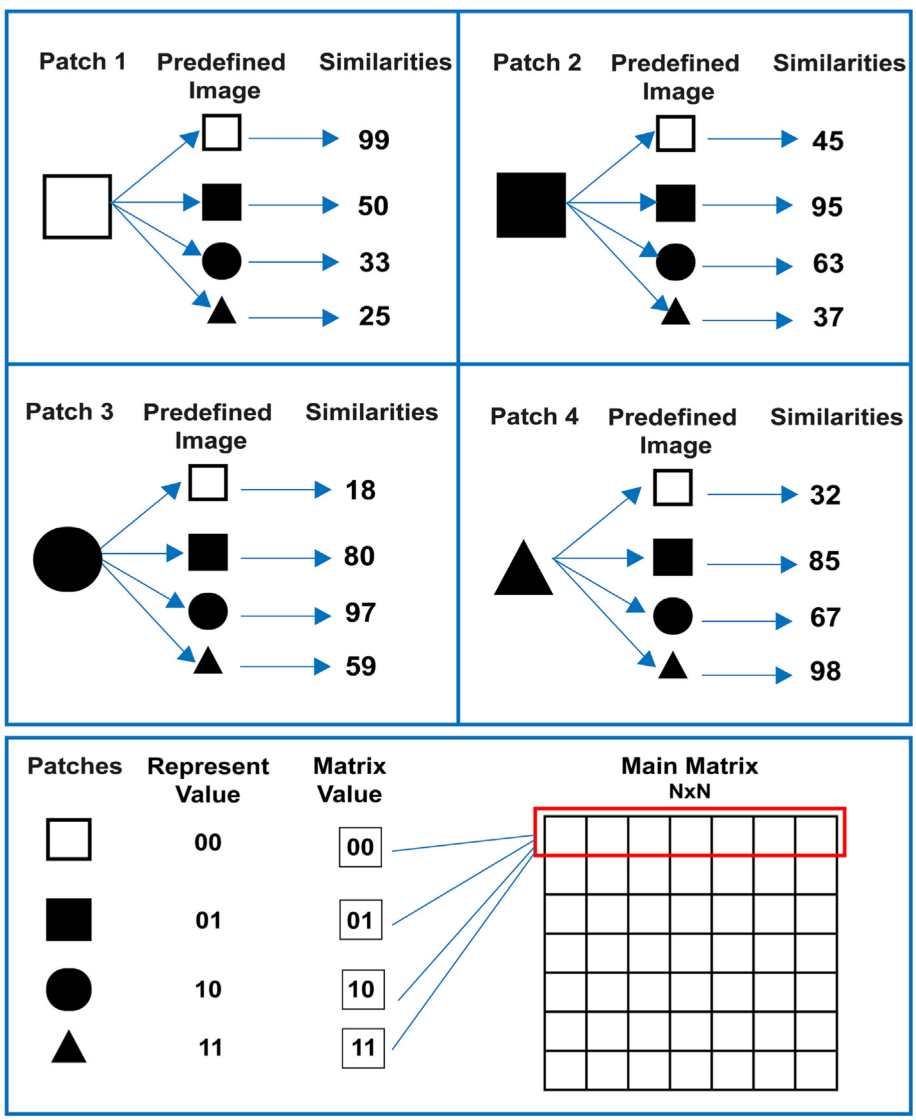


Fig. 6. Image Patch comparing and value assigning to Matrix in 4-State Pattern Technique.

Triangle (as binary number = 10), and Circle (as binary number = 11). Now we will get an output of the QR Code. (see Fig. 4).

Fig. 3, in the decoding section, the main idea is to decode actual data from a QR Code image. For this, the image is first sent to an image patcher after scanning it creates a data matrix from the image patcher data. Then sent to image-data-extraction data separation. after, data goes to the decoder to decode the Solomon[53] encoded data. Now the data gets separated and converted from digit to utf-8 character. Finally, we get an output which is the actual data. Moreover, Fig. 3 shows the architecture of the proposed system which follows the proposed "4-state pattern technique". where there are several processes that need to be followed accordingly with the diagrams.

In Fig. 3 in the encoding section, first the input module takes user data as input and passes that data to two different module data length modules and the iso-8859–1 encoding conversion module. The data length module finds out the length (how long the data is?) of the data, and the iso-8859–1 encoding conversion module converts data to digits. After doing the iso-8859–1 encoding conversion module it passes its data to two(2) modules, one is Solomon encoding and another is Conversion to binary. Solomon encoding converts its received data into polynomial data and generates digits. After that data length, iso-8859–1 encoded data, and Solomon encoded data converted into binary data. The system passes those data to 2x2 bits binary data splitter and creates a data group such as ”10 11 00 11 00 01 10 11”. Parallelly data length passes its data to the matrix size finder, which finds the accurate matrix for the given data and passes data to a module called matrix generator. The matrix generator builds a matrix filled with all the necessary fixed modules. For the final matrix generation, a module named matrix value placer gets 2x2 bits binary data splitter value and places those values into the matrix. Finally, Fig. 5 shows that there is a module called matrix to image generator, It creates an image by taking value from the matrix and pre-defined shapes/pattern of the images and creates a final image.

Fig. 3 shows the diagram of decoding, the first module is the input-image module that takes an image as input. Then passes into the image patcher, which decides the patch size and creates mini patches of the image. Then we applied our custom-designed adaptive YOLO v5 Algorithm, From that system gets the accurate size of an individual patch of images.

Fig. 6 shows the similarity measurement procedure of how those patches of the image go through a custom image comparator and compare those image with a system-predefined image to see their similarities after finding the highest number of similarities in which image get the highest number of, the predefined image has its corresponding integer value. Those integer values convert into 2(two) 2-digit binary values and are placed in a blank matrix. Finally, the system gets a matrix filled with binary values. After that image-data extractor goes through a specific way to get the data from the matrix and store it in a data stream. The next module is the data separator, which separates data into 2(two) blocks, one is the ”data block” and another is a ”Solomon-encoded data block”. After that next module decodes Solomon-encoded data and returns a data stream. From there data block & decoded data get compared, after comparing those output data and Solomon-decoded data. If compare result is similar, then go to the next module, else correct those error bits. finally, the final module is the digit to utf-8 character conversion module, in this state module system converts the data into its corresponding utf-8 character and returns it as output.

Algorithm 1. Algorithm of 4SQR-Code encoding

```
Input  : N_i
Output: F_i
for ( i=0; N_i ≠ 0; ++i) do
    L = i;
    return L;
end
N_i = rsc.encode(N_i);
while (N_i ≠ 0) do
    if (N_i[i]in list(ISO-8859-1.keys())) then
        keys ← N_i[i];
    end
    KL_i ← keys;
    i++;
end
while (N_i ≠ 0) do
    for (key, value in ISO-8859-1.Items()) do
        if (KL_i in key) then
            D_d ← value;
        end
    end
    return D_d;
end
RS_enc = rsc.encode(D_d);
while (L ≠ 0 and D_d ≠ 0 and RS_enc ≠ 0 ) do
    D_i ←''.join(f''{ord(i) : 08b}''for i in (L, D_d, RS_enc));
    return D_i;
end
n = 2;
for (i in range(0,len(D_i),n)) do
    Sd_i = D_i[i : i + n];
    tsd ← Sd_i;
end
QR_data ← tsd;
M = M_g(Ma_s(L));
M.F_p();
M.α_i(QR_data); M.δ_i();
Display F_i;
```

Proposed Algorithm 1’s Procedure

- Step 1: First take input that is going to use to encode.
- Step 2: Find the Data length.
- Step 3: Convert the character to a digital character.
- Step 4: Apply Solomon Encoding.
- Step 5: Converts steps 2,3,4 values into binary values and adds them to a list.
- Step 6: Split those binary values 2 bits by 2 bits.
- Step 7: From value finds the matrix size & generates a blank matrix.
- Step 8: Apply all the necessary patterns (version info, format info, position pattern, alignment pattern & timing pattern) to the matrix.
- Step 9: Apply list value to matrix & convert matrix into image.

Algorithm 2. Algorithm of 4SQR-Code decoding

```
Input  : I
Output: DisplayText
I = imageops.grayscale(Image.open(I));
  Im_a = np.array(I);
  for ( i ← 0 to ending) do
    if (image_array[i][j]==0) then
      i++, j++
    end
    i, j;
end
M_i = β_i(Im_a, i, j);
  x1=0 ; y1=0;
  for (w in range(0,I.shape[0],i)) do
    for (h in range(0,I.shape[1],j)) do
      y1 = y+i ; x1 = x+j;
       tiles = I[y:y+i,x:x+j]
       cv2.imwrite(path + I[i][j], tiles);
    end
end
for (i ←  0 to ending) do
  patch + str([i]) = path + I[i];
   cmp_r = ssim(patch + str([i]), PI);
   data ← cmp_r;
   data;
end
D_s = data_extrator.extract(data);
  Sd_i = [data[i : i + n]for i in range(0, len(data), n);
  for (i in Sd_i) do
    if (i=='00000000') then
      break;
    else
      fbs ← i;
       return fbs;
    end
end
for (i in range(Sd_i, −1)) do
  if (i ==' 00000000') then
    break;
  else
    RS_d ← i;
     return RS_d;
  end
end
D_s = bts.stc(fbs);
  d_d = RS_d.rsdecode();
  if (D_s == d_d) then
    Display D_s;
else
  Display d_d;
end
```

Where, N_i = Number of string, F_i = 4SQR image, i = Counter, L = Length, RS_{enc} = RS encoded data, KL_i = $keylist[]$, D_d = $digitaldata[]$, D_i = $Data[]$, Sd_i = Splitted data, M_s = Matrix size, Ma_s = Matrix_size(), M = Matrix, $M_g()$ = Matrix_generator(), $F_p()$ = Fixed_pattern(), α_i = Apply data, δ_i = image_generation(), I = Image, D_S = data stream, PI = Predefined Images[$I0$,$I1$,$I10$,$I11$], Img = image, Im_a = image array, M_i = Main image, β_i = $removeborder()$, cmp_r = compare result, fbs = formatted binary stream, RS_d = RS data, bts = $binary\_to\_string$, $stc()$ = $strconvert()$ and d_d = decoded data.

Proposed Algorithm 2's Procedure

- Step 1: First take an image as input.
- Step 2: Find a single shape of the image. and create an n*n image patch.
- Step 3: Match image patches with predefined images and place values into the matrix.
- Step 4: Check matrix size. If valid then go to the next step, if not valid repeat step 2 & 3 to find a valid matrix.
- Step 5: Extract data from the matrix and split data into 2 parts, the data part, and Solomon's encoding(rs) part.
- Step 6: From the data part, extract the encoding mode and data length
- Step 7: Decode the Solomon encoding using the rs decoding technique. Convert values to characters using UTF-8.
- Step 8: Match rs code with data, if match successfully then display data else correct wrong data with Reed–Solomon error correction technique if Reed–Solomon error correction technique fails repeat from step 1.

Experimental results and discussion

In the experiment part, we have used 12 GB RAM and an Intel Core i5 with 7^{th} gen processor with a base speed oimageGHz. We have also used pycharm and spyder for Python and OpenCV wrapper, imageIo wrapper, and image_similarity_measures wrapper have been used. Moreover, 64-bit Operating Systems with Windows 10® and Ubuntu 20.04 LTS have been used as an implementation platform. For testing purposes, we have created a few test cases and tried those test case scenarios to see whether our system can perform accurately or not. We have generated a list of 4SQR Code samples to test our system. Furthermore, we have shown a competitive analysis of memory and time in Table 1.

Performance metrics

Performance evaluation is the assessment of computer systems which includes their components, processes, and outputs. In performance evaluation there are a few parameters, those parameters are response time, latency, speed, throughput, and memory usage.

Response time: response time is the time taken by a system to do a task and return the result.

$$\Delta t = M + \lambda p - \alpha \quad (1)$$

$$start\ time = \Delta t_0$$

$$end\ time = \Delta t_1$$

$$time\ difference\ \Delta t_D = \Delta t_1 - \Delta t_0$$

Table 1
Test data & images of our encoding proposed system.

Input Data	Asif Mahmud Rial	Rokibul Islam	Ababil Islam Udoy
Length	16	13	17
Maximum Processing Memory Consumption (MB)	115.76	115.76	115.76
Execution Time (sec)	1	1	2
Output			
Input Data	Dr. Muhammad Aminur Rahman	Green University Of Bangladesh	I love my country
Length	26	30	17
Maximum Processing Memory Consumption (MB)	115.76	115.76	115.76
Execution Time (sec.)	1	1	1
Output			
Input Data	b3701e7c184b0bbf71		
19752603314c49a690cb26	e5f37c1b03f961a4b67ed44		
d6e9534d9b73665f9	be66ec47c0176381bc		
a1cbc348c5111e6aef1656			
Length	40	40	40
Maximum Processing Memory Consumption (MB)	155.22	155.22	155.22
Execution Time (in sec.)	1	1	1
Output			

Where, Δt = Time of a position, M = Mean anomaly, λ = The ecliptic longitude of the periapsis, α = The right ascension of the apparent sun, Start time = Δt_0 and End time = Δt_1.

Speed: The term speed is usually referred to the clock speed of the processor. It means how fast a processor can complete its task. We have tested our system on different machines and also found different results in different machines Because machine-to-machine speed defers.

Memory usages: Memory is the module where load and run applications(memory means primary memory). How much memory is needed in a system to run an application of that system? That's why memory is an essential parameter. In this system, we have taken that into consideration. We have also analyzed our system memory usage in this section.

$$M_U = M_T - M_F \quad (2)$$

$$M_P = MAX(M_U)$$

$$\therefore Peak\_memory\_usages = M_P$$

Where, M_U = Total used memory, M_T = Total Memory, M_F = Free Memory, $MAX(M_U)$ = Maximum memory usage from a group of sample.

Testing Data

In the system architecture of the proposed 4SQR, we have observe two(2) input, so for this proposed 4SQR technique author generate some testing data.

In the fields of QR Code, there are not many test cases, and mainly those test cases are in the input data stream and input images. On those, there are not many places for bugs/errors to happen, those test cases are analyzed in this section. If we follow those test cases and create test data, then there is no place for errors to occur in our system. In the performance evaluation parameters, we have shown some of our test data, on those test data we have used only "English Characters" because we have developed our system only for "English Characters". We have processed our data by its length and sorted those data. After we have collected our system run-time and main memory cost. Moreover, we have discussed in result in the result of the proposed system section. From those test results, run time, and memory cost we have come to a position where we could say our proposed system works adequately.

In this work, image generation is a testing case Fig. 3 as shown in our proposed system. There is the use of image generation as a testing case to test our system. When any data gets input in the proposed system **Algorithm 1** then the data is converted into an image. We generated 3,000 testing samples, and then we ran the encoding system for 3,000 different data sets and got the required sample. These samples are used for the next test case to see how the system hold up to different test case. The generation of an image is the process of image generation in Table 1 as shows that the image goes to a module and different sub-modules proceed with their individual task received from the system module. An image generation system should be robust to any changes in data. Because when taking sample data, data always may not be in the clear string, the system needs a clean data stream as input.

In the testing section, scanning is also a test case, as expressed in Fig. 3. Our system reads the file and creates mini patches with custom YOLO v5, we have tested our patches with the predefined image. Moreover, our system works with those patch system to create a matrix, and from that matrix system get output. We have tested our system externally and graphed it with standard QR system output. The scanning of the image is the process of image reading, in the image (YOLOv5), goes and gets read and converted the image to the matrix. Next, different sub-modules proceed with their individual task like data extraction, data separation, data decoding, and converting data stream to UTF-8 characters and complete all the processes received from the system module. An image reading system should be robust to any changes in the image. When taking sample image data, image data may always

Table 2
Test image & output data of our decoding proposed system.

Input Image			
Execution Time (in sec.)	5	5	5
Maximum Processing Memory Consumption (MB)	9.597498	9.597498	9.597498
Length	16	13	17
Output	Asif Mahmud Rial	Rokibul Islam	Ababil Islam Udoy
Input Image			
Execution Time (in sec.)	5	5	4
Maximum Processing Memory Consumption (MB)	9.469864	9.469864	9.469548
Length	26	30	17
Output	Dr. Muhammad Aminur Rahman	Green University Of Bangladesh	I love my country
Input Image			
Execution Time (in sec.)	7	7	8
Maximum Processing Memory Consumption (MB)	13.182203	13.183273	13.182011
Length	40	40	40
Output	b3701e7c184b0bbf711975 2603314c49a690cb26	e5f37c1b03f961a4b67ed4 4d6e9534d9b73665f9	be66ec47c0176381bca1c bc348c5111e6aef1656

Table 3
Comparative analysis of character capacity(Existing vs. proposed system

Version	Error Correction Level	Character Capacities Numeric Mode	Alpha numeric Mode	Byte Mode	Kanji Mode	Proposed System
1	L	41	25	17	10	34
	M	34	20	14	8	28
	Q	27	16	11	7	22
	H	17	10	7	4	14
2	L	77	47	32	20	64
	M	63	38	26	16	52
	Q	48	29	20	12	40
	H	34	20	14	8	28
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
39	L	6743	4087	2809	1729	5618
	M	5313	3220	2213	1362	4426
	Q	3791	2298	1579	972	3158
	H	2927	1774	1219	750	2438
40	L	7089	4296	2953	1817	5906
	M	5596	3391	2331	1435	4662
	Q	3993	2420	1663	1024	3326
	H	3057	1852	1273	784	2546

come in size and condition, and the system needs to process it as input. We can observe in Table 1. The raw string is used to generate the "4-State QR Code" and those QR Code images used in decoding Table 2.

Table 3 depicts the result of the proposed system and existing system, The existing system's data capacity is shown and on the other hand, we have shown the result of our proposed system. According to their study, they found a complex growth of their data, and we have also found complex growth of their data we have compared the existing system and our system data. We have designed our system in the base of Byte Mode, where we have double Byte mode data in our system.

Character capacity comparison

Fig. 7 depicts the character capacity of the QR systems (Conventional QR vs. 4SQR) compared to the number of various versions (5 to 40) of the QR Code system. At the beginning of the graph, there are no differences between the existing and proposed systems. Also, for a small number of versions, proposed and existing systems are almost similar. However, with increasing the number of QR versions (Over 10 versions), character capacity also increased linearly. However, it is clear that the proposed 4SQR Code system has better character capacity compared to the conventional QR Code system. For the 40 versions of the QR Code, the proposed

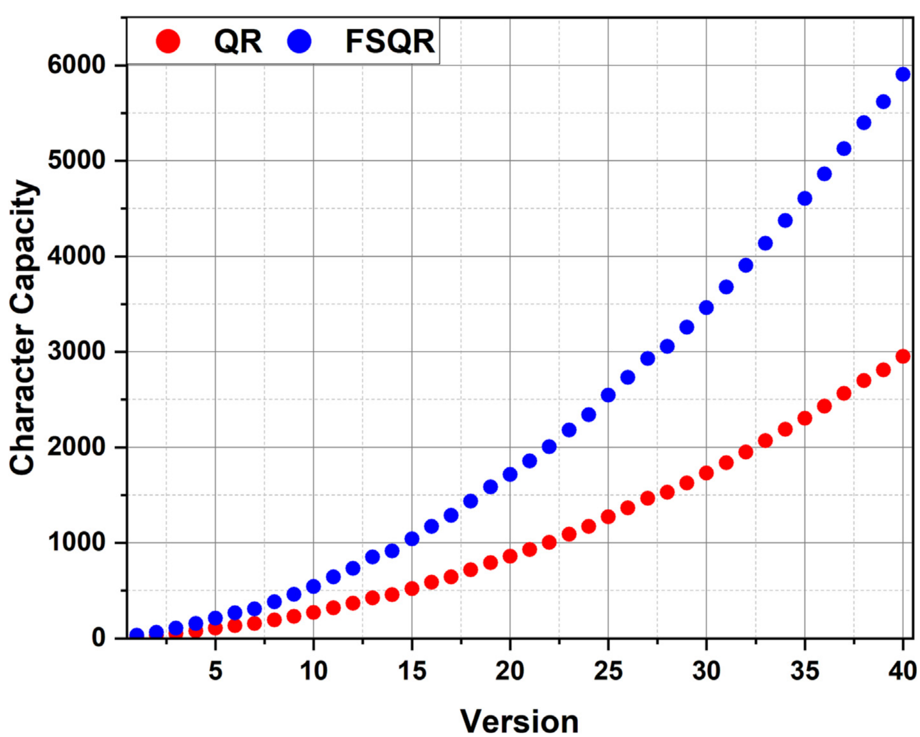


Fig. 7. Character Capacity Comparison with respect to the Versions.

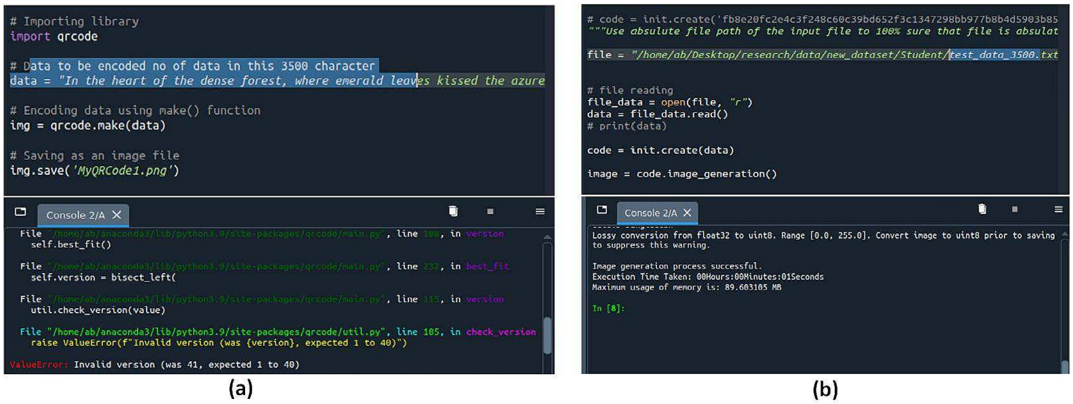


Fig. 8. Example outputs of the comparison of data storage capacity (a) Traditional QR codes fail to store large amounts of data (b) The proposed system (4SQR Code) can store the large data.

Table 4
On which no of data and specific version proposed system can generate 4SQR and QR can't

NO of Data OR Character Capacity	Version	QR	4SQR
17	1	✓	✓
25	1	X	✓
34	1	X	✓
32	2	✓	✓
40	2	X	✓
62	2	X	✓

4SQR-Code system has reached a peak of 6000 character capacity whereas conventional QR is on 3000 character capacity which is almost half of the proposed system. Overall, it can be seen that the character capacity of the proposed system is much better than the conventional QR system.

In Table 4, we have shown what number of versions and what number of characters conventional QR Codes can create and what QR can't be created. On the other hand, we test without data that whatever it is possible to create double the data on a specific version and we show that result in the Table 4. Finally Table 4 depicts that our system can perform its task as it intended. Also, Fig. 8 depicts traditional QR codes that fail to store large amounts of data but the proposed system can store the large data.

Fig. 9(a) depicts the complex timing growth of the QR systems (Standard QR vs. 4SQR) compared to the number of nodes. For the less (10 to 50) numbers of data, QR and 4SQR take 60 ms and 160 ms respectively. With the increasing of No. data both QR systems time(ms) also increasing. Moreover, we have observed that on average conventional QR takes around 65 ms on the other hand, the proposed system takes around 200 ms. It's a little bit behind the conventional QR Code system but this is not much behind as a newly proposed system, the reasoning is that the proposed system works instantaneously for the user. Also, the reason for taking more time is that the proposed system needs to convert the generated binary into a 2x2 bit split binary, and for those 2 digits binary, proposed systems need to create a unique pattern; generating those pattern take time and memory. (see Fig. 10,11).

Fig. 9(b) shows the memory cost(MB) of the QR systems (Conventional QR vs. proposed 4SQR) compared to the number of nodes. From the beginning of the graph, there is a big difference in memory cost. At first, we noticed that standard QR takes around 1 MB to 2 MB of memory, on the other hand, the proposed system takes around 98 MB to 130 MB of memory. For 1000 to 3000 numbers of data, this graph shows a straight line for the proposed QR Code system, on the contrary, the standard QR system shows a straight line from the very beginning. The reasoning is that conventional QR doesn't need to work with the extra module to create a QR Code, on the other hand, the proposed system needs to follow a few extra procedures to create a 4SQR Code with much more data. For that reason, In Fig. 9(b) we have observed a high consumption of memory in the proposed system.

Fig. 9(c) depicts the decoding time of the QR systems (standard QR vs. proposed 4SQR) compared to the number of nodes. From the graph, we have observed that conventional QR takes around 50 ms to 70 ms time(ms), The reasoning is that conventional QR doesn't need to go through many processes such as image patcher and image comparator. Conventional QR first converts the image into a grayscale/binary raster image and collects data for that reason to take less time. On the other hand, the proposed system has to go through an image processing procedure and use custom YOLO v5 to create a perfect patch, and from those patches, the system compares the patches' values and does other possessing and produces data, for that reason, 4SQR takes much more time compared to conventional QR system. However, The 4SQR is much more accurate in contrast to the conventional QR due to image processing that takes much more time to operate.

Fig. 9(d) depicts the decoding memory of the QR systems (Conventional QR vs. proposed 4SQR) compared to the number of various nodes. For less no of (50 to 100) nodes, the standard QR Code takes less than 1(MB) of memory, on the other hand, the proposed QR Code takes 13(MB). Moreover, after 500 no of nodes, the 4SQR Code is increased linearly, on the contrary, there is little change observed in the standard QR Code system. However, in this graph, we have also observed that conventional QR beat our system by far, but it's not a problem for current devices, the reason is that nowadays current devices use much more memory than the proposed system requirement. For that reason, it is not a matter of worry but in future studies, we can optimize the proposed system so that the 4SQR system can perform better with much less memory.

Applications

- During the COVID-19 pandemic [54], many restaurants began the use of QR Codes to provide a paperless menu to ensure contactless and safe dining.
- It can be applied for product packaging, process management, product stocking & product picking such as luggage Tags.

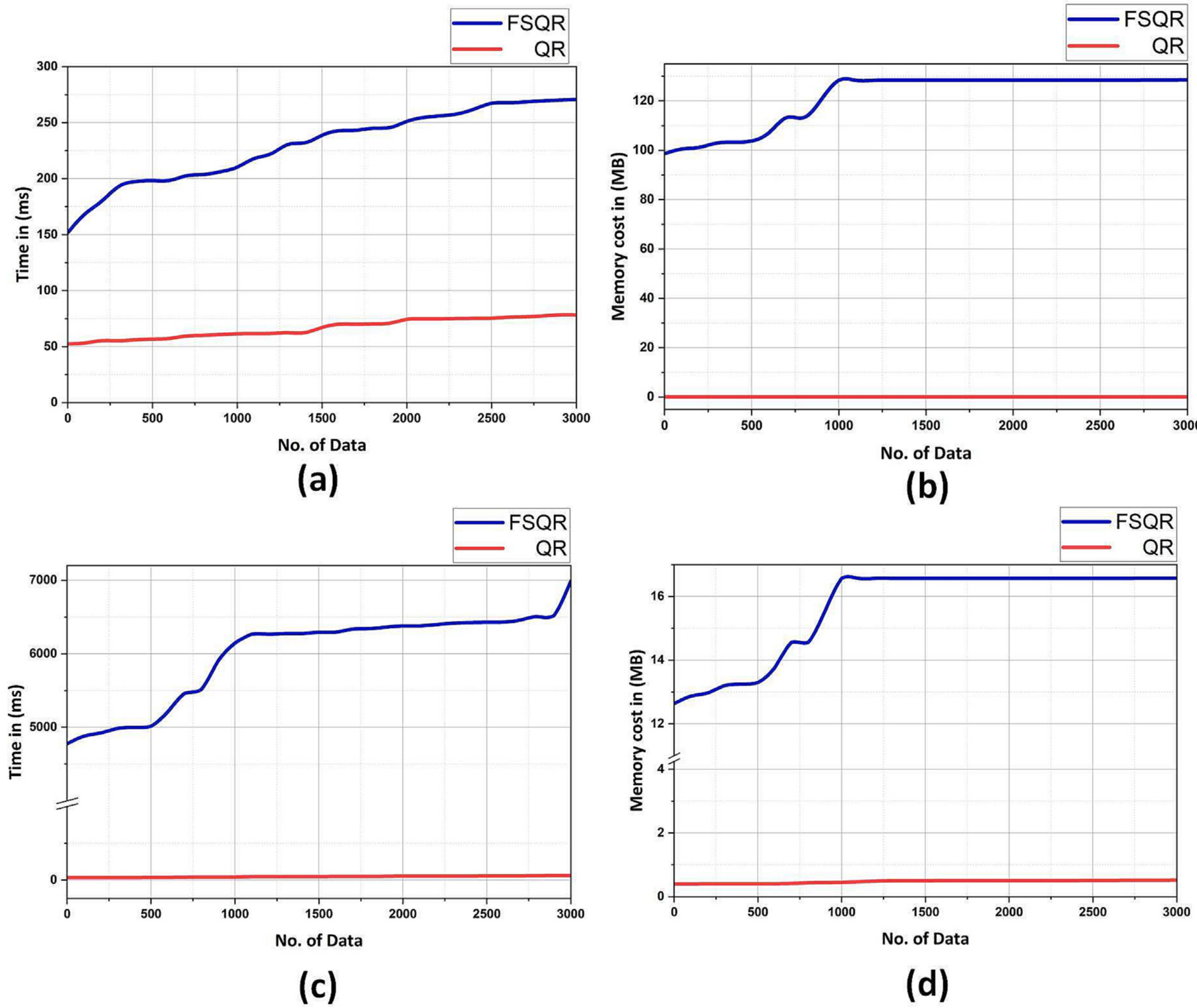


Fig. 9. (a) Encoding Time Cost, (b) Encoding Memory Cost, (c) Decoding Time Cost, and (d) Decoding Memory Cost(MB) of the proposed System with respect to standard QR Code.

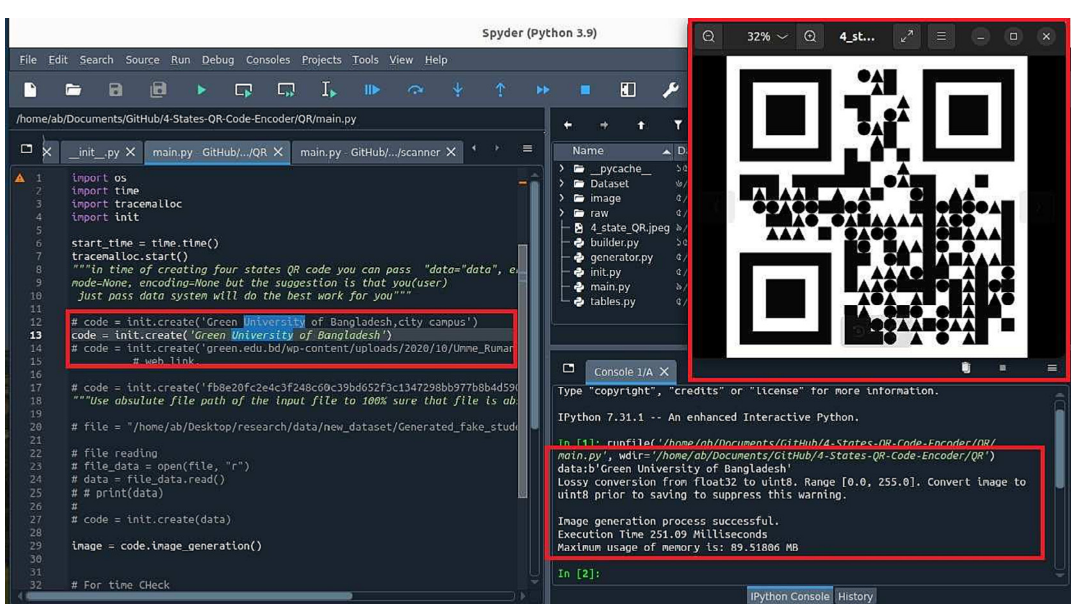


Fig. 10. Example output of the proposed system for Encoding process.

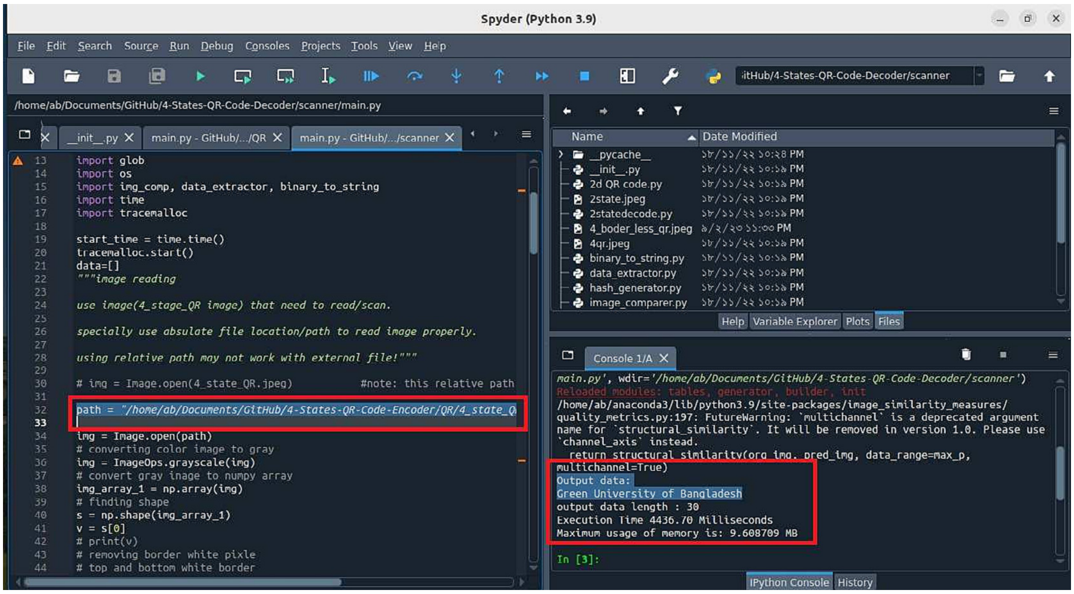


Fig. 11. Example output of the proposed system for decoding process.

- In many places people use bar Codes to scan products and also people use their hands to generate invoices.
- Nowadays, QR is used for Point of Sale (POS) such as American Express, MasterCard, Bkash, Rocket, and Nagad which introduced a feature on their mobile banking apps that enables sellers to accept payments through QR Codes.
- Moreover, Social Media such as Facebook, WhatsApp, and Snapchat allow users to follow accounts by scanning QR Codes.
- Furthermore, in the medical field application of QR is getting border, and entry of the contents of prescriptions is gotten smoother through the reading of QR Codes printed on prescriptions.
- Application scenes in the leisure field like walking in museums and historical sites can give users field information just by scanning it.
- Creates ads for mobile applications through 4SQR-Codes. For instance, Chinese Organizations/ corporations launch their application through drone shows, they show QR to scan and download their application to the audience.
- We can use the 4SQR Code in libraries to track our books.

Limitations and open challenges

- 4SQR has introduced only two versions (version1 & version2) future challenge is to generate the rest of the version.
- Currently 4SQR Code supports only the English language. It is possible to bring new languages to this system.
- This work found the possibility of bringing more states such as 8, 16 & 32 states for QR Codes.
- Moreover, 4SQR has an enormous increase in data storage capacity, but in the main time finding storage capacity, we had to face the limitation of time and memory compared to standard QR. This could be minimizable by uncovering a new and better algorithm.

Conclusion

The usage of the DT in the 4SQR system increases the data storage capacity by 2x as compared to the traditional QR Code. The DT allows us to increase storage capacity and opens up wider working domains. Considering the aims of this work, this research's main intention was to create a 4SQR-Code based on the traditional QR Code. Moreover, DT brings us the virtual simulation that opens up the possible expansion of the 4SQR capacity by 4x, 8x, 16x, and 32x. It can be noted throughout the work that the proposed algorithm brings a new and improved version of the QR Code, and this system can be used more easily and more efficiently in practical applications. Furthermore, the proposed algorithm is discussed in several aspects, such as complexity, and data storage capacity. However, this system needs to polish up more to get the best output from the system and also can be possible to work with this system and bring new and improved versions of the QR Code. Finally, the 4SQR system can generate an extensive number of QR Codes with increased data storage capacity. Also, after some future development, the 4SQR system can be the pioneer in the QR Code field.

Compliance with ethics requirements

This article does not contain any studies with human or animal subjects

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The authors acknowledge the Researchers Supporting Project number (RSP2024R34), King Saud University, Riyadh, Saudi Arabia.

References

[1] Scanova. Qr code statistics 2022: Up-to-date numbers on global qr code usage, [Online; accessed 14-February-2022]; 2022. URL https://scanova.io/blog/qr-code-statistics/
[2] Wikipedia contributors, Qr code — Wikipedia, the free encyclopedia, https://en.wikipedia.org/w/index.php?title=QR_code&oldid=1127627244, [Online; accessed 20-December-2022]; 2022.
[3] Grieves M. Digital twin: manufacturing excellence through virtual factory replication. White paper 2014;1(2014):1–7.
[4] Singh M, Srivastava R, Fuenmayor E, Kuts V, Qiao Y, Murray N, Devine D. Applications of digital twin across industries: a review. Appl Sci 2022;12(11):5727.

[5] Huang S, Wang G, Yan Y, Fang X. Blockchain-based data management for digital twin of product. J Manuf Syst 2020;54:361–71.
[6] Revetria R, Tonelli F, Damiani L, Demartini M, Bisio F, Peruzzo N. A real-time mechanical structures monitoring system based on digital twin, iot and augmented reality. 2019 Spring Simulation Conference (SpringSim). IEEE; 2019. p. 1–10.
[7] Wave D. Information technology automatic identification and data capture techniques qr code bar code symbology specification. Int Organ Standard, ISO/IEC 2015;18004:23.
[8] Deineko Z, Sotnik S, Lyashenko V. Confidentiality of information when using qr-coding.
[9] Huang H-C, Chen Y-H, Chang F-C, Lu Y-Y. Enhancements of data hiding applications with qr code concealment. In: 2021 IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech). IEEE; 2021. p. 425–7.
[10] Rahman A, Hossain MS, Muhammad G, Kundu D, Debnath T, Rahman M, Khan MSI, Tiwari P, Band SS. Federated learning-based ai approaches in smart healthcare: concepts, taxonomies, challenges and open issues. Clust Comput 2023;26(4):2271–311.
[11] Rahman A, Hasan K, Kundu D, Islam MJ, Debnath T, Band SS, Kumar N. On the icn-iot with federated learning integration of communication: Concepts, security-privacy issues, applications, and future perspectives. Future Gener Comput Syst 2023;138:61–88.
[12] Tan L, Lu Y, Yan X, Liu L, Zhou X. Xor-ed visual secret sharing scheme with robust and meaningful shadows based on qr codes. Multimedia Tools Appl 2020;79(9):5719–41.
[13] El-Sheref IM, El-Licy FA, Asad AH. Improving performance of the multiplexed colored qr codes. Int J Adv Comput Sci Appl 11 (3).
[14] Lu Z, Lv W, Zhu Y, Yang D, Zhou X, Wang H, Shi Y. Optical information encryption based on partially-update iterative system with azimuth multiplexing. Opt Commun 2022;510:127899.
[15] Yang Y, Lei C, Chen R, Huang H, Tan H-Z. Fast image process and detection of distorted blur-readable 2d barcode. In: 2022 7th International Conference on Signal and Image Processing (ICSIP). IEEE; 2022. p. 322–6.
[16] Chen R, Zheng Z, Yu Y, Zhao H, Ren J, Tan H-Z. Fast restoration for out-of-focus blurred images of qr code with edge prior information via image sensing. IEEE Sens J 2021;21(16):18222–36.
[17] Ali AM, Farhan AK. Enhancement of qr code capacity by encrypted lossless compression technology for verification of secure e-document. IEEE Access 2020;8:27448–58.
[18] Arora M, Verma AK, et al. Increase capacity of qr code using compression technique. 2018 3rd International Conference and Workshops on Recent Advances and Innovations in Engineering (ICRAIE). IEEE; 2018. p. 1–5.
[19] Baidong H, Yukun Z. Research on quickpass payment terminal application system based on dynamic qr code, in: Journal of Physics: Conference Series, Vol. 1168, IOP Publishing; 2019. p. 032059.
[20] Islam MM, Nooruddin S, Karray F, Muhammad G. Multi-level feature fusion for multimodal human activity recognition in internet of healthcare things. Inform Fusion 2023;94:17–31.
[21] Rahman A, Islam MJ, Montieri A, Nasir MK, Reza MM, Band SS, Pescape A, Hasan M, Sookhak M, Mosavi A. Smartblock-sdn: An optimized blockchain-sdn framework for resource management in iot. IEEE Access 2021;9:28361–76.
[22] Rahman A, Islam MJ, Band SS, Muhammad G, Hasan K, Tiwari P. Towards a blockchain-sdn-based secure architecture for cloud computing in smart industrial iot. Digital Commun Networks 2023;9(2):411–21. doi: https://doi.org/10.1016/j.dcan.2022.11.003.
[23] Islam MM, Nooruddin S, Karray F, Muhammad G. Human activity recognition using tools of convolutional neural networks: A state of the art review, data sets, challenges, and future prospects. Comput Biol Med 149 (106060).
[24] Islam MJ, Rahman A, Kabir S, Karim MR, Acharjee UK, Nasir MK, Band SS, Sookhak M, Wu S. Blockchain-sdn-based energy-aware and distributed secure architecture for iot in smart cities. IEEE Internet of Things J 2022;9(5):3850–64. doi: https://doi.org/10.1109/JIOT.2021.3100797.
[25] Rahman A, Islam MJ, Rahman Z, Reza MM, Anwar A, Mahmud MP, Nasir MK, Noor RM. Distb-condo: Distributed blockchain-based iot-sdn model for smart condominium. IEEE Access 2020;8:209594–609.
[26] Jiang Y, Yin S, Li K, Luo H, Kaynak O. Industrial applications of digital twins. Philos Trans Roy Soc A 2021;379(2207):20200360.
[27] Rahman A, Islam J, Kundu D, Karim R, Rahman Z, Band SS, Sookhak M, Tiwari P, Kumar N. Impacts of blockchain in software-defined internet of things ecosystem with network function virtualization for smart applications: Present perspectives and future directions. Int J Commun Syst 2023:e5429. doi: https://doi.org/10.1002/dac.5429.
[28] Stark R, Fresemann C, Lindow K. Development and operation of digital twins for technical systems and services. CIRP Ann 2019;68(1):129–32.
[29] Khasanov M, Krasnov F. Digital twin of a research organization: Approaches and methods. In: SPE Annual Caspian Technical Conference. SPE; 2019. p. D013S002R004..
[30] Purcell W, Neubauer T. Digital twins in agriculture: A state-of-the-art review. Smart Agric Technol 2023;3:100094.
[31] van der Valk H, Strobel G, Winkelmann S, Hunker J, Tomczyk M. Supply chains in the era of digital twins–a review. Proc Comput Sci 2022;204:156–63.
[32] Umaria MM, Jethava G. Enhancing the data storage capacity in qr code using compression algorithm and achieving security and further data storage capacity improvement using multiplexing. In: in: 2015 International Conference on Computational Intelligence and Communication Networks (CICN). IEEE; 2015. p. 1094–6.
[33] Abas A, Yusof Y, Din R, Azali F, Osman B. Increasing data storage of coloured qr code using compress, multiplexing and multilayered technique. Bull Electr Eng Inform 2020;9(6):2555–61.
[34] Wikipedia contributors, Huffman coding — Wikipedia, the free encyclopedia, [Online; accessed 13-February-2022] (2022). https://en.wikipedia.org/w/index.php?title=Huffman_coding&oldid=1071647715
[35] Bulan O, Blasinski H, Sharma G. Color qr codes: Increased capacity via per-channel data encoding and interference cancellation. Color and Imaging Conference, Vol. 2011. Society for Imaging Science and Technology; 2011. p. 156–9.
[36] Khan MSI, Rahman A, Debnath T, Karim MR, Nasir MK, Band SS, Mosavi A, Dehzangi I. Accurate brain tumor detection using deep convolutional neural network, Computational and Structural. Biotechnol J 2022;20:4733–45.
[37] Liu S, Fu Z, Yu B. Rich qr codes with three-layer information using hamming code. IEEE Access 2019;7:78640–51.
[38] Taveerad N, Vongpradhip S. Development of color qr code for increasing capacity. In: 2015 11th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS). IEEE; 2015. p. 645–8.
[39] Galiyawala HJ, Pandya KH. To increase data capacity of qr code using multiplexing with color coding: An example of embedding speech signal in qr code. in: 2014 Annual IEEE India Conference (INDICON). IEEE; 2014. p. 1–6.
[40] Querini M, Italiano GF. Color classifiers for 2d color barcodes. In: 2013 Federated conference on computer science and information systems. IEEE; 2013. p. 611–8.
[41] Zheng Z, Yu Y, Chen R, Huang H, Zhao H, Lu X. Localization method based on multi-qr codes for mobile robots. In: 2022 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA). IEEE; 2022. p. 1385–91.
[42] Liu N, Du Y, Xu Y. Qr codes blind deconvolution algorithm based on binary characteristic and l0 norm minimization. Pattern Recogn Lett 2018;111:117–23.
[43] Wikipedia contributors, Hexadecimal — Wikipedia, the free encyclopedia, https://en.wikipedia.org/w/index.php?title=Hexadecimal&oldid=1130974725, [Online; accessed 5-January-2023] (2023).
[44] Querini M, Italiano GF. Reliability and data density in high capacity color barcodes. Comput Sci Inform Syst 2014;11(4):1595–615.
[45] Melgar MEV, Zaghetto A, Macchiavello B, Nascimento AC. Cqr codes: Colored quick-response codes. In: in: 2012 IEEE Second International Conference on Consumer Electronics-Berlin (ICCE-Berlin). IEEE; 2012. p. 321–5.
[46] Gonaboina S, Burra LR, Tumuluru P. Secure qr-pay system with ciphering techniques in mobile devices. Int J Electron Comput Sci Eng 2012;1(4):1905–12.
[47] Rahman A, Montieri A, Kundu D, Karim M, Islam M, Umme S, Nascita A, Pescapé A, et al. On the integration of blockchain and sdn: Overview, applications, and future perspectives. J Netw Syst Manage 2022;30(4):1–44.
[48] Liu S, Fu Z, Yu B. A two-level qr code scheme based on polynomial secret sharing. Multimedia Tools Appl 2019;78(15):21291–308.
[49] Fu Z, Cheng Y, Liu S, Yu B. A new two-level information protection scheme based on visual cryptography and qr code with multiple decryptions. Measurement 2019;141:267–76.
[50] Cheng Y, Fu Z, Yu B, Shen G. A new two-level qr code with visual cryptography scheme. Multimedia Tools and Applications 2018;77(16):20629–49.
[51] Lei JTL, Chuin LS, Ernawan F. An image watermarking based on multi-level authentication for quick response code. In: 2021 International Conference on Software Engineering & Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM). IEEE; 2021. p. 417–22.
[52] Xiong L, Zhong X, Xiong NN, Liu RW. Qr-3s: A high payload qr code secret sharing system for industrial internet of things in 6g networks. IEEE Trans Industr Inf 2020;17(10):7213–22.
[53] Wikipedia contributors, Reed–solomon error correction — Wikipedia, the free encyclopedia, https://en.wikipedia.org/w/index.php?title=Reed%E2%80%93Solomon_error_correction&oldid=1120074370, [Online; accessed 21-December-2022] (2022).
[54] Rahman A, Chakraborty C, Anwar A, Karim M, Islam M, Kundu D, Rahman Z, Band SS, et al. Sdn–iot empowered intelligent framework for industry 4.0 applications during covid-19 pandemic. Cluster Comput 2022;25(4):2351–68.