

Enhancement of QR Code Capacity by Encrypted Lossless Compression Technology for Verification of Secure E-Document

AMMAR MOHAMMED ALI¹ AND ALAA KADHIM FARHAN², (Member, IEEE)

Computer Science Department, University of Technology, Baghdad 10066, Iraq

Corresponding author: Ammar Mohammed Ali (80128@uotechnology.edu.iq)

ABSTRACT This paper provides a novel method to improve the data storage of a quick response code (QR code) by applying encrypted lossless compression technology. QR codes are used in several domains, particularly when there is a need to transfer various types of text information. A key aspect of this work is to thus propose a new methodology to overcome the weaknesses of the limited size of the traditional QR code, which has long been an important issue in a wide range of areas. The proposed algorithm incorporates a clear and simple plan for overcoming this difficulty by inserting confidential information into a QR code message. The QR code is updated through the addition of levels that help to share secure messages of various sizes and to authenticate documents for verification and validation. In this work, the newly proposed QR code does not reconstruct the configuration or structure of the QR code. Rather, it provides better security because it relies on the features of the Huffman compression algorithm to reduce the size of the input data and the principles of encryption through the XOR function, which is done through a variable encryption key. The experimental results show the superiority of our method over the previous methods. The scope of this endeavour is thus wide, and there is potential for the encoding of different types of data with a high compression rate in the near future.

INDEX TERMS 1D barcode, quick response code (QR code), Huffman coding, security.

I. INTRODUCTION

The Quick Response code, which can be abbreviated as “QR code”, is used to access and read information through the easy use of 2D barcodes. The QR code has been the subject of many systematic investigations regarding how information is arranged and stored by organizing QR codes in a 2D matrix, along with the columns and rows of this matrix. The matrix represents a place to store data with elements that are visible in the black and white of QR codes, as shown in Figure 1. QR codes are used in domains that involve the transfer of text information, and these include mail messages, phone numbers, hyperlinks or other text files. This is done by capturing the image of the QR code, which is then interpreted by a QR code reader or smartphone applications that are prepared for this purpose. The QR code also contains different patterns: search patterns, alignment patterns, timing patterns, and other types, such as formatting information and time slots, along

with other variables. These make the QR code more susceptible to decryption and detection, thus allowing QR codes to be used in an easy and effective way [1]–[4].

For more than a century, scientists have been interested in data analysis and protection. This pursuit has thus been instrumental in our understanding towards finding an efficient and inexpensive method that does not require a high level of training to use it. For this reason, the QR code was created to represent and protect data based on computer methods. Indeed, the use of web applications and electronic document integrity can be achieved and verified while maintaining user privacy by using improved QR code algorithms, which appear in multiple fields and sectors, including digital signatures, data integrity, security, authenticity, security protocols and data transfer [5]–[7].

From the introduction above, it can be noted that the QR code is an important concept in the study of document authentication through the use of two-dimensional barcodes (QR codes) for document protection and readability. This paper thus proposes a clear and simple plan that can overcome the

The associate editor coordinating the review of this manuscript and approving it for publication was Chi-Yuan Chen¹.



FIGURE 1. Illustrated the arrangement of the information in a 2D matrix visible as black and white.

current difficulty in this aspect through the use of algorithms that can insert confidential information and messages into a quick response code. To design an essential and effective QR code application, the main challenge faced by many researchers is how to achieve readability and confidential sharing on QR modules directly. It must be noted that this work does not reconstruct the configuration or structure of the QR code but rather investigates the challenges related to that structure. The main objective of this paper can be summarized in the following sections. First, it addresses the handling of an authentication system through the use of reliable performance measurements by adding functions that increase security. Despite its long clinical success, the QR code has a number of problems when utilized. As such, there is an urgent need to address these problems. This article presents a plan to overcome all the difficulties that have plagued the previous methods applied. Integrating significant secret data within a quick response code requires the preservation of its contents while enabling a control of the amount of information involved in that code and maintaining the highest degree of security when integrating the data. The second section addresses the enhancement of the QR code to improve the data storage capacity by implementing lossless compression technology in an encrypted way. The last section of this paper displays the experimental results and discusses them in detail.

II. RELATED WORK

From previous studies, it is possible to record and perceive that in many cases, there is a general issue in recognizing and reading 2D barcodes (quick response codes) in different conditions. For example, the large amount of text required to be converted into a QR code makes it tough for devices to read the QR code due to high noise in the pattern. Indeed, there has been an increasing amount of literature associated with QR codes in recent years. Researchers have proposed ways to change the structure of the QR code by using a grey level to store and disseminate data. The proposed method has helped to increase the storage capacity of data associated with QR codes, despite success in increasing the storage capacity of the quick response code. However, there are several conceptual and methodological weaknesses that expose chances of high error opportunities, and the different brightness conditions caused by the introduction of grey levels are among the most important problems [8]. Another study emphasized the importance of two levels of security and how building information was used in the QR code by improving the information

on privacy levels and lightweight accounts, which affect the rest in turn. Such approaches, however, have failed to address encoded decryption processes because a limitation of this work is the use of Base 64, which is restricted to representing 64 characters [9]. The main purpose of this research effort is to improve the data storage of QR codes. This can increase the employment field of the QR code, mainly for smart cities that must process considerable amounts of data. Despite this need, it has been noticed that researchers have not achieved a clear increase in general accounting expenses due to the use of multicast and even multicolour QR codes in much detail [10]. Numerous studies have attempted to explain how separate messages are encrypted to obtain two-layer QR codes, with both the top and bottom layers encrypted in the left and right display modes to encrypt two required messages. However, some evidence suggests that it is difficult to read the QR code by increasing the complexity of the recognition of images that are enlarged, although research is required to confirm the output findings [11]. An additional method used to store more data in less space than the QR code adopted zip compression of data by replacing a small code word with a different type of character through a secure multiplex [12]. Other researchers focused on increasing and expanding the amount of stored data by using a multicolour QR code, which was used in market cash registers that scanned the QR code for payment [13]. Several attempts have also been made to clearly demonstrate that it is possible to transplant information from three layers into the QR code and Hamming code, where the ability to modify and correct the error in the QR code itself depends on the capability to insert confidential information. However, it has been noted that this approach suffers from several key aspects, such as the difficulty of reading QR codes and accurate detection in the event that multiple QR codes have significant distortions and blockages in images with varying lighting and noise. [14]. The use of a Base 64 algorithm in the encryption and decryption stage had a significant effect on increasing the data storage capacity in a QR code [15]. This 2LQR code has two levels: general and private. The general level can be read by any QR code reader, while the second level requires a special application that includes specific input data. Despite the success of this method, the storage size of the 2LQR code needs to be expanded. The research to date has not been able to determine whether QR codes can carry a large amount of data regardless of the success of this method; the storage size of the 2LQR code must be expanded and improved [16]. A QR code authentication system was achieved by adding built-in authentication data, such as secure message data and encryption signatures. Much uncertainty still exists about the relationship between the authentication procedures, and applying this method must be tested under more difficult conditions (blur an image, poor lighting, etc.) [17].

III. COMPRESSION USED IN THE PROPOSED SYSTEM

Different methods proposed in the classification of this work have suggested using the lossless compression technique,

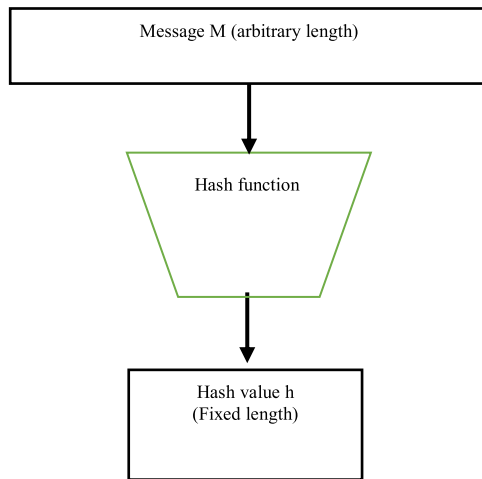


FIGURE 2. Show the integrity of data using hash function.

a technique that does not neglect or lose any bits in the process stages. There are many types of lossless compression techniques, but the most suitable for data integrity in relation to this research is Huffman coding.

A. HUFFMAN CODING

Huffman coding is a lossless data compression algorithm. It considers that quantitative measures would usefully supplement and extend the basic structure of this algorithm depending on two main parts: the first part is concerned with the production of the Huffman tree, while the second part traverses the tree to locate symbols. As such, depending on the basic idea propelling this algorithm, a variable-length code is assigned to enter different characters. Thus, the frequent use of the character and its repetition in the text is immediately associated with the length of the code applied, where the letters that have a high repetition in the text take the smallest symbols and similarly, the fewer the symbols repeated, the longer the code. The complexity used to identify the repeating of each letter is measured by the following law: $O(n \log n)$. All public and private information is encrypted after being compressed as simultaneous sub-pixel blocks in the secret-sharing algorithm of QR [18].

B. INTEGRITY OF DATA USING HASH FUNCTION

To ensure data integrity, several mathematical functions are used to transform a diverse arithmetical input value into another compressed value. The input to that function is of arbitrary length, but the output is of a continually fixed length. The hash function is one of the most important and significant functions needed to accomplish this task. Values returned by a hash function are named hash values; more details on hash functions are available in [19]. References [20] and [21] highlight the use of strict breakdown criteria (SAC) and bit independence criteria (BIC) to deal with changes in the output bit by relying on input bits. The main structure of the hash function is shown in Figure 2 below.



FIGURE 3. Illustrated the arrangement of the information in a 1D barcode visible as black and white.

IV. COMPARISON BETWEEN 1D BARCODE AND QR CODE

Barcodes are delineated through their extensive use in society because of their host characteristics. In relation to the activity in reading and efficiency in the results, the data and information are often stored in one direction within the barcode (see Figure 3).

Subsequently, there is a need for extra enhancements to the barcode to increase the amount of data stored in it and enable the storage of various information and different characters in the output, which can then be printed in a small storage space. Improvements and modifications on the barcodes thus proceeded to respond to these problems by producing two-dimensional barcodes.

The information is stored in the QR code in a vertical and horizontal format, so it retains the information as a two-dimensional code.

The QR code is superior to the barcode in terms of storage, as it has the ability to store various information such as alphabets, numbers and other data with a storage capacity that exceeds the storing capacity of a bar code. A QR code stores up to 7089 characters, while a bar code can hold only 20 digits. The QR code has a small print size, is resistant to dirt and damage and has the ability to be read in different directions.

A. OVERVIEW OF QR CODE STRUCTURE

Two-dimensional barcodes were introduced for the first time in 1990 and have since played a great role in the fields of confidentiality and copyright, in addition to their ability to encode more data than 1D barcodes. Consequently, 2D barcodes have been extensively used in various areas since their conception. There are many types of QR codes, with more than 30 different kinds of 2D barcodes presently obtainable in the marketplace. The QR code has been used in many countries, including Japan. The QR code, which was developed by Denso Wave in 1994, is authorized by the Denso Wave website wherever “QR” indicates “quick response” [22]. Indeed, the QR code collection has different categories of data with high storage, and QR can be read appropriately [23]. The main structure of the QR code comprises the following five parts:

1. The finder pattern used to arrange the determined position, size and angle of the QR code denotes the squares placed on three of the QR corners.
2. The alignment pattern can be used for alteration restoration to classify and ensure that the QR code covers nonlinear distortions located in the alignment pattern by consuming an independent black cell.

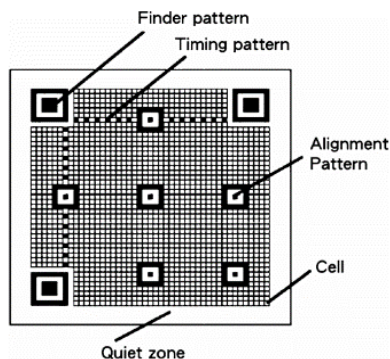


FIGURE 4. Show the main five parts of QR code structure [24].

3. The timing pattern is denoted by a three squares finder pattern. These squares are a collection of black and white cells in horizontal and vertical lines. It can be used to calculate the coordinate of a symbol and can be used to make sure a symbol is stable and can deliver properly without distortion.
4. The quiet zone is the blank region positioned around the structure of the QR code. That region is essential for reading the QR code. Generally, it has four or more cells.
5. The data in the QR code will take binary forms if “0” and “1” are constructed on Reed-Solomon codes, and the code represents the data to be saved (see figure 4).

B. BENEFITS

Symbol Version: The QR code has many ranges of symbol versions that lie between versions (1-40), and each version has a number of modules and a dissimilar module arrangement. The module is denoted by two colours (black and white) that represent the points needed to create the structure of the QR code. The number of modules delimited in each symbol indicates the “module configuration” that starts (21×21 modules) for Version 1 and is active up to (177×177 modules) for Version 40, with each version having four various modules.

C. DATA CAPACITY

The data capacity of the QR code differs depending on the data it carries; it can carry up to 7,089, 4296, or 2953 characters for numeric, alphanumeric, or binary/byte formats, respectively, while a traditional one-dimensional barcode can carry a maximum of only 20 digits. The QR code is disparate from other barcodes, as the QR code is easily read and can detect the data by using both physical scanners and mobile devices.

V. PROPOSED METHOD

The majority of participants agreed with the statement of the proposed system, which showed a QR code that dealt with two basic levels of confidential data sharing. The first level was dependent on lossless data compression with the

encryption of data in different stages of compression. All information was highly encrypted and accurate.

This view was echoed by another informant, as it can be demonstrated that the security dependence of the proposed algorithm in this paper depends on the mechanism needed to deal with the general structure of the algorithm. This would be done by taking advantage of an integration of the Huffman data compression algorithm and encrypting the output by entering it into an Exclusive OR (XOR) function with a variable encryption key of the same length of text.

The majority of participants agreed with the statement that proposes a new way to overcome the problems and weaknesses of the previous QR algorithm.

Together, the results in this paper have provided important insights into storage capacity, which has then been successfully expanded as follows:

1. Converting data, whether text or image, to binary encoding [0, 1].
2. Converting binary notation [0,1] to hexadecimal code, that is, [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F].
3. Dealing with the output of the second step as text that is entering the Huffman coding algorithm.
4. A binary-encoded file is obtained as output from our Huffman coding algorithm. We symbolize it as X.
5. The resulting binary file from step 4 is read in reverse, and the reading result is stored in a new file that is the same length as the original file that is produced in step 4 with the key symbol.

In other words, this paper depends on a stream cipher concept, and a symmetric key cipher has been used. In a stream cipher, each plain-text digit is encrypted one at a time with the identical digit of the key stream to provide a digit of a stream cipher-text, and the combining operation is an exclusive-or (XOR).

Generating keys used in this research paper must be easy and satisfy the most important requirements in the general structure of the proposed system that serves the quick response code. In considering the use of this key, the encryption must be characterized by strength and complete confidentiality with a balance between time and complexity, which is generated with a high dynamic that is provided by our method while maintaining speed. We also know that to benefit from the QR code, the methods used with it must be fast; Table (1) shows the execution time for the key generation of different file sizes.

6. Pronouncing the XOR function between (X) and (Key), we input the result in a new variable (Y). In this step, a roadblock is placed for an attacking party by making the compression method use one-way data compression.
7. Revert to the second step and repeat the operations sequentially until we reach a conclusion.

However, when taken together, these results suggest that there is an association between the XOR operations used

TABLE 1. Calculate the execution time of different.

File size	Time need to create key
139986	6.85E-04 second
224448	0.0011 second
245840	0.0014 second
280168	0.0015 second
317940	0.0016 second
542388	0.0026 second
813330	0.0038 second
906822	0.0043 second
1186990	0.0055 second
1725514	0.0084 second

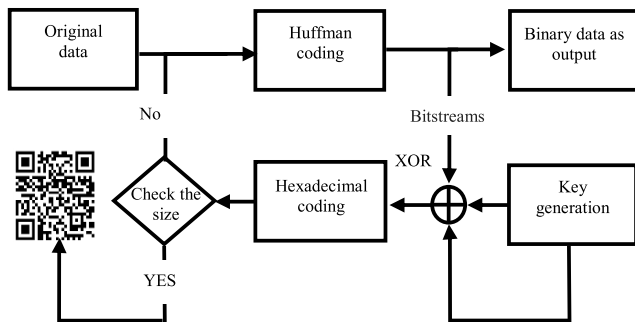


FIGURE 5. Demonstratd the proposed model for data compression.

with the selected secret key in this paper, which would then greatly reduce the computational complexity of encryption and decryption, thereby helping to overcome the aforementioned weaknesses faced by the previous QR algorithm.

The secret level information can then be extracted and decrypted to obtain the information stored in the QR code. The findings from this study make several contributions to the current literature, and the important contributions are summarized to demonstrate a clear approach on how to increase the storage capacity of the QR code (See Figure 5).

VI. EXPERIMENTAL RESULTS AND DISCUSSION

The methods for measuring QR codes have varied somewhat across this research area, and a case-study approach was adopted to obtain further in-depth information towards verifying how to secure electronic documents by creating a digital signature that was prepared by several algorithms to assure highly secure and confidential lossless compression. Consequently, this is finalized in the form of a quick response code (QR) through the suggested approach. The QR code consists of the outputs pertaining to the encrypted compression function that have been attached to improve the properties of the QR function by providing the highest amount of data that can be absorbed in the QR code while ensuring that the data exchange maintains a high level of confidentiality. A variety of methods are used to assess QR codes, and each has its advantages and drawbacks. This work considered the quantitative measures that would usefully supplement and extend the QR code by applying the two methods

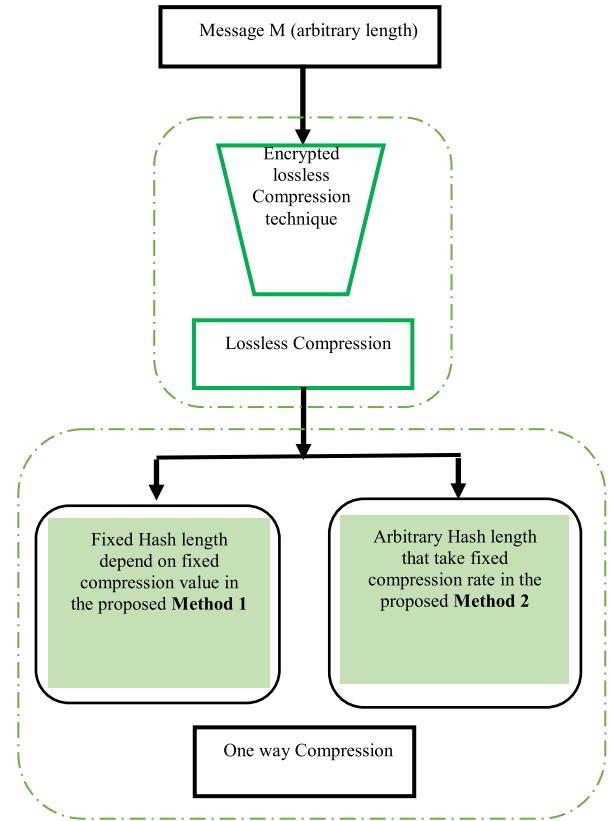


FIGURE 6. Encrypted lossless compression technique into one way compression technique by two methods.

























incorporated into the proposed algorithm, as demonstrated in Figure 4:

- **Method 1:** The input for the first method is the arbitrary length of the message, which can be of any size, while the output is the fixed hash value that is determined in advance. This feature gives a dynamic impression of this method, which means that the expansion of the output length is flexible and dynamic and depends on the value that the user sets in preparing the program to determine the proposed output hash function. For example, the user selects 64, 128, 96, 256,..... or any value that meets the user requirements for this program, as this value is a threshold parameter that controls the level of data compression and access to the final value of the hash.
- **Method 2:** The input of the second method is the arbitrary length of the message, with the output being a fixed ratio of the length of the message, meaning that the output is also flexible and dependent on the ratio in the program, which is set by the user to determine the output suggested hash function. This method has been relied on in files that require control over an amount of a pressure ratio. For example, we want the amount of pressure to be (0.05,001,0.004,0.001)..or any percentage of the file size determined in advance. This percentage shall be

TABLE 2. Apply the methods 1 IN 5 different files and different compression ratio on each file.

File Size in bits 51870 (bit)	Compression rate	25	32	48	64
	Number of iteration	149	147	140	132
	Hash value	'0909F67D F51115F7C DF212'	'12306D82D3509B8 03B21596836C189'	'5BACDB68666652DFA5B94B6 143694ED2FDA533330B6D9AE D'	'2E8AC639A38BE8F7B06EAFE36 02AD09F7C85AA0363FABB06F7 8BE8E2CE31A8BA'
	Hash length	22	30	48	64
	Average of execution time	0.8984	0.9436	0.9209	0.8697
File Size in bits 68530 (bits)	Compression rat	25	32	48	64
	Number of iteration	125	123	118	110
	Hash value	'0455CB89 127F03F92 2474EA88'	'0E70B4A1C0E178 644C3D0E070A5A 1CE'	'3623E55436AA2883D6096C141 B4835E08A2AB61553E236'	'05A38A17C2870F6EAB6FFCD0C 4448BA02E89111859FFB6ABB78 70A1F428E2D0'
	Hash length	25	31	46	63
	Average of execution time	1.0844	1.0802	1.0602	1.0223
File Size in bits 11712 (bit)	Compression rat	25	32	48	64
	Number of iteration	124	121	116	112
	Hash value	'1506A404 923FC4920 2560A8'	'10828E1E5B391B7 7BDDDB139B4F0E2 821'	'09396BAAB06CD99F846A7C1 541F2B10FCCD9B06AAEB4E4 8'	'11A49888D8899745B6EE09C264 3400B6D002C264390776DA2E99 11B11192588'
	Hash length	23	32	47	63
	Average of execution time	0.7512	0.6377	0.6259	0.5656
File Size in bits 68796	Compression rat	25	32	48	64
	Number of iteration	146	144	141	134
	Hash value	'3DF2E4B BD02F749 D3EF'	'5C4663033E69857 750CB3E6063311D'	'37E3BFA55BCFFDAB7342C0D 0B3B56FFCF6A97F71FB'	'686DFFEDA021455D91DC45448 EEE5805A01A777122A23B89BA A28405B7FFB616'
	Hash length	19	30	42	64
	Average of execution time	1.2313	1.1566	1.2446	1.1063
File Size in bits 83846	Compression rat	25	32	48	64
	Number of iteration	193	191	187	180
	Hash value	'0EFDD7C 1868EE2C 307D77EE'	'74243E6F3036EA2 004576C0CF67C24 2E'	'671968C34B3D46C0804060990 602010362BCD2C31698E6'	'4F00409A126AE0C5E97E32CAA 7FF408410817FF2A9A63F4BD18 3AB242C810079'
	Hash length	23	32	46	64
	Average of execution time	1.5576	1.5389	1.5231	1.4880

TABLE 3. Apply the methods 2 IN 5 different files and different compression ratio on each file.

File Size in bits 51870 (bit)	Compression rate	0.01	0.008	0.005	0.004
	No. of iteration	113	121	132	139
	Hash value	'2A668DDD78DF081801861DB127F87FFFE0007FAB01A0F0CD35846F5A228E6DBB76D9C5116BD886B2CC3C160357F8001FFFF87F9236E186006043FEC7AEEC5995'	'24E116CFCE01AF90A12C253D7FA4302FDF7182A0544EE E2988F788CA3BB91502A0C77DFA0612FF5E521A4284FAC039F9B44392'	'2E8AC639A38BE8F7B06EAFE3602AD09F7C85AA0363FABB06F78BE8E2CE31A8BA'	'1116ABC778587E064353A63E1B0F8CB9584C0FC343DC7AAD11'
	Hash length	129	102	64	50
	execution time	0.7401 sec	0.8442	0.9573	1.1325
					
File Size in bits 68530 (bits)	Compression rat	0.01	0.008	0.005	0.004
	No. of iteration	78	91	102	108
	Hash value	'04A3DA5B27629966C4306796946CC0673D2C12500612ECDB918030CE38626799802E0CD32F98B062273CF391183467D32CC1D00667991871CC3006276CDD21802920D2F3980CD8A5A798308D9A651B93696F148'	'0DFD84034698491B4A250C424DB85C9993655CD3524427B17F150652982A3FA3790892B2CEA9B2664E876C908C2914B6248658B0086FEC'	'85EFFD4187DAD47B4A5DDBD2ADA987731BF35FC9F606F93FACFD8CEE195B54BDDBA52DE2B5BE182BFF7A1'	'1C936E2CDD5014A80B201F054C29FC1E6000CF07F286541F009A02A50157668ED927'
	Hash length	167	110	85	68
	execution time	0.8457	0.9855	1.0453	1.0399
					
File Size in bits 11712 (bit)	Compression rat	0.01	0.008	0.005	0.004
	No. of iteration	122	124	127	128
	Hash value	'2C62854A94427703B908A54A8518D'	'1506A404923FC49202560A8'	'2266273CE46644'	'A0A1F8505'
	Hash length	29	23	14	9
	execution time	0.6588	0.7720	0.6350	0.6518
					
File Size in bits 68796	Compression rat	0.01	0.008	0.005	0.004
	No. of iteration	98	105	125	133
	Hash value	'0CC30320909204FD9E9263F380311E097FA13F8830F1072186187A7664E93C4FB17E3C624F1104CCC8223C918F1FA37C8F25C99B978618613823C3047F217FA41E230073F1925FEFC81242413030CC'	'3285594F26701E4C2550F129C5426A3E4AF1421083780000AAB1200B9A583466DA300316D98B069674012355400007B04210A3D49F1590A8E523C2A90C9E03993CA6A853'	'02A044E18ABE33F06657E0E3EC55A357462A805A3201316805518BAB16A8DF1C1FA9983F31F5461C88150'	'0FEEEC94A77DB9111FFD57FA21DBE6954C6552CFB708BFD57FF1113B7DC A526EEFE'
	Hash length	158	136	85	67
	execution time	1.0410	0.9892	1.0445	1.1185
					
	Compression rat	0.004	0.003	0.002	0.001
	No. of iteration	176	181	188	194
	Hash value	'73C9D915F2631105D70D0BAD5C93564505A62058E38D0232D05135649D5AE85875D0446327D44DC9E7'	'2419E7A05643C0DF11C8062FB862E9AD65D1877D1804E23EC0F09A8179E609'	'002E073F6EEAF439C68000058E70BD5DDBF381D00'	'2B8888159DCD40888EA'
	Hash length	84	62	41	19
execution time	1.5719	1.5762	1.5725	1.6200	
					
File Size in bits 83846	Hash value	'73C9D915F2631105D70D0BAD5C93564505A62058E38D0232D05135649D5AE85875D0446327D44DC9E7'	'2419E7A05643C0DF11C8062FB862E9AD65D1877D1804E23EC0F09A8179E609'	'002E073F6EEAF439C68000058E70BD5DDBF381D00'	'2B8888159DCD40888EA'
	Hash length	84	62	41	19
	execution time	1.5719	1.5762	1.5725	1.6200
					

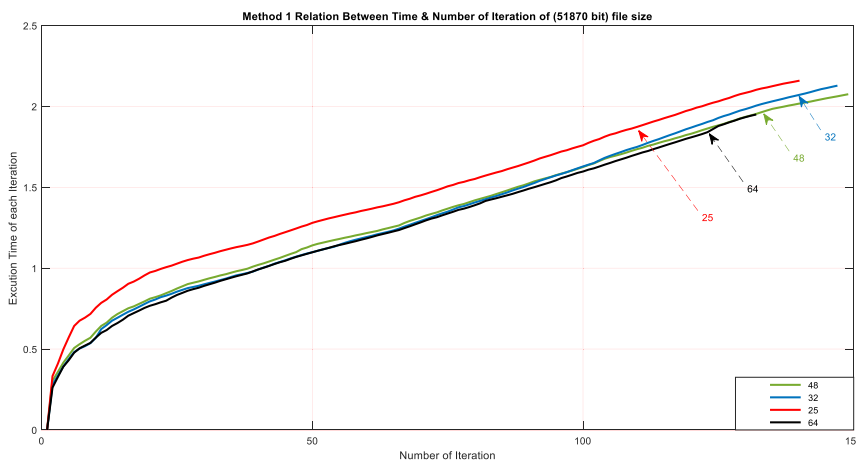


FIGURE 7. Apply method1 on (51870 bit) file size to describe the relation between time & the number of iterations.

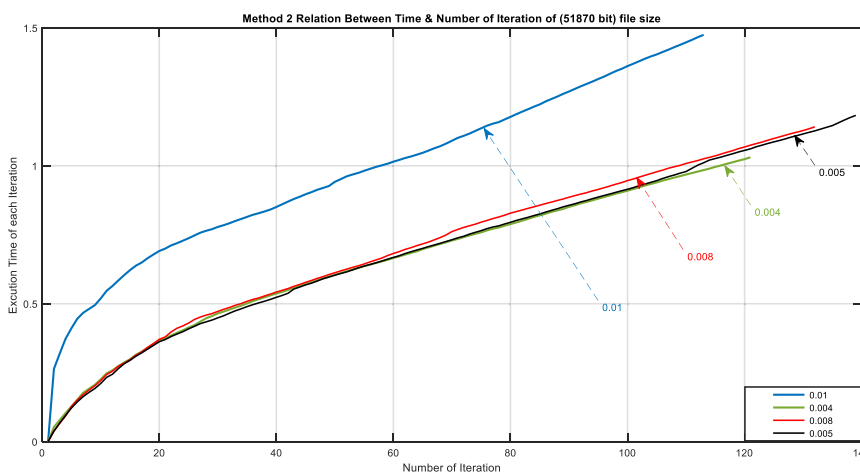


FIGURE 8. Apply method 2 on (51870 bit) file size to describe the relation between time & the number of iterations.

considered the threshold upon which the data entries are compressed into the quick response code.

In other words, this method provides a dynamic compression ratio that stops the predetermined condition where the entered file is calculated, and based on the percentage that has been determined, the number of turns required to reach the hashtag size that meets the user requirements and is proven by the threshold value is determined.

Figure 6 shows the main structure used to convert the lossless compression technique into a one-way compression technique in encrypted ways by two methods (method 1, method 2).

In general, the following observations can be noted:

















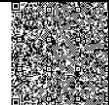
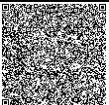











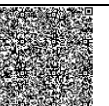





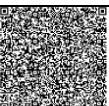












Method 1 and Method 2 in this paper are used to improve the storing ability of the QR code and increase the security level, thus making the security level of both methods better than the traditional QR code.

The QR code resulting in this work has the potential to handle large files successfully. The proposed method is able

to feed data of different sizes to the QR code in an encrypted way. In case we want to store large and important data in a safe manner, we recommend using our method, but if the data size is small and not important and does not require a high level of security, a traditional QR code method is sufficient. The experimental results are recorded in Table 2 and Table 3. The required compression ratio of the data is predetermined to obtain the final value of the hash. It is directly proportional to the number (rounds) used in this program. These rounds are not determined but depend on the amount of compression required, which depends on how important the data are. This means that the time spent in the first method, whose outputs are constant values, is higher than the time spent in the second method and vice versa.

For example, if we take a file with a bit size of 51870 and the first method is applied to a compression threshold value of 48, the resulting hash size (48 digits) required to complete this hash is 118 cycles, and the execution time is 0.9209 seconds. However, if we use the second method on the same file that was used in the first method and with a high

TABLE 4. Test the applicability of reading QR code of eight different files size that created from various compression rate (method1 & method1).

	Method 1			Method 2			Readability for the QR code	
	Compression rate	64	128	256	0.001	0.003		0.005
File Size in bits 139984 (bit)	No. of iteration	184	160	99	191	170	137	Easy Readable For both Method 1&2
	Writing of Printed QR code							
	Hash length	63	126	250	33	104	174	
	execution time	1.6168	1.5114	1.2595	1.7918	1.6121	1.5561	
	No. of iteration	143	124	100	139	101	78	
File Size in bits 317944 (bit)								Easy Readable For both Method 1&2
	Hash length	64	121	239	78	238	374	
	execution time	2.7586	2.6978	2.5531	2.7704	2.7329	2.5718	
File Size in bits 245840 (bit)	No. of iteration	157	137	88	157	112	74	Easy Readable For both Method 1&2
								
	Hash length	61	127	241	61	182	301	
File Size in bits 1186992 (bit)								Easy Readable For both Method 1&2
	Hash length	61	122	252	296	885	1442	
	execution time	11.7450	11.6468	11.7127	12.2641	11.9077	11.8647	
File Size in bits 813336 (bit)	No. of iteration	170	150	127	131	108	101	Easy Readable For both Method 1&2
								
	Hash length	64	128	244	189	595	862	
File Size in bits 906824 (bit)	execution time	5.4729	5.5216	5.3534	6.0332	5.5024	5.3982	Easy Readable For both Method 1&2
	No. of iteration	159	142	112	115	87	68	
								
File Size in bits 542384 (bit)	Hash length	63	128	256	215	549	1125	Easy Readable For both Method 1&2
	execution time	5.5330	5.4342	5.2856	5.7359	5.6003	5.4695	
	No. of iteration	183	162	125	159	109	95	
File Size in bits 1725512 (bit)								Easy Readable For both Method 1&2
	Hash length	63	126	255	135	406	628	
	execution time	4.4087	4.2757	4.1010	4.6208	4.3773	4.3752	
File Size in bits 1725512 (bit)	No. of iteration	191	169	146	124	93	79	Easy Readable For both Method 1&2
								
	Hash length	64	127	250	429	1277	1892	
	execution time	12.9833	12.8069	12.8280	11.9981	11.8185	12.5335	

compression threshold value of 0.004, the length of the hash-tag is 50 and needs 108 cycles to complete this hash, and the implementation time is 1.1325 seconds. The first method

was used on the same file and extracted the value of the following hash length of 64 by adopting the compression ratio 64:

```
'2E8AC639A38BE8F7B06EAFE3602AD09F7C85AA036
3FABB06F78BE8E2CE31A8BA'
```

The number of cycles was 132, and the second method was applied on the same file.

The same hash value of the same length, which is 64, is derived by the compression ratio and a ratio of 0.005 and 132 rounds; more details can be found in Table 2 and Table 3. Both methods have dynamic compression ratios.

For the first method, the data can be converted to any length and stored in the QR code. The required compression ratio of the data volume can also be determined in Method 2. This method has been used and applied effectively and successfully to protect and verify electronic documents. The QR code this paper is a digital signature used to protect and verify confidential and important documents. The data used in these two tables are binary. The time required for each round, which is directly related to the number of rounds, as shown in Figure 7 and Figure 8, which describes the relationship between the number of repetitions and the execution time required for each iteration, and applies Method 1 on a 51870 bit file size with four different compression values (25, 32, 48, 64) that represent the levels of output hash. Then, method 2 is applied on a 51870 bit file size with four different compression rates (0.01, 0.008, 0.005, 0.004) that represent the levels of output hash from the original length file.

A summary of the test applicability of reading QR codes of eight different file sizes (139984, 245840, 317944, 542384, 813336, 906824, 1186992, 1725512) created from various compression rates (Method 1 & Method 2) is provided in Table 4. The binary file used in this table depends on the text file. For method 1, the threshold value (64, 128, 256) was applied. For method 2, the threshold value was 0.001, 0.003, and 0.005. then output of data is printed as QR code. All these QR codes are easy to read for both Methods 1 and Methods 2.

It can be seen from experimental tests that if files are not processed using our suggested method, it is very difficult to convert large files such as (139984, 245840, 317944, 542384, 813336, 906824, 1186992, and 1725512 bits) directly to the QR code, and if they are converted, You will not be able to read it. You can find details of more files with hash values in the supplemental files of this paper.

Both methods use a robust and flexible way to formulate a novel and secure one-way compression technique to help create a new QR code for the verification of data integrity. It is easy to recognize two levels; the first level of each method is a public level, which can be decrypted and distinguished by any standard QR reader, while the second level is a private level and can be read only with the help of a compression algorithm that is used with a secret key to help the encryption process. The experimental results in Table 2 and Table 3 show that documents can be verified using this algorithm with a high accuracy rate.

VII. CONCLUSION

This project was undertaken to design a novel method for evaluating the information of electronic documents using a QR code, especially concerning confidentiality for the rapid detection of information and documents at high speed. The aim of the present research was to examine the technique of concealing confidential information.

Considering the objectives procured from this work, the main intention was to create a QR code based on encrypted and compressed data. The improvement in this work thus encourages an expansion of the number of secret words to be exchanged and transmitted. It can be noted through the scheme of the proposed algorithm that the new design can protect information at several levels based on the QR code and Huffman code, and this design can be used easily and more efficiently in practical applications. The proposed algorithm was discussed in the schemes of several aspects, such as secrecy, durability, complexity, and storage capacity. As such, we have resorted to methods that provide a higher level of security than others, as it has become possible to accommodate a large number of words that could not be absorbed by the QR code in the normal case. A number of known attacks can be resisted through the building of this new QR code model that satisfies security requirements while maintaining the speed features unique to the QR code.

REFERENCES

- [1] S. Liu, Z. Fu, and B. Yu, "A two-level QR code scheme based on polynomial secret sharing," *Multimed Tools Appl.*, vol. 78, no. 15, pp. 21291–21308, Aug. 2019.
- [2] X. Zhang, H. Luo, J. Peng, J. Fan, and L. Chen, "Fast QR code detection," in *Proc. Int. Conf. Frontiers Adv. Data Sci. (FADS)*, Oct. 2017, pp. 151–154.
- [3] R. Kr and B. M. Sagar, "Quick response code for fast detection and recognition of information," *Int. J. Emerg. Technol. Adv. Eng.*, vol. 4, no. 2, pp. 271–275, 2014.
- [4] L. Belussi and N. Hirata, "Fast QR code detection in arbitrarily acquired images," in *Proc. 24th SIBGRAPI Conf. Graph., Patterns Images*, Aug. 2011, pp. 281–288.
- [5] S. Goyal, S. Yadav, and M. Mathuria, "Exploring concept of QR code and its benefits in digital education system," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Sep. 2016, pp. 1141–1147.
- [6] A. F. Kadhim and H. I. Mhaibes, "A new initial authentication scheme for kerberos 5 based on biometric data and virtual password," in *Proc. Int. Conf. Adv. Sci. Eng. (ICOASE)*, Oct. 2018, pp. 280–285.
- [7] S. V. Uttarwar and D. B. A. M., "Two-level QR code for secured message sharing and document authentication," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 6, no. 6, pp. 508–511, Jun. 2017.
- [8] A. H. Tank, M. M. Unde, B. J. Patel, and P. Raskar, "Storage and transmission of information using grey level QR (quick-response) code structure," in *Proc. Conf. Adv. Signal Process. (CASP)*, Jun. 2016, pp. 402–405.
- [9] Z. Fu, Y. Cheng, S. Liu, and B. Yu, "A new two-level information protection scheme based on visual cryptography and QR code with multiple decryptions," *Measurement*, vol. 141, pp. 267–276, Jul. 2019.
- [10] Y. Cheng, Z. Fu, B. Yu, and G. Shen, "A new two-level QR code with visual cryptography scheme," *Multimed Tools Appl.*, vol. 77, no. 16, pp. 20629–20649, Aug. 2018, doi: [10.1007/s11042-017-5465-4](https://doi.org/10.1007/s11042-017-5465-4).
- [11] T. Yuan, Y. Wang, K. Xu, R. R. Martin, and S.-M. Hu, "Two-layer QR codes," *IEEE Trans. Image Process.*, vol. 28, no. 9, pp. 4413–4428, Sep. 2019.
- [12] M. M. Umariya and G. Jethava, "Enhancing the data storage capacity in QR code using compression algorithm and achieving security and further data storage capacity improvement using multiplexing," in *Proc. Int. Conf. Comput. Intell. Commun. Netw. (CICN)*, Dec. 2015, pp. 10–12.

- [13] M. Arora, C. Kumar, and A. K. Verma, "Increase capacity of QR code using compression technique," in *Proc. 3rd Int. Conf. Workshops Recent Adv. Innov. Eng. (ICRAIE)*, Nov. 2018, pp. 1–5.
- [14] S. Liu, Z. Fu, and B. Yu, "Rich QR codes with three-layer information using Hamming code," *IEEE Access*, vol. 7, pp. 78640–78651, 2019.
- [15] A. Abas, Y. Yusof, and F. K. Ahmad, "Expanding the data capacity of QR codes using multiple compression algorithms and base64 encode/decode," *J. Telecommun., Electron. Comput. Eng.*, vol. 9, no. 2, pp. 41–47, 2017.
- [16] I. Tkachenko, W. Puech, C. Destruel, O. Strauss, J.-M. Gaudin, and C. Guichard, "Two-level QR code for private message sharing and document authentication," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 3, pp. 571–583, Mar. 2016.
- [17] C. Chen, "QR code authentication with embedded message authentication code," *Mobile Netw. Appl.*, vol. 22, no. 3, pp. 383–394, Jun. 2017.
- [18] D. Huffman, "A method for the construction of minimum-redundancy codes," *Proc. IRE*, vol. 40, no. 9, pp. 1098–1101, Sep. 1952.
- [19] A. K. Farhan and M. A. A. Ali, "Database protection system depend on modified hash function," in *Proc. Conf. Cihan Univ.-Erbil Commun. Eng. Comput. Sci.*, Mar. 2017, pp. 1–101.
- [20] A. F. Webster and S. E. Tavares, "On the design of S-boxes," in *Proc. Conf. Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 1985.
- [21] A. K. Farhan, R. S. Ali, H. Natiq, and N. M. G. Al-Saidi, "A new S-box generation algorithm based on multistability behavior of a plasma perturbation model," *IEEE Access*, vol. 7, pp. 124914–124924, 2019.
- [22] Denso Wave. *QR Code Features*. Accessed: 2002. [Online]. Available: <http://www.qrcode.com>
- [23] H. Kato and K. T. Tan, "Pervasive 2D barcodes for camera phone applications," *IEEE Pervas. Comput.*, vol. 6, no. 4, pp. 76–85, Oct. 2007.
- [24] B. Furht, Ed., *Handbook of Augmented Reality*. Tampa, FL, USA: Springer, 2011, doi: 10.1007/978-1-4614-0064-6.



AMMAR MOHAMMED ALI received the B.S. degree in computer science from the University of Technology, Baghdad, and the M.S. degree in computer science (computer programming) from Harbin Engineering University, China, in 2012. He is currently pursuing the Ph.D. (data security) degree in computer science with the University of Technology. His research interests include privacy, security, biometric techniques, image processing, and pattern recognition applications.



ALAA KADHIM FARHAN (Member, IEEE) received the bachelor's degree in computer science and the M.Sc. degree in information security from the Department of Computer Science, University of Technology, Baghdad, in 2003 and 2005, respectively, and the Ph.D. degree in information security from the University of Technology, in 2009. He is an Assistance Professor with the Department of Computer Science, University of Technology. In 2005, he joined the Department of Computer Science, University of Technology, as an Academic Staff Member. He has been the author of numerous technical articles, since 2008. His research interests include cryptography, programming languages, chaos theory, and cloud computing.

• • •