

Received August 27, 2019, accepted September 23, 2019, date of publication September 25, 2019, date of current version October 16, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2943754

# Cellular-Automated Protocol to Safeguard Confidentiality of QR Codes

**ABDULLAH M. ILIYASU<sup>1</sup>**, (Member, IEEE)

<sup>1</sup>Electrical Engineering Department, College of Engineering, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia

School of Computing, Tokyo Institute of Technology, Yokohama 226-8502, Japan

School of Computer Science and Technology, Changchun University of Science and Technology, Changchun 130022, China

e-mail: a.iliyasu@psau.edu.sa

This study was sponsored by the Prince Sattam Bin Abdulaziz University, Saudi Arabia via the Deanship for Scientific Research funding for the Advanced Computational Intelligence and Intelligent Systems Engineering (ACIISE) Research Group under Project 2019/01/9862.

**ABSTRACT** Cellular automata (CA) is a veritable tool that provides useful insights into the intricate composition of physical systems. By exploiting this as well as the ability of CA to utilise local interactions of individual CA cells to function as a receptacle for information processing, we propose a CA-based security protocol (CASP) to safeguard the integrity of QR codes. First, undertaking an extensive and conscientious study of the composition of QR codes, we propose delineating an encrypt-able area ( $E_A$ ) that excludes key areas needed to retain physical appearance and properties of an innocuous QR Code. Further, we adduce a zoning structure that demarcates the  $E_A$  into seven zones. Our analysis shows that careful adulteration of contents of at least two zones are enough to produce encrypted versions of the QR codes. Second, each zone is partitioned into  $m$  tiles, each a  $3 \times 3$  sub-block and then local interactions emanating from the occupancy (or strength) of the tiles are used to determine the composition of first- and second-tier rulesets. Third, to steer the evolution of the QR Codes, we propose the use of zone and cell-wise dextral boundary conditions (DBC) that combine a troika of cells permeating contents of a tile at state  $t$  to determine the left-most cell entry at state  $t+1$  of its evolution. Further, we impose a pixel-wise constraint that ensures that each encrypted tile has a discordance that is no less than the in-built error correction tolerance of the code. This property guarantees adequate scrambling of the QR code to mitigate unauthorised access to it and the information it conceals. Meanwhile, considering the properties (balanced, linear and reversible) and nature of our rulesets, the proposed CASP protocol recovers QR codes that are seamlessly scannable as conduits leading authorized users to confidential information. We validated our protocol by implementing both the encryption and recovery procedures on different versions of QR Codes and our results suggest that, on average, modifications up to a minimum of 2 to 5% of any two zones of the delineated  $E_A$  were enough to securely encrypt Versions 2, 3, and 4 QR Codes that were reported in the experiments; thus, rendering them unscannable and the information they conceal inaccessible. Similarly, the recovery process yields an average 97% fidelity between the original and recovered QR Codes, which is enough to restore full functionality of the codes without need for any special hardware or add-ons. Moreover, rulesets employed in our protocol are profuse, dynamic and complex, which are key properties for CA-based cryptography. This demonstrates the efficacy of our protocol as a tool to confine the transitory role of QR Codes to authorised users.

**INDEX TERMS** Cellular automata, confidentiality, encryption, information security, QR codes, reversible computing.

## I. INTRODUCTION

### A. BACKGROUND

The massive amount of information in the public domain combined with the ever-increasing sophistication aimed at

The associate editor coordinating the review of this manuscript and approving it for publication was Orazio Gambino<sup>2</sup>.

illicit, malicious, and/or unauthorised access to such information makes efforts to safeguard critical information a paramount priority concern for individuals, organisations, governments, etc. Among others, it is important to protect personal information, such as financial, health, etc. records as well as critical government infrastructure from misuse, tampering, and pernicious attacks.

The CIA triad consisting of confidentiality, integrity and availability form the fundamental principles underpinning information security [1]. These three criteria provide lenses to assess information security and guide security policies at various levels.

Succinctly put, confidentiality is a set of rules that limit access to information; integrity is the assurance that the information is trustworthy and accurate; and availability, which is more hardware-focused, is aimed at guaranteeing reliable access to information [2]. When priority is assigned to confining access to the information to selected users, then confidentiality takes centre stage. Therefore, from users' perspectives, confidentiality is focused on authenticating the credentials of users to ensure they are authorized to access such content. Roughly speaking, confidentiality equates to privacy since access is granted to those with verified credentials. Meanwhile, confidentiality focuses on limiting access to the underlying information itself [3]; hence, measures undertaken to ensure confidentiality are designed to prevent sensitive data from reaching people without such privileges [2]. The science used to enforce data confidentiality is known as *cryptology*, which involves encryption and decryption of the data. The main aspects of such information security paradigms include privacy, data integrity, authentication, and non-repudiation. This requires encrypting the information, a process that entails transformation of a message (plaintext) into ciphertext; and the opposite process (i.e. decryption) to recover the unenciphered or pristine version of the message. These two complementary operations satisfy the demand of privacy, which is at the core of safeguarding confidentiality.

## B. LITERATURE REVIEW AND CONTRIBUTIONS

A Quick Response (or QR) code is a two-dimensional (2D) array of bits that are visually encoded in a binary grid. Since their introduction almost two decades ago, QR codes have gained widespread popularity which is attributed to its attractive features in terms of high storage capacity, minimal space requirements, support for different data formats, error correction, etc., [4]–[6]. A detailed outline of QR codes can be digested via [4], whereas a more succinct overview can be accessed via [6].

As a medium to access transitory media, there are concerns related to QR code transmission. Moreover, their design as scan-based tools means the security of an entire network or organization can be jeopardized by scanning an unsafe QR code. Depending on use, form, etc. approaches related to QR code security (or QRS for short) can be viewed from three perspectives. On one hand, as noted earlier, because of the nature of their information transmission, QR codes could serve as source of security threats to other content that is ensconced via its access. On the other hand, the physical use of the QR codes in different security frameworks could be considered. This can be further divided into two approaches: first, revolves around the susceptibility to use the QR codes as attack attacks aimed at repudiating the confidentiality of the content it is designed to enmesh. Second, in instances

where the QR codes are used to protect other media, security vulnerabilities associated with their physical use could entail the threats arising from the use of QR codes to safeguard other media (images, text, entire networks, etc.) from illicit tampering. While there is a dearth of studies considering various uses of the QR codes enumerated above, it is pertinent to provide a copious, yet succinct overview on different QRS frameworks. Doing this provides an outlet to clearly enunciate the scope as well as aspects deemed outside the purview of the proposed study. Furthermore, our effort to exploit the dynamic potency of cellular automata in safeguarding the integrity of QR codes entails providing rudimentary background both the CA and Qr codes, which stretches the length of the literature review.

Whereas the primary concern of this study is physical QR code security, for omniety, we note that, as construed earlier, literature on transmission based QRS infer the use of QR codes as attack vectors [7]. In other words, in these techniques the QR codes are considered as malicious threats that contain links embedded with malware [8]. QRshing (which is a portmanteau from combining 'QR' in QR codes with 'phishing') is a term used to describe QR code semantic attacks, i.e. phishing, that masquerade as trustworthy entities with aim of appropriating illicit access to sensitive data. Similarly, in QR code typosquatting (another portmanteau combining 'typo' with 'squatting') QR codes are used in breaching security by redirecting users to fake weblinks (or URLs). In most cases, the genuine and fake weblinks differ in slight (deliberately created) misspellings in their web addresses [9].

Reference [9] presented a non-experimental survey-based approach to assess challenges related to different types of attacks. Meanwhile, in [10], a social engineering perspective to QRS was adopted to assess the vulnerabilities associated with the (use and) transmission of QR codes as well as different attack vectors particularly for smartphone applications and mobile usage of QR codes.

Meanwhile, the so-called signed QR codes (SQR) [4], [11] are designed to mitigate the effects arising from transmission based QR security threats. They are focused on allowing the reader to verify the source of the SQR before any action is performed [11]. Therefore, if the verified source is trusted, the user proceeds to open the link or perform any other action initiated via the code. However, standard SQR protocols require more modification than encryption methods because the code must contain the message, signature, and a way to identify the sender [11].

A general overview on QRS is presented in [12], while a review tailored for transmission based QRS can be digested via [11].

Reverting to the physical deployment of QR codes to protect other media, we elucidated that such QRS schemes can be further construed it in terms of whether the QR code is the cover image or it is being used as a watermark for ownership verification. As is the norm in the latter types of information security techniques, visible QRS methods are ones in which the QR code is visibly recognisable. Conversely, the QRS is

invisible when the presence of the QR code is not readily discernible.

In [13], a probabilistic sharing visual cryptography scheme (VCS) was combined with QR codes to enlarge the allowable maximum size of a secret image. Authors of that study claim that the error-correction capabilities of the shares are preserved in their technique.

Meanwhile, [14] used a combination of discrete wavelet transform (DWT) and discrete cosine transform (DCT) to embed QR codes as watermarks for safeguarding other images. The authors assert that their approach facilitated the recovery of watermark information even after its likely exposure to adversarial manipulation.

In [15], a stego technique was proposed to embed secret information into QR codes. At the likely detriment of safeguarding the QR code itself, the focus in that study was on safe recovery of the hidden information.

Elsewhere, in [16] presented an infrared visible watermark scheme to hide a QR code by concealing its presence with another image. This involves suffusing the QR code information onto an image to form an explicit graphic code, which uses different intensities of implicit QR codes and serial numbers to watermark the QR code. The result is a masked image in which only the finder patterns of the QR code are visible. A downside of this method is that the recovery or reading of the code imposes need for twin readers consisting of an infrared detector to extract the implicit information and a general QR code reader to interpret the explicit graphic QR code. This complicates an otherwise easy task of reading the QR code.

In terms of the target audience for the transitory media, majority of the published QRS literature could be classified as being *user-* or *owner-*focused [17]–[21]. Whereas the primary interest in user-focused QSR schemes is to protect users from accessing unsafe content, owner-focused schemes are designed to safeguard the confidentiality of the information concealed via the QR codes. It is mainly aimed at keeping information secret from all but those authorised to access it.

Cellular automata (CA) are highly parallel and discrete dynamical systems whose behaviour is completely dictated by local interactions [22]. CA consists of a collection of cells arranged in an  $N$ -dimensional lattice, such that the state of each cell evolves temporally according to a set of defined rules that depend on the cell's neighbours (aka local interactions). In other words, the state of a cell at an instant  $t + 1$  but also depends only on its current state (i.e. at instant  $t$ ) and the states of cells in its predetermined neighbourhood. In the manner described, all cells are updated synchronously and, so, the state of the entire lattice advances in discrete time steps. Notwithstanding its apparent simplicity, CA have been proven to exhibit very complex dynamical behavior [23].

An image can be interpreted as a 2-dimensional CA where each cell represents a pixel in the image and the intensity of each pixel is represented by the state of that cell [8]. Structurally, a CA is made up of an unknown binary grid of cells from different spatial and temporal scales [24]. Exploiting

this close affinity between its structure and the grid representation used to encode images, CA has found widespread applicability in many areas of image processing. Notably, CA has been extensively applied in edge detection [25]–[32], and it has shown potential for applications in image enhancement [33], pattern recognition [34], [35], image compression [36], [37], segmentation [38], [39], noise filtering and removal [40]–[42] as well as other uses in general image operations such as translation, zooming and thinning [43]–[46]. In areas more related to our study, CA has been applied in image security protocols such as cryptography [47], [48], watermarking [49]–[51] and encryption [52].

As adduced in [9], the first recorded literature on CA-based block cryptography is credited to Gutowitz, in [47], wherein permutative rules were used to determine a preimage for the diffusion phase of a cipher. Next, Seredynski and Bouvry [22] investigated the use of second-order CA to model avalanche properties via local interactions. In their contribution, Szaban and Seredynski [54] constructed ciphers based on a selection of rulesets that exhibited nonlinearity and autocorrelation. This resulted in six rulesets that were later assessed for bijectivity over a considerable array of lengths.

As highlighted earlier, and detailed in various QRS literature cited above, to the best of our knowledge (and within limits of our resources) none of the previous studies quite meets our intended use of CA in user (or destination) focused QRS with the QR code as cover image. The closest attempt was Kapsalis' study in [9], where attack mechanisms aimed at violating the integrity of QR codes were assessed. Unlike our study, the aim of that study was to retrieve an alternated content that was decoded by repainting the QR code. In another departure from our cogitation, the author (of [9]) adopted an empirical, survey-based approach to ascertain the dynamics involved in "tricking users to scan potentially malicious QR codes." Apparently, the similitude between [9] and our proposed study stops at the cell-focused manipulations. Therefore, applications, objectives and validation of the two studies differ. Moreover, as outlined, unlike the rigorous use of image-based quality metrics planned to validate our proposed CASP, [9] adopted a social engineering perspective to ascertain users' level of security awareness through online surveys. In fact, in terms of validation, besides being thorough in terms of its review of QRS techniques, the study in [12], which utilised standard image-based metrics, is more like our study. However, their study was tailored for generating coloured (cyan, magenta and yellow) QR Codes. Other QRS schemes, such as [16] used recognition rate to assess the performance of the method, while in [13] the probabilistic sharing approach that is popular in visual cryptography schemes (VCS) was employed in the validation.

Finally, we note that another distinguishing feature between our proposed CA-based QSR protocol and other QSR schemes, is that unlike their use of standard symmetric-key encryption (SKE)-based algorithms (where separate plaintext, key, and cover content are required), in our CASP the QR code percolates as both the plaintext and

cover content. Additionally, during the recovery process, information from the encrypted QR code is used as ciphertext to decode, read, and access the content enshrouded via the QR code.

To reiterate, as already elucidated, the design of our CASP protocol provides a tool to safeguard the confidentiality of the content ensconced using the QR code. We validate the utility of our proposed scheme in terms of standard image and encryption metrics as well as the feasibility to for the QR code to sustain its role as a conduit to access and retrieve details and content of the QR code.

The rest of the paper is outlined as follows. An overview on the cellular automata paradigm is presented in Section II, which is followed by a similar outline on QR codes, its essential features, as well as some rudiments of the QSRs that are presented in Section III. The background in these introductory sections serves as the building blocks of our proposed user-focused CA-based QR code security protocol (CASP) that is presented in Section IV. Finally, we present a framework that assesses the performance of our proposed technique in terms of its core properties of being an image and a security contrivance, a transitory media that is intended to confine access to certain content only to authorised users.

## II. OVERVIEW ON CELLULAR AUTOMATA

Cellular automata (CA) are dynamical systems that can exhibit complex global behaviour from simple local interactions and computations.

In its simplest form, a CA is composed of a grid of *cells*, each applying a *local rule* to its neighbourhood in order to compute its next state. The *CA global state* is the configuration of the states of all cells making the grid at a given instant, and the dynamical behaviour of the CA is determined by updating all cells in parallel in discrete time.

In the traditional definitions of CA, it has three fundamental features as follows [55], [56]:

- Homogeneity – all cell states are updated by the same set of rules
- Parallelism – all cell states are updated simultaneously
- Locality – all interactions take place on a purely local basis and a cell can only communicate with a few other cells [24]

Moreover, [24] elucidated that CA is composed of three parts: a neighbourhood, a local transition rule, and a discrete lattice structure which consists of many cells that are occupied by states from a finite set of discrete values. Similarly, as noted in [55], a fundamental precept of CA is that the local transition function determining the state of each individual cell at a particular time step should be based on the state of those cells in its immediate neighbourhood at the previous time steps. Thus, the rules are strictly local, and each cell becomes an information processing unit integrating the state of the cells around it as well as its own state *in unison* [55]. Consequently, global features emerge from the strictly local

interactions of individual cells each of which is only aware of its immediate environment.

### A. TYPES OF CA

Depending on structure, applications, etc., CAs can be classified into the different categories that are briefly outlined in the sequel.

#### 1) UNIFORM AND HYBRID CA

A uniform CA is one whose cell evolution is governed by the same ruleset throughout [55], [57]. Otherwise, the CA is considered non-uniform or hybrid.

#### 2) LINEAR AND NON-LINEAR CA

If the evolution a CA can be conveyed using exclusive OR (X-OR) logic operations it is considered linear. Otherwise, such a CA is classified as non-linear.

#### 3) COMPLEMENT AND ADDITIVE CA

If the ruleset of a CA involves only complement exclusive OR (i.e. XNOR) operations, then it referred to as complement CA. A CA composed of combinations of XOR and XNOR rules is called an additive CA.

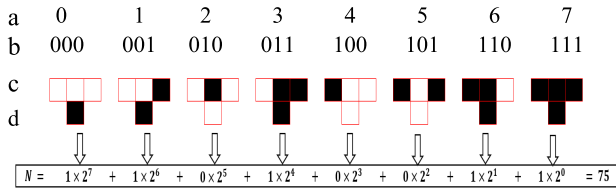
### B. TRANSITION FUNCTIONS AND GRIDS

The study in [24] defines a cell neighbourhood as a set of cells over both space and time that are directly involved in the evolution of the cell. Furthermore, the neighbourhood structure varies depending on the construction of the cell itself, but, as noted earlier, the neighbourhood can include cells from different spatial and temporal scales.

The global state of a CA is the configuration of the states of all cells that constitute the grid at a given moment, and the dynamical behaviour of the CA is determined by making all cells update in parallel and in discrete time steps.

A 2-state 3-neighbourhood CA has  $2^3 = 8$  distinct configurations and  $2^{2^3}$  distinct mappings from all its neighbourhood configurations to the next state, where each mapping represents a non-linear CA rule. This indicates that the configuration 1111111, which produces the decimal value 256, is the maximum state of the 2D CA (described earlier) whose general neighbourhood scenarios in grid and graphical form for 1D CA is presented in Fig. 1.

The first two rows (a and b) are analogues to a truth table where the first row is the address or position of the cell group, i.e. the neighbourhood, and the second row, i.e. the configuration, describes the states of the neighbourhood. These underlying rules are used to steer the evolution of cells in a neighbourhood and ultimately the CA itself, which are known as the transition function of that CA. In other words, a transition function is the set of instructions that determine the new state of a cell and its movement – right or left. A convenient notation to depict the transition state (i.e. state of cell  $i+1$  after transition is dictated by the function specified by the transition rule. The third (i.e. d) row presents a grid



**FIGURE 1. General neighbourhood details of a cell in CA and its transition function: (a) cell address, (b) configuration, (c) grid, and (d) outcome for transition function for rule 75.**

(0 for white and 1 for black) equivalent for the neighbourhood configuration (i.e. row b). Finally, the fourth row indicates the next state of the middle cell (at  $t+1$ ).

Actually, the grid in Fig. 1(c) shows the transition rule 75, where the binary conversion of the decimal number  $N = 75 = 01001011$  (in the bottom row of Fig. 1) is obtained from the generalised decimal to binary method of successive division of  $N$  by 2 until the  $k^{\text{th}}$  iteration, whence the result is 0. Finally, the remainder from the first to  $k^{\text{th}}$  division is read bottom-up yielding a binary sequence (of 8 bits) as presented in (1).

$$B = b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0 \quad (1)$$

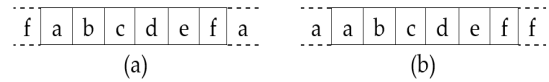
where  $b_k \in \{0, 1\}$  and since (as noted earlier) in CA we are constrained to eight possible entries, then  $k$  varies between 0 to 7. It is trivial that  $B$  has upper and lower configuration given by 00000000 and 11111111 corresponding to decimal values  $N = 0$  and  $N = 256$  respectively.

**C. BOUNDARY CONDITIONS AND REVERSIBLE CA**

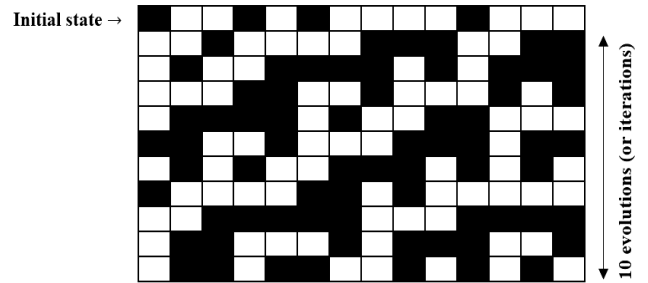
To obtain insights regarding physical systems, the CA structure is simplified with a restriction to local interactions among its cells. Among others, boundary conditions are imposed to support the evolution of a CA based on preassigned dictum known as the ruleset. Boundary conditions are a set of constraints or guidelines that specify actions of the ruleset at a terminal cell. This terminal cell could be to the right or left of the cell. Depending on the nature of a CA, periodic, reflective, null, etc. boundary conditions could be imposed to steer the CA evolution process.

A periodic boundary condition (PBC), aka wrapping, ensures that a finite 1-dimensional array of cells (labelled  $a, b, c, d, e, f$ ) with extreme entries  $a$  (left) and  $f$  (right) will have left and righthand neighbourhoods formed by abutting the entries  $f$  and  $a$  to extend the hitherto available entries ( $a$  to  $f$ ) as shown in Fig. 2(a). In a reflective boundary condition, the same array of cells will have the first and last cells repeated to serve as their own neighbours. This is depicted in Fig. 2(b).

Given an initial or quiescent state  $q = 1001010000100$ , its middle cell at time  $t$  is altered (according interactions determined by the transition function, which includes its own state and states of its two nearest neighbours as well as the configuration of the rule used (i.e. rule 75) to collectively determine its a new state at time  $t + 1$  as presented in the



**FIGURE 2. Periodic and reflective boundary conditions for 1-D CA.**



**FIGURE 3. PBC-evolution of initial state 1001010000100 (first row) and outcomes using rule 75 after ten iterations.**

second row of Fig. 3. The evolution continues in that manner through subsequent iterations as shown in the figure.

A CA is said to have null boundary conditions if both the left and right neighbours (of the leftmost and rightmost terminal cells) are connected to an abutted 0 entry. Further, depending on the application and the connectedness of the terminal cell, other boundary conditions such as fixed, adiabatic, and reflexive are used.

As depicted in Fig. 1, in CA, each cell can exist in one of two possible states (not simultaneously) that are usually denoted by symbols 0 and 1. In forming a CA grid, we may denote states using a graphical representation where white and black fills in each cell. In the parlance adopted in this study, a cell is occupied when it has a fill and the number of such cells denote the *occupancy* of an image or any part thereof.

Based on the definitions in [57], a rule is deemed irreversible if its presence in a rule vector makes the CA irreversible. More generally, it has been suggested that reversible rules for use in image processing should satisfy the following conditions [55], [57]–[59]:

- **Balancy** – a rule is considered balanced if it contains the same number of 1s and 0s in its 8-bit binary representation.
- **Linearity** – it implies all cell states are updated simultaneously, a property that itself implies additivity and homogeneity. Whereas homogeneity was briefly explained earlier, additivity implies a CA composed of a combination of XOR and XNOR rules [59].

Based on Theorem 4 in [57], an unbalanced rule is irreversible, i.e. it cannot be mirrored. Approximately one-quarter of the total 256 rulesets in 2D CA are considered reversible (listed in Fig. 4(a)). Furthermore, as warned in [57], not “every sequence of reversible rules in a rule vector corresponds to a reversible CA.”

Consequently, despite their satisfying the *balancy* criterion, eight (8) rules are deemed irreversible. Moreover, [57]

	15		75	77	83		
23		27	85	86	89		
30		39	90	92	99		
43		45	101	102	105	106	108
51	53	54	57	58		113	114
		60				120	
	135		195	197	198		
141	142		147	149	201	202	204
150	153		154	156	210	212	216
163	165		166	169		225	228
170	172		177	179			232
		180				240	

29	71
46	116
139	209
184	226

(a) (b)

FIGURE 4. Look up table (LUT) for balanced CA rulesets. (a) balanced reversible rules and (b) balanced irreversible rules.

asserts that these balanced yet irreversible rules exhibited absence of cyclic states in its state transition diagram. In total, reversible CA can only be formed from the list of sixty-two (62) reversible rules in Fig. 4(a), while the eight (8) balanced but irreversible rules [57] are presented in Fig. 4(b).

### III. OVERVIEW ON QR CODES

Quick response (or simply QR) codes are two-dimensional (2-D) bar codes that consist of binary level modules used to encode arbitrary text strings. As noted in [4] and [61], QR codes exhibit many attractive features including high capacity data storage, fast, omnidirectional (360 degrees) scanning, minimal space requirement (i.e. small print size), support for different data types (including Japanese, Korean, and Chinese characters), distortion compensation, capability for different amounts of error correction, structured appending, and direct marking.

There are forty (40) versions of QR codes, each with its data capacity. The smallest QR code has  $21 \times 21$  modules (or ‘pixels’ as it is referred to in the language of image processing), and the largest have a  $177 \times 177$  dimension. The sizes are called versions, such that the  $21 \times 21$  is known as version 1,  $25 \times 25$  as version 2, and so on. The largest sized QR code ( $177 \times 177$ ) is known as version 40. Additionally, as well as having error correction ingrained into their build up, QR codes contain redundant data that help in reading the code even when parts of it are unreadable or unavailable. The capacity of a QR code is determined by its version and error correction level. In terms of data mode, a QR code supports encoding numeric, alphanumeric, binary or Kanji data formats. Each mode encodes text as a string of bits (0s and 1s), but each uses a different technique to convert the text into bits. Finally, the encoded data together with the necessary error correction codes, which are needed to facilitate recovery of the codes, are interleaved, masked, and then structured into the final QR code matrix.

Fig. 5 presents a brief overview of the composition of a typical QR code showing the finder (or position), timing, and alignment patterns, while the diagrammatic structure of versions 1, 2 and 40 QR codes are presented in Fig. 6.

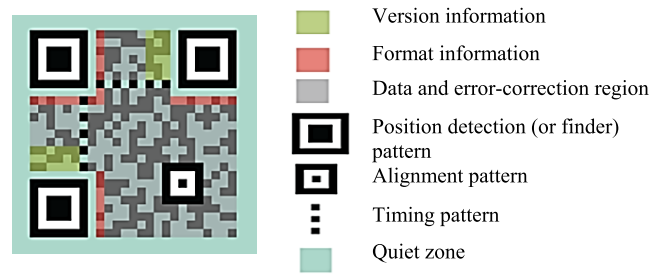


FIGURE 5. Structure of a typical QR code (figure adapted from [4]).

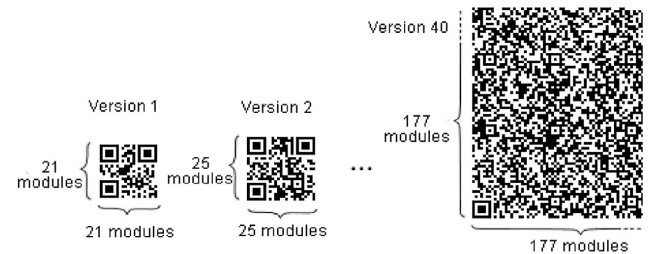


FIGURE 6. Composition of versions 1, 2 and 40 QR codes (figure adapted from [4]).

It has been demonstrated that the readability of QR codes is satisfactorily immune to: missing data and error correction information (up to 30% for H level error correction QR codes); geometric alterations, such as shrinkage, rotation, skewing, etc.; chromatic manipulation, such as variation in contrast and/or hue, blurring, etc.; addition of different types of noise; introduction of foreign impurities, including dirt, dampness, etc.; and tampering with its position pattern.

Further, whereas higher version QR Codes have multiple Alignment Patterns, they all a reference or main Alignment pattern located at the lower right side of the QR Code [62].

TABLE 1. Error correction in QR codes.

QR Code level	Tolerance of correction available (%)
Low (L)	7
Medium (M)	15
Quartile (Q)	25
High (H)	30

Table 1 summarises the error correction tolerance of different versions of QR codes wherefrom we see that four levels of error correction are available. The higher the level, the greater the error correction, but also the larger the QR Code version [4], [62].

The secure QR Code (SQRC) is Denso proprieted QR Code that “has data reading restrictions” [4]. These codes can be used to manage private information. However, as noted in [4], “this function does not guarantee the security of the coded data.”

#### IV. CA-BASED PROTOCOL TO ENHANCE SECURITY OF QR CODES

The first recorded literature on CA-based block cryptography is credited to Gutonitz, in [47], wherein permutative rules were used to determine a preimage for the diffusion phase of a cipher. As outlined in Section II, however, information about the value of the current configuration of a CA and its value at the previous time step, aka transition function, have been established as prerequisites for computing the next state of a 2D CA. In this regard, Sredynski and Bouvry [22] investigated second-order CA based on the avalanche properties of CA (of length  $n = 32$  and  $n = 46$ ) that are equipped with different local rules. In their contribution, Szaban and Sredynski [54] constructed ciphers based on a selection of rulesets that exhibited nonlinearity and autocorrelation. This resulted in six rulesets that were subsequently assessed for bijectivity over a considerable array of lengths.

In terms of access to the hidden message or its recovery, encryption techniques are divided into two categories [58], [63]: symmetric-key and public key. If both sender and receiver use the same key, or it is easy to obtain one from another, then the system is referred to as symmetric-key encryption (SKE) [64]. Conversely, if the sender and receiver use different keys and it is computationally infeasible to determine one from the other, i.e. without knowing some additional information, then the system is referred to as a (asymmetric or) public key encryption (PKE). Symmetric key encryption can be further divided into techniques using block or stream ciphers. A block cipher divides the message into blocks of fixed length and encrypts one block at a time. As suggested by its name, a stream SKE encrypts data one stream or byte at a time. Elsewhere, in [52], Sredynski et al. opine that reversible rules to be used in cryptography should: (1) be many in number, and (2) exhibit complex behaviour.

Based on the foregoing, if classified as a cryptographic protocol, then in terms of structure, the encryption process of our proposed CASP strategy could best be categorised as a CA-based block SKE method. This arises from the CA-based intuition used to divide and modify contents of an image like [64] and [65]. However, while [64] limited the choice of reversible gates to just two rulesets (i.e. Rule 236 and Rule 19), in this study the pool is extended to cover almost a quarter of all rulesets employed in 1D CA, i.e. sixty-two (62) rulesets (that were presented earlier in the look-up-table (LUT) in Figure 4(a)). Furthermore, the choice of which ruleset to use is solely determined by the composition of the original (unencrypted) image. As well as qualifying it as a public key encryption, this makes our proposed protocol dynamic and adaptive. Additionally, while a plane text cipher was added to the initial state of the CA in [64], our proposed CASP protocol depends on the global neighbourhood (i.e. composition) of the original image (i.e. QR code) as well as local interactions between neighbouring cells located within partitions of the image. This *glocal* interaction is combined with our preference to steer the evolution of the CA by in a manner that constricts its growth for content along the

left-to-right direction only. The latter specification results in the *dextral* boundary condition that is an important precept of our proposed CA-based user-focused security protocol.

Finally, whereas [66] notes that most CA-based block SKE methods “have not been subjected to rigorous security analysis”, we validated our proposed strategy in terms of standard statistical, encryption and image-based metrics. Further, we tailor the implementation of our proposed protocol to enhance the integrity of QR codes whose ubiquity in today’s communication is deserving of such efforts. These perspectives are the objectives of our protocol whose detailed outline is presented in the remainder of this section.

#### A. BACKGROUND AND ASSUMPTIONS SUPPORTING PROPOSED CASP PROTOCOL

We start by clarifying that, our notion of an image entails a 2-dimensional grid that shows the position of each pixel, while its amplitude is expressed as its greyscale (or, more accurately, a binary) value. Meanwhile, as in 2D CA, a nine-cell neighbourhood is used for information processing. In this manner, our intuition is that 2D CA structure could be used to describe the image whereas the manipulations to transform the content of the image will be guided by insights employed in 1D CA. This strategy is tenable since operations on 2D CA can be effectively built from 1D CA. Therefore, we adopt a convention where the *encrypt-able* content of a QR code is composed of  $m$  sub-blocks or *tiles* each comprising of nine neighbouring modules (i.e.  $3 \times 3$  pixels (or cells) dimension). Consequently, the *encrypted* state of a pixel (or cell) in a QR code is influenced by composition of its tile  $T_m$ , which is determined by its own state and that of eight pixels in its  $3 \times 3$  neighbourhood. Furthermore, it is the composition of  $T_m$  that dictates the ruleset (composed of 1D CA interactions) employed to steer the evolution of an unenciphered QR code tile to its encrypted state. Finally, since the encrypt-able space is composed of  $m$  tiles, then juxtaposing these encrypted tiles – side-by-side – is adequate to produce the encrypted QR code. These are important precepts on which the foundation of our proposed protocol is built whose details are enunciated in the next subsections.

##### 1) ENCRYPT-ABLE SPACE, CELL OCCUPANCY, AND TILE STRENGTH

In traditional encrypted or secured QR codes (SQR), the resulting QR code is considered as an attack vector; therefore, attention is given to restricting the ability to scan the QR codes to certain reading devices as well as composition of public and private data (this entails support for 2-level control of information – i.e. public and private) in one code, retention of appearance and apparent properties of regular QR codes [4], [67], etc. Unlike the aforementioned SQRs, our proposed CA-based QR code security protocol is not constrained to a specially designed reader nor does it require tweaks to available readers in the market. Instead, CASP is designed as a tool to produce plug-and-play secure codes expected to seamlessly work with average technology

that is already available in the market. However, like in SQRs [4], [67], ‘‘CASPed’’ QR codes are formulated to retain the aesthetic appearance and ambiance of a typical innocuous QR code. Therefore, we seek to exclude the three finder patterns, the main alignment pattern (i.e. in higher version codes this is the located in the rightmost lower part of the code), the separator cells (one cell all around the 3 finder patterns), two additional cells encircling the separator cells, which incidentally include the format and version information about the code and, finally, the one-line cells connecting two pairings of the 3 finder cells. Combined, these excluded cells consist of 301 modules constituting between 68% (in version 1) to 1% (in Version 40) of the QR Code area varying proportionally the size of the QR Code. For example, the excluded region constitutes 48 and 36% of the total area of Versions 2 and 3 QR Codes respectively. The remaining *encrypt-able* content of the code is confined to the + -like region of the code as shown in Fig. 7(a) for a Version 2 QR Code. As seen in the figure, the three position detection patterns and one alignment patterns are excluded from demarcated encrypt-able space.

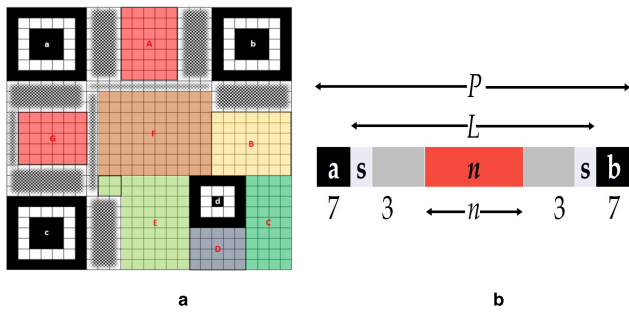


FIGURE 7. Structure and CASP zoning of a QR Code: (a) example of a version 2 QR Code and (b) dimensions of its loop.

Guided by the structure of the QR Code depicted in Fig. 7(a), we deduce the relationship between key parameters of a QR Code and the size of its loop ( $L$ ), which is the space between the two main finder patterns (labelled a and b in Fig. 7(a)) as presented in Fig. 7(b).

Consequently, using the two figures in Fig. 7, the encryption area of a QR Code ( $E_A$ ) can be computed using (2)

$$E_A = D - (298 + 2L + 2n) \tag{2}$$

Next, the demarcated encrypt-able space is delineated into seven zones each comprising of  $m$  sub-blocks of nine cells called tiles. Structurally, the zoning percolates regions between two pairings of the 3 finder patterns (i.e. a with b and a with c in Fig. 7(a)) which produce similar zones whereas areas surrounding the main alignment pattern are divided into three zones. In higher version QR Codes, the leftmost of these two zones merges with the zone closest to the finder pattern labelled c. Similarly, in such higher version codes, the central zone (i.e. Zone E in Fig. 7(a)) merges with parts of the zone directly below the top right finder pattern (labelled as b in Fig. 7(a)).

Notwithstanding the mergers mentioned here, our analysis shows that, based on our zoning strategy, modifications to at least any two zones is enough to encrypt contents of the QR Code, i.e. to render the code inaccessible (unscannable).

TABLE 2. Casp parameters for different versions of QR codes.

Parameter	Version	1	2	3	4	...	40
$P$		21	25	29	33	...	177
$Z$		441	625	841	1089	...	31329
$L$		7	11	15	19	...	163
$n$		1	5	9	13	...	157
$k$		0	2	6	8	...	104
(for Zone A)							
$E_A$		149	317	517	749	...	30413
$N_T$	T	13	32	54	80	...	3376
	A	6	28	48	70	...	N. A
$\% E_A$		0	6	11	10	...	3

Table 2 summarises some essential properties of the zoning for different versions of QR Codes. Meanwhile, based on the zone merging explained earlier, it is trivial that in the process of concatenating the  $3 \times 3$  tiles, some tiles will be orphaned, i.e. they cannot be part of any tile; thus, adding to the unaltered number of cells of the QR Code. Furthermore, in the tiling within each zone we assign a preference to demarcating the  $3 \times 3$  tiles in a clockwise direction but towards the top and to the right side. Therefore, the actual number of tiles  $N_T$  deviates from the theoretical values reported in Table 2 (shown as  $T$  and  $A$  respectively in Table 2). For example, as presented in Fig. 8, Versions 2 and 3 QR codes will have 28 and 48 tiles respectively. These results are reported as actual ( $A$ )  $N_T$  values in Table 2. Meanwhile, the ‘‘orphaned’’ cells are highlighted in blue for Versions 2 and 3 QR Codes in Fig. 8(a) and (b).

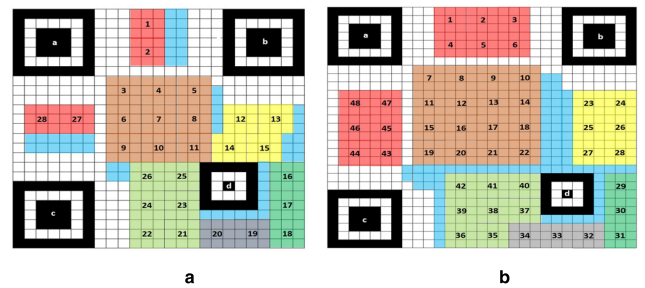


FIGURE 8. Illustration of actual CASP tiling for versions 2 and 3 QR codes in (a) and (b) respectively.

We conclude the description of the encrypt-able space of a QR code by pointing out that, structurally,  $S_Q$  consists of  $m$  carefully selected tiles whose spatial location across the seven zones of the pre and post encrypted QR code are known beforehand. Further, as enunciated in Section I, in both the original and encrypted QR codes ( $Q$  and  $Q^*$ ) the cells of tile  $m$  ( $T_m$ ) are in either one of two possible states – 0 or 1. Furthermore, a cell is considered occupied if it is filled (i.e. state 1) and empty otherwise.

Fig. 9 presents the outline of a typical tile,  $T_m$ , of a QR Code where the numbering in red indicates labels designating



0	1	2
1	2	4
3	4	5
8	16	32
6	7	8
64	128	256

FIGURE 9. Structure of a QR code tile.

the address of each pixel (i.e. cell or module) in the tile, while the other numbers specify the positional value of each cell in computing the sum total of occupancy or strength of the tile.

At this point, we define the *strength*,  $V_m$ , of tile  $m$  as the cumulative position-wise occupancy (i.e. being filled or not) of cells in tile  $m$ . Here, *occupancy* refers to cells with black (i.e. value 1) values in tile  $T_m$ , and so,  $V_m$  can be computed using (3).

$$V_m = \sum_{j=0}^8 (b_j \times 2^j) \quad (3)$$

where each  $j \in 0$  to 8 which, as mentioned earlier, are labels designating the address of each cell in the tile (shown in red in Fig. 9).

Later in Section V we will expatiate on the use of the LUT in Fig. 4(a) to attune each  $V_m$  to a CA ruleset, which, as outlined earlier, can produce any of  $2^{2^8}$  rulesets for tile  $T_m$ . However, based on the structure of our tile in Fig. 9, for two state, nine neighbourhood CAs only around a quarter of (i.e.  $2^8$  or 256) of these rules similize with requirements of our CASP protocol. Further, we limit the pool to only linear, balanced and reversible rulesets (Fig. 4(a)) [55], [57].

## 2) DEXTRAL BOUNDARY CONDITIONS AND DBC PERMEATED TRANSITION RULES

At this juncture, we recall that, as presented in Section II, in standard CA, at state  $t + 1$  its middle cell (row d in Fig. 1) is modified subject to constraints imposed on it and its neighbours via a transition rule. In this study, however, we employ a slight modification to the transition process. Given a cell of the 3-neighbourhood (or radius of 1) CA at state  $t$  such that contrary to usage in standard CA boundary conditions (periodic, reflective, etc.), we opt for a right-going mutation or triumvirate dextrality to steer the evolution of each ternary cell unit at the next state (i.e.  $t + 1$ ). This implies that each ternion of cells at state  $t$  produces the leftmost entry at the next iteration of the CA's evolution.

Given the earlier illustration of a periodic boundary condition (PBC) for six cell entries  $a, b, c, d, e, f$  (i.e. in Fig. 2(a)), variations in terms of the fecundity of the CA evolution obtained for the PBC and our proposed dextral boundary condition (DBC) at state  $t + 1$  are expounded in Fig. 10 (a) and (b) respectively. As seen therefrom, the DBC is a pseudo-PBC, but with the constraint that it alters each cell by one position to the right. Although (as presented in Fig. 10) the impact of our proposed DBC in altering lower level evolution of a CA is minuscule, compared to the PBC its higher-level imprint on the composition of an image is colossal.

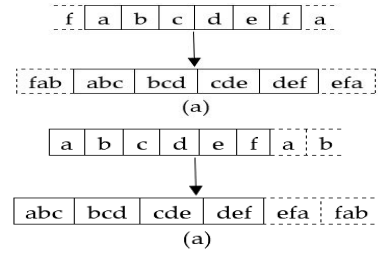


FIGURE 10. Right-going periodic boundary conditions employed for 3-neighbourhoods in the CASP protocol.

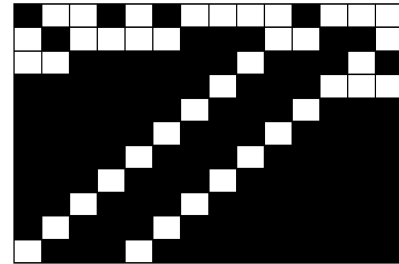


FIGURE 11. DBC-evolution of initial state 1001010000100 (first row) and outcomes using rule 75 after ten iterations based on the dextral boundary conditions of our proposed CASP protocol.

To illustrate the imprint of the proposed DBC on an image, consider the quiescent state  $q = 1001010000100$  (used earlier in Fig. 3) whose evolution using Rule 75 through ten iterations as presented in Fig. 11. Comparing this outcome with that obtained after ten iterations using a PBC boundary condition in Fig. 3 presents a cursory illustration of how the DBC impacts on the execution of a CA ruleset at each iteration. Comparing the two outcomes provides an outlook regarding the impact of the DBC on the evolution of an image.

Meanwhile, our instinct for using the tile strength exploits the Boolean composition of each ruleset (i.e. obtained from occupancy of the tiles), as a mechanism to use Boolean operations in transforming one rule set to another. In addition to the reversible rulesets that were highlighted in Section II, we propose the use of first and second-tier functions to serve as beacons for accomplishing the encryption and recovery procedures. These rulesets and their implementation are broached in the sequel.

## 3) FIRST AND SECOND TIER CASP RULESETS

By assuming  $b_0 = a, b_1 = b, b_2 = c, b_3 = d, b_4 = e, b_5 = f, b_6 = g, \text{ and } b_7 = h$ , we can generalise Eq. (1) to define a given ruleset  $D$  as

$$B = a b c d e f g h \quad (4)$$

where, like its use in (1),  $D$  has upper and lower configuration of 00000000 (i.e. equivalent to a decimal value  $N_B = 0$ ) and 11111111 (i.e. equivalent to a decimal value  $N_B = 256$ ), then its complement (i.e. binary  $\bar{B}$ ) and its equivalent to a decimal  $N_{\bar{B}}$  can be defined in (5) and (6) respectively.

$$\bar{B} = \overline{a b c d e f g h} \quad (5)$$

$$N_{\bar{B}} = 255 - N_B \tag{6}$$

Similarly, we can define the inverse of  $B$  (as  $\hat{B}$ ) in the form

$$\hat{B} = hg fedcba \tag{7}$$

and its mirror  $\tilde{B}$  as

$$\tilde{B} = aecgbfdh \tag{8}$$

However, we emphasize that, throughout the implementation of above operations,  $B$  is assumed to be a balanced ruleset, i.e. having an occupancy of  $2^{n-1}$ . Consequently, since all the subsequent operations emanate from  $B$ , it is trivial that they are also balanced rulesets.

Moreover, unlike in traditional nomenclature where inversion and complementation mean the same thing, and are considered interchangeable; here, as seen in (5) and (7), the inverse and complement operations are not the same. In our usage, only the complement operation retains the standard meaning whence a state 0 flips to 1 and vice versa. We use Table 3 to further elucidate the difference between these two operations as well as the mirror operation on state  $B$ . Using Rule  $N_B = 75$  for which  $B = 75 = 01001011$  produces a complement  $\bar{B} = 101100100$  and inverse  $\hat{B} = 11010010$  where it is palpable that these are equivalent to decimal numbers 180 and 210 for the complement and inverse respectively. As mentioned earlier, the expression in (6) provides the mathematical interpretation of the complement operation in decimal format and so, using it,  $N_{\bar{B}} = 255 - 75 = 180$  as is also deducible from Table 3. Another useful ruleset is the mirror of  $B$  denoted as  $\tilde{B}$  (Eq. (6)), which produces  $\tilde{B} = 01011001$  equivalent to decimal number 89 in the case of Rule 75 (i.e. Table 3).

**TABLE 3. Illustration of first and second-tier rulesets emanating from rule 75.**

TAG	First-tier rulesets				Second-tier rulesets		
	$B$	$\bar{B}$	$\hat{B}$	$\tilde{B}$	$\Lambda_B$	$\Omega_B$	$\psi_B$
$a$	0	1	1	0	0	1	1
$b$	1	0	1	1	0	0	0
$c$	0	1	0	0	1	0	1
$d$	0	1	1	1	0	1	0
$e$	1	0	0	1	1	1	0
$f$	0	1	0	0	1	0	1
$g$	1	0	1	0	0	1	1
$h$	1	0	0	1	1	0	0
Decimal	75	180	210	89	45	154	166
$f_{\bar{B}}$	5	5	5	5	5	5	5
$n_B$	11	4	2	9	13	10	6

Generally, as outlined so far, since the rulesets  $\bar{B}$ ,  $\hat{B}$  and  $\tilde{B}$  are obtained from direct transformations on rule  $B$ , we call them first-tier rulesets. Fig. 12 presents a mapping of complement operation as a reflection of one rule from another where the highlighted (in red) rules represent a selection from the balanced reversible rules presented earlier in Fig. 4.

Next, by adopting a nomenclature whereby uppercase letters depict states that have been transformed twice, we broach

the complemented mirroring, inverted complementation and inverted mirroring operations, which can be accomplished via sequential execution of the complement, inverse and mirror operations. Table 4 summarises these second-tier rulesets, including their notation, naming and interpretation.

**TABLE 4. Notations for second-tier recovery rulesets.**

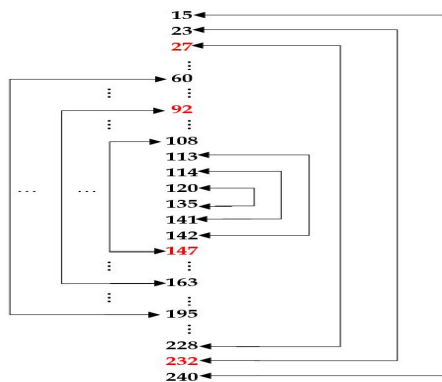
Notation	Nomenclature	Formula
$\Lambda_B$	Complemented inverse	$(\bar{\hat{B}}) = \overline{HG FEDCBA}$
$\Omega_B$	Mirrored inverse	$(\tilde{\hat{B}}) = \overline{HDFBGCEA}$
$\psi_B$	Complemented mirror	$(\bar{\tilde{B}}) = \overline{AECGBFDH}$

It is noteworthy that, contrary to the use in standard Boolean logic where the complement complement of  $B$  and/or inverse inverse of  $B$  (usually denoted as  $\bar{\bar{B}}$  or  $B''$ ) is  $B$ ; here, the complemented inverse and inverted complement of  $B$  may or may not be equal to  $B$  as demonstrated via the  $\Omega_B$  operation in Table 4. The last three columns of Table 3 illustrate the use of the the first-tier transformations in equations (3) through (7) and Table 4 to obtain the second-tier rulesets for Rule 75.

Whereas the two categories of rulesets (i.e. first and second tier) expounded here will play decisive roles in our CASP protocol, it is important to emphasize that, in CA, only specific sequences of reversible rules (i.e. mirror-able rules) form reversible CA. For example, based on the rule min term (RMT) concept in [57], the CA state arising from the sequence  $90 \rightarrow 15 \rightarrow 85 \rightarrow 15$  is considered reversible, whereas  $90 \rightarrow 85 \rightarrow 15 \rightarrow 15$  is not. Moreover, a CA is considered reversible only if the global transition function is one-to-one mapping. This implies that every configuration not only has one successor but also one predecessor [55]. Operations to QR codes in our proposed CASP are built with this intuition, a property that also guarantees efficient retrieval of the original QR code during the recovery process. Further, it has been demonstrated that reversible CA configurations after  $t$  evolution steps can be realised from rulesets that are two steps back (i.e.  $t - 2$ ).

At this juncture, we reiterate that, whereas we envision instances when 100% recovery of pristine QR code tiles will be accomplished, the main goal of our CASP protocol is to ensure that authorised users have access to the full functionality of the code as a transitory media. Therefore, 100% recovery of the unenciphered QR code is not necessary since concordance between the original ( $Q$ ) and recovered ( $Q'$ ) codes is less than the error correction threshold  $e$  for that code. Consequently, while full decryption of an encrypted QR code is desirable, it is not a necessary condition for the functionality of CASPed QR codes.

Furthermore, as adduced earlier, unlike standard symmetric-key encryption (SKE)-based algorithms where separate plaintext, key, and cover content are required, our proposed CASP assumes the option to securely share a recovery key (if needed), while the QR code itself serves as



**FIGURE 12.** Mapping the complement operation as the reflection of one rule from another.

both as the cover image and source for the plaintext and other information required for the encryption process.

Additionally, as highlighted earlier in Section III, the property of “balancy” is an important aspect of reversible rulesets. From this, we deduce another important property of our first-tier rulesets known as flicking or flipping. This denotes the number of times entries in a ruleset flip from 0 to 1 and/or from 1 to 0. Counting this flipping value (highlighted using red and blue arrows in Table 3) computes the total flipping value of a ruleset of  $B$  denoted as  $f_B$ .

By now one would have noticed that throughout we have used 8 bits (i.e. 1 byte) to represent our rulesets. However, as widely used in other areas of computer science and engineering, a byte can be divided into two nibbles each 4 bits long. Therefore, a ruleset can be divided into a lower (i.e. bits  $abcd$ ) and upper (i.e. bits  $efgh$ ) nibble. For example, the lower nibble for rule  $B = 75$ ,  $n_B$  be would be 1011 (i.e. decimal value of 11). This aspect of our ruleset is important in accomplishing the recovery process of our CASP whence balanced lower nibbles will be sought after.

Both the encryption and recovery procedures of our proposed CASP protocol will be built on the conventions of “balancy”, occupancy, DBC, lower nibble value, and flipping value of the first and second-tier rulesets to produce and retrieve the encrypted and recovered QR codes. The interplay between these properties and operations result in the encryption and recovery algorithms of our CASP protocol, which are presented in the remaining sections of the study together their implementation and experimental validation.

**B. ENCRYPTION AND DECRYPTION ALGORITHMS OF PROPOSED CA-QRS PROTOCOL**

We start by briefly recapitulating the objectives of our CASP protocol whose main building blocks are divulged in this section.

First, unlike standard protocols where the efficiency of a decrypted image is measured in terms of its fidelity with its pristine version, in our scheme we seek to guarantee confidentiality which entails restricting access to certain only to users with prior privileges to do so. Further, as QR Codes

access infers the ability to successfully use them as conduits to access the information then the confidentiality of sensitive information is safeguarded when access is denied, i.e. the QR Code is unscannable. In other words, access to the information concealed via the QR codes is confined to selected or authorised users. Consequently, only authorised users will be able to recover full functionality of the code, which takes them to the correct landing web page or information. Therefore, we reiterate that total fidelity between the recovered and uncyphered QR code is neither a requirement nor a necessity for safeguarding confidentiality.

Second, contrary to traditional encryption schemes where an encrypted signal is measured in terms of the distortions it undergoes, our study relies on conscientious study of the structure and singularity of the QR code to realise the objective of safeguarding confidentiality of the information it conceals. Hence, it is neither necessary nor expedient to distort large areas of the encrypted QR code.

A first impulse would be to tailor the encryption around the most sensitive content such as the finder patterns, alignment patterns, version information, etc. of the QR code. Indeed, modifying these areas would produce inaccessible QR codes. However, it would also yield codes whose physical appearance easily instigates suspicions from potential trespassers. Additionally, to preserve the above-mentioned criteria one would consider avoiding these areas and undertaking wholesome contortions to the encrypt-able space – say, by inverting every black cell to white and vice versa – but this approach would yield weakly encrypted codes. As a compromise between the desire to retain the physical appearance of the QR codes while also realising a strong, dynamic strategy to encrypt them, we undertook extensive and conscientious study of the QR code structure and identified certain regions containing mostly data and error correction cells where small changes impact on the encrypted codes. Our analysis identified subtle relationships between different areas or zones of the QR codes and we seek to use this to subitize encrypt-able cells and zones of the QR code. Therefore, in encrypting a QR Code, we must remain conscious of its error correction capacity.

Furthermore, whereas QR Code are ingrained with error correction capability to restore data if the code is dirty or damaged it increases the amount of data allowance and size of the QR Code.

Although the encryption and recovery procedures of our CASP differ in terms of execution they are both built on the same four initial parameters – namely, strength  $V_m$ , equivalent rule value  $E_m$ , transition rule  $R_m$  and binary equivalent of the ruleset  $B_m$  – all of which are extracted from the tiles that make up the original and encrypted QR codes, i.e. ( $Q$ ) and ( $Q^*$ ). These four parameters form the acronym *VERB*; hence, the names *VERB<sub>E</sub>* and *VERB<sub>R</sub>* for the encryption and recovery algorithms of our CA-based QR code security protocol (CASP) designed to safeguard the integrity of QR codes, which are discussed in the sequel.

**Algorithm 1** VERB<sub>E</sub> Algorithm for QR Code Encryption

Input: QR code ( $Q$ )  
 Output: Encrypted QR code ( $Q^*$ )

- 1 Demarcate the encrypt-able space,  $S_Q$ , of  $Q$
- 2 Delineate the zones of  $S_Q$
- 3 For each zone  $z$  in  $S_Q$  do
- 4     Partition each zone into  $m$  tiles,  $T_m$
- 5     Decide on key zone  $k$
- 6     For each tile in  $k$ , i.e.  $T_{m_k}$ , compute the encryption location using (9)

$$T_{m_k} = \frac{\sum v_i}{2 \frac{N_T}{n}} \tag{9}$$

where  $\sum v_i$  is the strength of tile  $m_k$  as defined in (3) and both  $N_T$  and  $n$  depend on the version of the QR code as presented in Table 2.

- 7     Encrypt tiles  $T_{m_k}$  of zone  $k$  using the operation

$$T_{m_k}^* \xrightarrow{240} T_{m_k} \tag{10}$$

where  $\xrightarrow{240}$  denotes the DBC permeated evolution of tiles  $T_{m_k}$  using Rule 240

- 8     End
- 9     For each encryption tile,  $T_{m_k}$ , compute the VERB<sub>E</sub> parameters as follows
- 10         Compute the strength  $V_m$  (using (3))
- 11         Determine the equivalent value  $E_m$  (using (11))

$$E_m = \begin{cases} V_m & \text{if } V_m \leq 128 \\ 255 - V_m & \text{if } V_m > 128 \leq 255 \\ \lfloor \frac{1}{2} (511 - V_m) \rfloor & \text{if } V_m > 511 \end{cases} \tag{11}$$

where  $\lfloor \cdot \rfloor$  is the floor operation that returns the integer part of the result

- 12         Match each  $E_m$  value to the ruleset,  $R_m$ , closest to its value in the look up table (LUT) in Fig. 4. Here, preference is given to lower values co-located in the same quadrant of the LUT.
- 13         Convert each  $R_m$  value to its binary value  $B_m$  (using (1))
- 14         Determine the first tier (i.e. encryption) rulesets  $B$ ,  $\bar{B}$ ,  $\hat{B}$  and  $\check{B}$  (using (3) to (8) and Table 3) for the binary value  $B_m$ .
- 15         Compute the equivalent decimal value for each ruleset as  $N_B$ ,  $N_{\bar{B}}$ ,  $N_{\hat{B}}$  and  $N_{\check{B}}$ .
- 16         Compute the flipping value of each ruleset as  $f_B$ ,  $f_{\bar{B}}$ ,  $f_{\hat{B}}$  and  $f_{\check{B}}$ .
- 17         For  $f_B^* = \lfloor f_B, f_{\bar{B}}, f_{\hat{B}}, f_{\check{B}} \rfloor$  do
- 18             For  $N_B^* = \lfloor N_B, N_{\bar{B}}, N_{\hat{B}}, N_{\check{B}} \rfloor$  do
- 19                 Execute the operation

$$S_m^i = \epsilon_m T_m \tag{12}$$

where  $\epsilon_m$  is the DBC permeated CA operation using the ruleset that satisfies  $f_B^*$  and  $N_B^*$  applied on tile  $T_m$ . This produces a tile  $S_m^i$  at each iteration  $i$  of  $\epsilon_m$ .

- 20                 While  $V_m \neq 0$  or  $255$  do
- 21                     Terminate (12) if any of the 3 cases arising from (13) is satisfied

$$v_m^i = v_m^j \tag{13}$$

Case 1:  $j = 0$ , then  $T_m^* = S_m^{i-1}$   
 Case 2:  $j = i - 1$ , then  $T_m^* = S_m^i$   
 Case 3:  $j = 1$ , then  $T_m^* = S_m^j$

where  $v_m^i$  and  $v_m^j$  are the strengths of tiles  $T_m^*$  at iterations  $i$  and  $j$ ; and  $S_m^j$  is the operation defined in (11).

- 22                     Retain outcome in line 21 if the discordance between  $T_m$  and  $T_m^*$  as

$$d_m^i \geq e \tag{14}$$

Else, flip the highest occupied and unoccupied cells simultaneously until the condition in (14) is satisfied. Then,

$$T_m^* = rS_m^i \tag{15}$$

where  $rS_m^i$  is the revised composition of  $S_m^i$  after flipping its content as outlined earlier

- 23                     End
- 24                     End
- 25                     Otherwise,

$$T_m^* = S_m^{i-1} \tag{16}$$

Obtain

$$Q^* = T_1^* \cdots T_m^* \tag{17}$$

where  $T_{*mk}$  are the encrypted versions of  $m$  tiles in  $Q$  and  $\dots$  indicates the zone and tile-wise juxtaposing of the encrypted content to form the encrypted QR code  $Q^*$

- 26     End
- 27     End
- 28     End
- 29     End
- 30     Return  $Q^*$

**Algorithm 2**  $VERB_R$  Algorithm for QR Code Recovery

```

Input: Encrypted QR code (or tiles thereof)
Output: Recovered (i.e. retrieved) QR code
1 Demarcate the encrypt-able space,  $S_Q$ , of  $Q$ 
2 Delineate the zones of  $S_Q$ 
3 For each zone  $z$  in  $S_Q$  do
4   Partition each zone into  $m$  tiles,  $T_m^*$ 
5   Using secure communication, the User retrieves the zone from the Owner and un.masks recovery zone  $k$  and therefrom obtain the location of
   the recovery tiles  $T_{m_k}^*$  using (7)
6   For all tiles in  $k$  ( $T_{m_k}^*$ ), obtain the recovered tiles using (18)
                                      $T'_{m_k} \xrightarrow{240} T_{m_k}^*$  (18)
   where  $\xrightarrow{240}$  denotes the DBC permeated evolution of tiles  $T_{m_k}^*$  using Rule 240
7   End
8   For each tile,  $T_{m_k}^*$ , compute the  $VERB_R$  parameters as follows
9     Compute the strength  $V_m^*$  (using (3))
10    Determine the equivalent value  $E_m^*$  (using (7))
11    Match each  $E_m^*$  value to the ruleset,  $R_m$ , closest to its value in the look up table (LUT) in Figure 4. Here, preference is given to lower
    values co-located in the same quadrant of the LUT.
12    Convert each  $R_m^*$  value to its binary value  $B_m^*$  (using (1))
13    Determine the second tier (i.e. recovery) rulesets  $\Lambda_B$ ,  $\Omega_B$ , and  $\Psi_B$  (using Table 4) for the binary value  $B_m^*$ .
14    Compute the equivalent decimal value for each ruleset as  $N_\Lambda$ ,  $N_\Omega$ , and  $N_\Psi$ .
15    Compute the lower nibble value of each ruleset as  $n_\Lambda$ ,  $n_\Omega$ , and  $n_\Psi$ .
16    For  $n_B^* = [n_\Lambda, n_\Omega, n_\Psi]$  do
17      For  $N_B^* = [N_\Lambda, N_\Omega, N_\Psi]$  do
18        While  $m_B^* = rN_B^* \neq \Omega_m$  do
19          where  $m_B^*$  and  $rN_B^*$  are the rulesets satisfying lines 16 and 17 respectively
          Execute the operation
                                      $S'_m = \mathfrak{R}_m \overline{T_m^*}$  (19)
          Where  $\overline{T_m^*}$  is the cell-wise inversion (i.e. 0 to 1 and vice versa) of  $T_m^*$ 
          Else, execute
                                      $S'_{m,i} = \Omega_m T_m^*$  (20)
          where  $\Omega_m$  is adapted as the default recovery ruleset. However, when  $\mathfrak{R}_m = \Omega = \Lambda$  (or  $\Psi$ ) we use  $\mathfrak{R}_m = \Lambda_m$  (or  $\Psi_m$ ).
20          While  $V_m^i \neq 0$  or 255 do
21            Terminate operations in (19) and/or (20) if any of the 3 cases in (20) is satisfied
22                                     Case1 :  $V_m^i = V_m^0$ , then  $T'_m = S'_{m,i-1}$ 
                                     Case2 :  $V_m^i = V_m^{i-1}$ , then  $T'_m = S'_{m,i}$ 
                                     Case3 :  $V_m^i = V_m^j$ , then  $T'_m = S'_{m,l}$  (21)
          where  $V_m^i$  and  $V_m^j$  strengths of tiles  $T_m^*$  at iterations  $i$  and  $j$ ;  $S'_{m,l}$  is the resulting tile between  $i$  to  $j$  with the lowest
          strength  $V'_{m,l}$ 
          Otherwise,
                                      $T'_m = S'_{m,i-1}$  (22)
23          End
24        End
25      End
26    End
27  End
28  Obtain
                                      $Q' = T'_1 \cdots T'_m$  (23)
  where  $T'_m$  are the retrieved versions of  $m$  tiles in  $Q$  and  $\cdots$  indicates the zone and tile-wise juxtaposing of the recovered content to form
  the encrypted QR code  $Q'$ 
29 End
30 End
31 Return  $Q'$ 

```

The step in line 22 of this algorithm ensures that the encrypted QR code ( $Q^*$ ) has a discordance greater than some threshold, which itself depends on the error correction ( $e$ ) specification of the QR code version (Table 1). For a version H QR code  $e = 0.3$ , which signifies maximum (i.e. three in one) tolerance to geometric alterations, chromatic manipulation, addition of different types of noise; introduction of foreign impurities and tampering with its position pattern. This threshold value can be adjusted to accommodate QR codes with different error correction levels.

Given an encrypted QR code  $Q^*$ , the following steps enumerate the procedure to retrieve its recovered version  $Q'$ .

We note that, for the condition in line 22, counting of the iterations of the CA starts from the inversion of  $T_m$ , if used (i.e. in the case of the recovery operation in (14)). Furthermore, as noted in Lemma 4.1 of [68], if a CA is reversible, then its inverse is a cellular automaton. This fundamental construction supports the conclusion that if a global process described by local interactions is invertible, then the inverse global process equally has a local map (or construction). In other words, an inverse global process can be realised via local interactions.

Nevertheless, unlike the common practice in the decryption process of many encryption-based security schemes where 100% recovery of the unenciphered versions of the media is targeted, as mentioned earlier, in our CASP strategy, recovery is deemed successful when discordance between the recovery pair (i.e.  $Q$  and  $Q'$ ) $d_R$  is less than the error correction threshold  $e$  (i.e.  $d_R < e$ ) of the code. Moreover, since both our encryption and recovery algorithms are based on one tile, it is trivial that (like the so-called shares in visual cryptography schemes (VCS) [13] these tiles are juxtaposed (i.e. abutted side by side) to form the encrypted QR code as well as its recovered (or retrieved) version, i.e.  $Q^*$  and  $Q'$  that is inferred in the last steps of both of our algorithms.

Finally, we complete this section by presenting a simple example to illustrate the execution of both the encryption and recovery procedures of our CASP protocol.

**C. EXECUTION OF THE CASP PROTOCOL**

Given six arbitrarily synthesised tiles  $T_A$  through  $T_F$  (in Fig. 13) of some zone of a QR code ( $Q$ ) as presented in Fig. 14(a), in the encryption process, we start by computing their  $VERB_e$  parameters (i.e. using (3)-(17)) for each tile as summarized in Table 5. Next, following execution of the remaining steps of the encryption algorithm, each tile will have a different encryption (i.e. first tier) ruleset,  $\epsilon_m$ , as presented in Table 6. Assuming  $Q$  has a level H error correction (i.e.  $e = 0.3$ ), then by executing the  $VERB_E$ , i.e. encryption, algorithm we obtain the encrypted versions of each tile as  $T^*_A$  through  $T^*_F$  (Fig. 14(b)).

Similarly, using (14)-(23),  $VERB_R$  parameters are computed and subsequently used to execute the recovery algorithm. The recovery (i.e. second tier) rulesets,  $\mathfrak{R}_m$ , required to assign recovery operations for each tile is presented in the last column of Table 6.

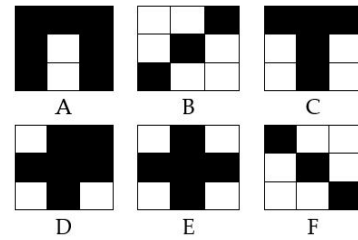


FIGURE 13. Six tiles of a synthetic QR code  $Q$ . See text for further explanation.

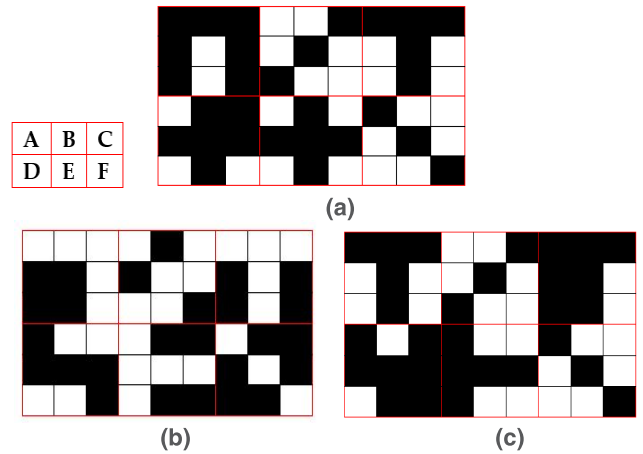


FIGURE 14. (a) Original code (juxtaposing tiles in Fig. 13), (b) Encrypted, and (c) recovered versions of the tiles A through F in Fig. 13.

TABLE 5.  $VERB_e$  encryption parameters for the tiles in fig. 13.

Parameter	A	B	C	D	E	F
$V$	367	84	151	190	186	273
$E$	72	84	104	65	69	18
$R$	75	83	102	60	60	15
$B$	01001011	01010011	01100110	00111100	00111100	00001111

In all the stages enumerated, abutting the six tiles produces the original ( $Q$ ), encrypted ( $Q^*$ ) and recovered ( $Q'$ ) QR codes as presented in Fig. 14(a), (b) and (c) respectively.

We especially note the discordance  $d_Q$  between the encryption pairing of original versions of tiles  $T_A$  and  $T_C$  with their encrypted versions  $T^*_A$  and  $T^*_C$  (aka encryption pairs). Here, discordance is the statistical computation of cell-wise dissimilarity between a pristine and encrypted tile. The discordances  $d_A$  and  $d_C$  (for tiles  $T_A$  and  $T_C$ ) are both 100% indicating absolute discord between the two encryption pairings. Likewise, we note 100% concordance between the recovery pairing for tiles  $T_A, T_B$  and  $T_F$  (i.e. with their respective recovered tiles  $T'_A, T'_B$  and  $T'_F$ ). Here, concordance measures fidelity between the recovered tiles, i.e.  $T'_A, T'_B$  and  $T'_F$  and their pristine versions, i.e.  $T_A, T_B$  and  $T_F$ . (aka recovery pairs).

Conclusively, comparing the encryption pairing of the QR code (i.e.  $Q$  (in Fig. 14(a) and  $Q^*$  in Fig. 14(b)), we see a discordance of 74%. This is more than double the maximum tolerance that the QR code can withstand (i.e. 30% for a level H code). Similarly, the recovery pairing (i.e.  $Q$  (in Fig. 14(a)

TABLE 6. Encryption and recovery rulesets for tiles in Fig. 13.

TILE	ENCRYPTION (FIRST TIER) RULESETS					RECOVERY (SECOND TIER) RULESETS						
	$N_B$	$N_{\bar{B}}$	$N_{\hat{B}}$	$N_{\tilde{B}}$	$\epsilon_m$	$N_{\Lambda}$	$N_{\Omega}$	$N_{\psi}$	$n_{\Lambda}$	$n_{\Omega}$	$n_{\psi}$	$\mathfrak{R}_m$
<i>A</i>	75	180	210	89	$\hat{B}$	27	172	202	11	12	10	$\Omega_m$
<i>B</i>	83	172	202	59	$\hat{B}$	225	86	149	1	6	5	$\Omega_m$
<i>C</i>	102	153	102	60	$\bar{B}$	45	154	166	13	10	6	$\Lambda_m$
<i>D</i>	60	195	60	102	$\bar{B}$	57	156	198	9	12	6	$\Omega_m$
<i>E</i>	60	195	60	102	$\bar{B}$	195	102	153	3	6	9	$\Omega_m$
<i>F</i>	113	142	142	43	$\hat{B}$	15	170	170	15	10	10	$\Lambda_m$

TABLE 7. Description and properties of QR codes as experimental dataset.

Image						
Code						
Properties						
Notation	$Q_{DB}$	$Q_{AmE}$	$Q_{LA}$	$Q_{AL}$	$Q_{AM}$	$Q_R$
Owner/landing site	Dropbox	American Express	Lufthansa Airlines	PDF file containing [69]	PDF file containing [69]	Research Gate profile
Version	2	2	2	2	3	4
Error correction, $e$	L	L	L	L	M	Q
Dimension, $Z$	625	625	625	625	841	1089
Loop size, $L$	11	11	11	11	15	19
$n$	5	5	5	5	9	13
$k$	2	2	2	2	6	8
$E_A$	317	317	317	317	517	749
$N_T$	32	32	32	32	54	80

and  $Q'$  in Fig. 14(c) produces a concordance of 82%, for which the overall contamination  $100 - 82 = 18\%$  is below the 30% threshold required to successfully read the code. Thus, although not an exact match with its original version, if this were a complete QR Code, its recovered version would effectively serve its role as a transitory media to access the information it obscures.

In the next section, the intuition demonstrated via these synthetic tiles are used in real QR Codes to showcase the utility of our CASP scheme in safeguarding the confidentiality of sensitive information by confining access to authorized users.

## V. EXPERIMENTAL VALIDATION OF PROPOSED CASP PROTOCOL

Our study considers a QR as both an image and a transitory media. Consequently, we tailor the evaluation of the QR Code encryption and recovery procedures in terms of standard encryption and image-based metrics, especially the ability to scan both QR codes (i.e. encrypted and recovered) as conduits to access the concealed information – a test we call scanability. To clarify, in the latter validation, i.e. the recovery test, our experiment is based on the intuition that, a recovered QR Code is lawfully obtained; so, it should be scannable. However, users of the encrypted version are unauthorised to access to such data and so their codes should be unscannable. To the best of our knowledge, this is the first study that undertakes the evaluation along these indicators.

## A. EXPERIMENTAL SCENARIOS AND SETUP

It is important to start by clarifying the objective and possible scenario that our proposed CASP protocol can be deployed to. Our experiments are targeted at confining access to the information concealed via the QR code to legitimately authorized users. Consequently, unlike [9], our strategy is not aimed at using the QR code as an attack vector whence the encryption is tailored to “trick” unsuspecting users to malicious weblinks that resemble their target web address [8]–[11]. For example, [9] demonstrated how QR codes attack vectors are used to provide subtle yet malicious distortions to web addresses such as “aamericanexpresis” instead of “americanexpress”, “dropbnx” instead of “dropbox”, etc. Therefore, unsuspecting users could be exposed to different types of cybercrimes. Unlike these instances, our target is primarily to safeguard access to the intended weblink and/or data it leads to not minding where or what link it redirects the unauthorized user. This could have potential applications in telemedicine or online shopping whereby access to sensitive information is granted to users via use of the QR code as a transitory media.

Employing the same procedure used in the preceding section (i.e. for the synthetic tiles in Figs 13 and 14), here, we implement the encryption and recovery of six real and scannable QR codes (including those used in [9]) whose properties are summarized in Table 7. We assume that all the codes provide access to sensitive information and our sole

target is confine access to selected (i.e. authorized) users. Again, we emphasize that our strategy is not concerned with where such unauthorised access leads in as much as the confidentiality of the sensitive data is protected.

However, we point out the default choice of the loop, i.e. Zone A, as the key zone. Therefore, all keys used in the reported experiments are from this zone. Nevertheless, we stress that any other zone could be chosen and securely communicated to authorized users.

**B. ENCRYPTION AND IMAGE-BASED EVALUATION METRICS**

In this section, we outline the performance metrics and benchmarks used to assess the effectiveness of our proposed protocol. Most of the metrics employed are standard yardsticks employed to validate encryption and information security methods [70], [71].

**1) PIXEL CHANGE RATE**

Earlier in Section IV(A), we utilised the discordance and concordance metrics to loosely assess fidelity or lack thereof between pairings of the original and encrypted codes (i.e. the encryption pair) and that between the recovered and original codes (i.e. the recovery pair) respectively. The interpretation was that high discordance between the encryption pair guarantee that the authenticated information is inaccessible. Similarly, low concordance below the threshold ensures access to the information by legitimate users that are authorised to access the data. Much like this, the so-called number of pixel change rate (NPCR) is a ratio that is widely used in evaluating the effect of changing pixel values in an original image relative to its corresponding encrypted version [70], [71]. Mathematically, NPCR is computed using (24).

Loosely put, our use of this metric measures the concordance between an original ( $Q$ ) and recovered ( $Q'$ ) QR Codes, also known as the recovery pair. However, our computation of NPCR, which we denote as  $c_{Q'}$  in (24) (and as  $d_{Q^*}$  later in (26)) is a cell-wise analysis that assesses the spatial fidelity or mapping variations in cell occupancy between cells in an original QR Code and its encrypted and recovered version.

$$c_{Q'} = \frac{\sum_{i,j} X(i,j)}{N}, \tag{24}$$

where

$$X(i,j) = \begin{cases} 0 & \text{if } Q(i,j) = Q^*(i,j) \\ 1 & \text{if } Q(i,j) \neq Q^*(i,j) \end{cases}$$

and  $N$  denotes the dimension of the code. As defined in (24), it is trivial that higher values of  $c_{Q'}$  indicate fidelity between  $Q$  and  $Q'$ .

**2) HETEROGENEITY,  $\rho$**

Histogram analysis is a widely used measure in image analysis that reflects the frequency distribution of pixel values

in the image [71]. A well-designed image encryption algorithm should have uniform histograms for different encrypted images, which indicates potential for resistance against statistical attacks [71].

However, in binary images, since the pixel distribution is concentrated in the ends, the greyscale histogram presents long valley consisting of flattened out values [12]. Therefore, such a histogram may not effectively show the frequency distribution.

Meanwhile, during the design of Custom QR Codes, the notion and routine of masking is employed ascertain the ratio of black to white cells in a QR Code. This is considered because a graphic pasted in the QR Code will generate errors in the QR Code build-up, so a scanner could misread it as a corner marker (i.e. cell). By applying masks, this could be curtailed by having the least amount of errors that help ensure that the code useable [72]. The mask function in a QR Code is to distribute the white and black modules in such a way that the overall ratio is close to 50% and unwanted patterns are mostly eliminated [72].

We use the intuition of masking as explained above to present a pseudo histogram analysis in terms of assessing (1) how balanced (50:50) the distribution of white to black pixels are in the pristine unenciphered QR Codes and (2) how this ratio is affected by the CASP process, i.e. masking between the encryption and recovery pairings of the QR Codes. We call this property the heterogeneity of a QR Code ( $\rho$ ).

Consequently, heterogeneity,  $\rho$  measures the distribution of black to white cells in the code. Its usefulness stems in its use to interpret the preservation the distribution of white to black cells in the pristine (i.e. original) QR code as well as changes thereof in the encrypted and recovered versions of the code. We compute  $\rho$  using (25)

$$\rho = \frac{C_B}{C_W} \tag{25}$$

where  $C_B$  and  $C_W$  are the percentage of black and white cells in an  $N$ -dimensional code and values closer to 1.0 are desirable because they indicate better diffusion as would be expected in a QR Code.

Intuitively,  $\rho$  helps to measure the extent a Code retains the physical appearance of a standard QR Code.

**3) DISCORDANCE,  $d_{q^*}$**

As mentioned earlier, when assessed in terms of the recovery pairing between an original code ( $Q$ ) and its recovered version ( $Q'$ ), NPCR measures concordance (as defined in (10)) and in terms of the encryption pairing of original code ( $Q$ ) and its encrypted version ( $Q^*$ ) the same metric measures discordance. Here, we further formalise our description of this important metric. As the inverse of concordance, this metric provides quantitative assessment of dissimilarity between the encryption pair ( $Q$  and  $Q^*$ ). Mathematically,  $d_{Q^*}$  is the same as (24) but the numerator computes one-to-one cell-wise



dissimilarity between  $Q$  and  $Q^*$  as presented in (26).

$$d_{Q^*} = \frac{\sum_{i,j} X(i,j)}{N}, \quad (26)$$

where

$$X(i,j) = \begin{cases} 0 & \text{if } Q(i,j) = Q^*(i,j) \\ 1 & \text{if } Q(i,j) \neq Q^*(i,j) \end{cases}$$

and  $N$  denotes the dimension of the code.

Notwithstanding the computations in (25) and (26), we emphasize that, based on the CASP, the scan-ability test is the best test of discordance between the encryption pair.

#### 4) ERROR ALLOWANCE, $\delta$

By their design, QR Codes are ingrained with error correction capacity. These are hidden codes suffused in the code to enable scan-ability of the code within certain limits of noise, impurities, etc. As explained in previous sections of this study, the process of CASPing a QR Code introduces distortions to the pristine versions of the codes. Despite this, to retain functionality of the QR Code as a conduit to access some concealed information CASPed QR Codes should still have room for additional impurities that come with usage. The error correction allowance  $\delta$ , which are defined in (27) quantifies the leg room available in a CASPed QR Code.

$$\delta = e - d_{Q^*} \quad (27)$$

where  $e$  is the error correction threshold of the QR Code and  $d_{Q^*}$  is the discordance of the encryption pairing of the code. From (27), we can also compute the percentage of available error correction used up by CASPing the code and by this  $\Delta$  provides a measure of how much error a CASPed QR Code can tolerate as defined in (28).

$$\Delta = 100 - \left( \frac{\delta}{e} \times 100 \right) \quad (28)$$

#### 5) KEY SPACE ANALYSIS, $S$

The key space is quantitative assessment of the total number of dissimilar keys that an encryption protocol can accommodate. It assesses whether a scheme can withstand brute force and other attacks aimed at violating the integrity of the information being protected. Key space ( $h$ ) is calculated using the simple formulation in (29)

$$s = 2^h \quad (29)$$

where in the CASP scheme  $h = N_T \times k$  where  $N_T$  and  $k$  are the total number of tiles in a code and number of keys both of which depend on other QR Code parameters that were explained earlier in Section IV.

Based on extensive studies, over the years, benchmarks have been developed for most of the metrics listed above. These are outcomes adjudged as standard for any effective encryption protocol [70]. Within the precepts of our study, these metrics are summarized in Table 8 as guide to discussions on the performance of our CASP protocol.

**TABLE 8. Benchmark values for performance analysis.**

Metric	Notation	Benchmark values
Concordance (NPCR)	$C_Q$	> 48% (Binary) > 99% (Greyscale)
Discordance	$d_{Q^*}$	< $e$
Heterogeneity	$\rho$	1.0
Key space	$s$	> $2^{100}$

### C. RESULTS AND DISCUSSION

Assessment of the performance of our CASP protocol are divided into two – encryption and recovery tests – whose results and discussions are presented in this section.

Table 10 presents an overview of the encryption test using different pairings of the original and encrypted versions of the six codes in our dataset, i.e.  $Q$  and  $Q^*$  for  $Q_{DB}$ ,  $Q_{AmE}$ ,  $Q_{LA}$ ,  $Q_{AL}$ ,  $Q_{AM}$  and  $Q_{RG}$  QR Codes whose properties were enumerated earlier in Table 7.

The first row presents the heterogeneity or ratio of black to white cells in the original QR Codes (Table 7). Following that, the second shows the encrypted versions ( $Q^*$ ) of each of the six QR Codes (labelled (a) through (f) for  $Q_{DB}$ ,  $Q_{AmE}$ ,  $Q_{LA}$ ,  $Q_{AL}$ ,  $Q_{AM}$  and  $Q_{RG}$ ) and in each the distorted (i.e. modified) cells are highlighted in red. Further analysing this result, we see that in each of the codes, the encryption tiles emanate from at least two zones of the encryption space ( $E_A$ ) of the QR code. Next, in the third and fourth rows, results of the discordance values and error allowance for each encryption pair are presented. As seen from these results, using our CASP protocol, between 18 to 65% of the error correction capability of Version 2 QR Codes (i.e.  $Q_{DB}$ ,  $Q_{AmE}$ ,  $Q_{LA}$ ,  $Q_{AL}$ ) is retained. Similar caps of around 2 and 30% are retained for Versions 3 and 4 QR Codes (i.e.  $Q_{AM}$  and  $Q_{RG}$  codes). The fifth row shows that all the encrypted codes have enough mix of black and white cells (i.e. heterogeneity,  $\rho \approx 1.0$ ) needed to retain expected physical appearance of any innocuous QR Code. Finally, the fifth row presents outcomes of sample scan-ability test were each encrypted QR Codes was scanned using cheap scanners available to anyone. Outcomes show that all the encrypted QR Codes failed this test, and so, none of the six encrypted codes is accessible; hence, protecting the contents that scanning them leads to.

Table 10 showcases the recovery analysis for our proposed protocol whose metrics are based on pairings of the original and recovered versions of the six codes in our dataset as presented in Table 7.

The first row shows the recovered QR Codes wherein the cells highlighted in green indicate the recovered cells (labelled (a) through (f) for  $Q_{DB}$ ,  $Q_{AmE}$ ,  $Q_{LA}$ ,  $Q_{AL}$ ,  $Q_{AM}$  and  $Q_{RG}$ ). Like in the encryption, these cells traverse at least two zones from encrypt-able space ( $E_A$ ) of each QR Code. The second row presents the concordance values for each recovery pair. Notably, these values vary between 98% for Version 2 (i.e.  $25 \times 25$  for  $Q_{DB}$ ,  $Q_{AmE}$ ,  $Q_{LA}$ ,  $Q_{AL}$ ) and Version 3 (i.e.  $29 \times 29$  for  $Q_{AM}$ ) QR Codes to 99% for the

TABLE 9. Results of encryption analysis.













Test	Code	$Q_{DB}$	$Q_{AmE}$	$Q_{LA}$	$Q_{AL}$	$Q_{AM}$	$Q_R$
$\rho$ (%)		1.145	1.155	1.083	1.016	1.030	1.066
Encrypted							
		(c)	(a)	(b)	(d)	(e)	(f)
$d_{Q^*}$ (%)		4.16	4.32	1.28	3.20	7.30	7.30
$\Delta$ (%)		59.4	61.7	18.3	45.7	29.2	30.8
$\rho_{Q^*}$ (%)		1.118	1.042	1.049	1.049	1.044	1.074
Scan-ability		Failed	Failed	Failed	Failed	Failed	Failed

TABLE 10. Results of recovery analysis.

Test	Code	$Q_{DB}$	$Q_{AmE}$	$Q_{LA}$	$Q_{AL}$	$Q_{AM}$	$Q_R$
Recovered							
		(c)	(a)	(b)	(d)	(e)	(f)
$C_Q$ (%)		98.72	98.56	98.40	98.80	97.80	99.17
$\rho_{Q^*}$ (%)		1.255	1.049	1.016	1.070	1.051	1.006
Scan-ability		Passed	Passed	Passed	Passed	Passed	Passed
$s$		$2^{64}$	$2^{64}$	$2^{64}$	$2^{64}$	$2^{324}$	$2^{640}$

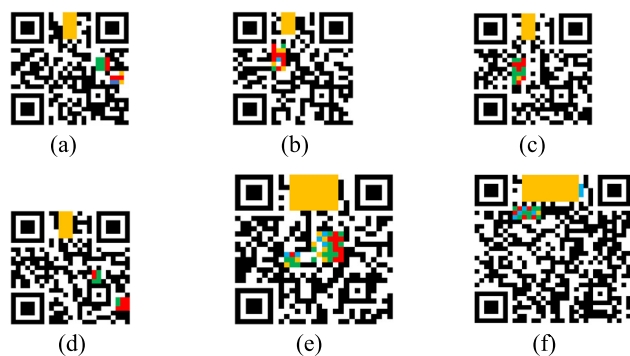
larger Version 4 (i.e.  $33 \times 33$  for  $Q_{RG}$ ) QR Codes reported. We note that, as earlier reported in Table 2, the theoretical critical benchmarks for NPCR values vary from 49% and 99% for binary and greyscale images respectively. Therefore, the fidelity exhibited between the recovery pairs is within the highest standards [70]. The third row of Table 10 reports the heterogeneity values for the recovered QR Codes. As expected, all the codes have values within the range of 1.0 which confirms an almost equal proportion between black and white cells, which suggest that the recovered QR Codes retain the physical appearance expected of QR Codes. Ultimately the objective of the recovery test is to regain full functionality of the QR Code. Results of the scan-ability test are presented in the fourth row of Table 7 and as reported all the recovered codes passed the test which means they could be scanned using standard scanners.

Finally, in the last column of Table 10, we present the key space analysis for the six QR Codes. Here, we recall, as mentioned in the opening remarks of this section, our choice of the loop zone as the default key zone. As a result of this, Version 2 QR code will have only two keys which explains the low key space ( $2^{64}$ ) for the  $Q_{DB}$ ,  $Q_{AmE}$ ,  $Q_{LA}$ ,  $Q_{AL}$  codes. It is trivial that, while the key space will be  $2^{64}$  in Zones A, D and G, changing the key zone will increase the key space to  $2^{96}$  for Zone C,  $2^{128}$  for Zone B,  $2^{192}$  for

Zone E and  $2^{288}$  for Zone F. Consequently, the Owner can decide which zone to use as the key zone and communicate same to authorised Users.

As seen from both the encryption and recovery the two tests (in Tables 8 and 9), our CASP protocol confines access to authorised persons. An unauthorised user with access to any of the encrypted QR Codes will (1) suspect nothing untoward because the encrypted QR Code looks like any other harmless, ready-to-use QR Code, and (2) be unable to access content concealed via its access. Meanwhile, validated users will be able to regain full functionality of the QR Code, and, so it is able to serve as a conduit to the concealed information.

Additionally, Fig. 15 presents CASP maps that depict the clutter in the encryption and recovery pairings of the different QR Codes used. While the NPCR presents a quantitative assessment of concordance and/or discordance, this map shows the specific cells that were altered in the encryption (in red) as well as recovery (in green) processes. Furthermore, whereas cells highlighted in yellow show areas of each code that were modified twice, i.e. during both the encryption and recovery processes, those in blue show cells that were untouched throughout both processes. Somewhat analogous to plotting the histogram difference [70], this map presents a detailed pixel-wise mapping of concordance (in green for the recovery pairing) and discordance (in red for the encryption



**FIGURE 15.** CASP maps for QR Codes (i.e. those labelled (a) to (f) in Table 9) examined in experiments reported in Tables 9 and 10.

pairing) as well as yellow and blue for the cells that were altered in both processes or preserved throughout both procedures.

To conclude, we surmise that the simple, yet quantitative experimental setup and results presented here demonstrate the tenability of our CASP protocol as a strategy to safeguard the confidentiality of sensitive information by restricting access to authorized users.

## VI. CONCLUDING REMARKS

Cellular automata (CA) is a universal machine capable of constructing any other machine once it is furnished with a set of instructions [55]. It is veritable tool that provides useful insights into the intricate composition of physical systems. Exploiting these astounding properties of CA, our study proposes a CA-based security protocol (CASP) to safeguard the integrity of QR codes whose main objectives are twofold: first, to render otherwise useable QR Codes unscannable to unauthorized users, and second, to do so while retaining the physical appearance and properties of an innocuous QR Code.

Guided by a conscientious study of the structure and properties of QR Codes and in order to attain the identified objectives of the CASP protocol, we developed a series of concepts to delineate a QR Code into its encrypt-able and unaltered regions with the latter comprising essential areas that aesthetically define a QR Code, i.e. the finder, alignment patterns, etc. unaltered. Meanwhile, the former, i.e. the encrypt-able space ( $E_A$ ) is further demarcated into seven zones each consisting of  $m \times 3 \times 3$  cells called tiles. Next, using the spatial location of white and black cells, aka occupancy, in each tile, a ruleset obtained globally from a 9-neighbour tile of a QR code was used to steer local interactions between internal states and immediate neighbours of the cells. This *glocal* framework produced evolutions in the state of a pristine QR Code to obtain its encrypted version, which could be considered as an emergent state determined by the 9-neighbourhood composition of the tiles in the unencrypted image and locally defined transition function that depends on the 3 neighbours of each pixel. In other words, global features of the tile shape the local interactions that evolve into a new global image state. The resulting encrypted image could be deemed as having *glocal* awareness in terms of its composition. Similarly, exploiting

the balanced, linear and reversible properties of CA used in the study, evolution of the encrypted QR Code to realise its recovered version is accomplished.

We employed image-based encryption metrics to validate the performance of our proposed protocol. Outcomes show variations in discordance, which measures dissimilarity between pairings of the original and encrypted codes, and concordance, which measures fidelity between pairings of the original and recovered QR Codes, ranging from 1 to 7% and 97 to 99% respectively for Versions 2, 3 and 4 QR Codes reported in the experiments. Furthermore, our results show that heterogeneity, which measures the distribution of white and black cells, is maintained in both the encrypted and recovered QR Codes. This metric indicates that all the QR Codes retained the physical appearance required to mitigate any untoward attempt to violate their integrity. Moreover, whereas all the encrypted QR Codes failed the scan-ability test, the recovered versions of all the codes passed the same test; thus, ensuring full functionality of the QR Codes as conduits to access sensitive information. Consequently, our proposed CASP can safeguard the confidentiality of QR Codes and the important data they are intended to conceal.

We envision that the emergence and ubiquity of 5G networks, IoT, blockchains, etc. will usher in new roles for QR Codes. Therefore, applications like the CASP presented in this study are important for emerging technologies for data storage in cloud computing, data transfer in Internet of Things (IoT), access control in online shopping, teleradiology, blockchain healthcare platforms, etc.

## REFERENCES

- [1] *Understanding the Fundamentals of Information Security*. Accessed: Aug. 30, 2019. [Online]. Available: <https://www.imagexmedia.com>
- [2] *Confidentiality, Integrity and Authentication (CIA) Triad*. Accessed: Aug. 30, 2019. [Online]. Available: <https://www.whatistechtarget.com>
- [3] *General Security*. Accessed: Aug. 30, 2019. [Online]. Available: <https://www.infosecurityinstitute.com>
- [4] *QR Code Essentials*. Accessed: Aug. 30, 2019. [Online]. Available: <https://www.Denso.com>
- [5] *Information Technology—Automatic Identification and Data Capture Techniques*, International Standard ISO/IEC 18004, pp. 1–144. Accessed: Aug. 30, 2019. [Online]. Available: <https://www.iso.org>
- [6] S. Tiwari, “An Introduction to QR Code Technology,” in *Proc. Int. Conf. Inf. Technol. (ICIT)*, Bhubaneswar, India, Dec. 2016, pp. 39–44.
- [7] P. Kieseberg, M. Leithner, M. Mulazzani, L. Munroe, S. Schrittwieser, M. Sinha, and E. Weippl, “QR code security,” in *Proc. 8th Int. Conf. Adv. Mobile Comput. Multimedia (MoMM)*, vol. 10, 2010, pp. 430–435.
- [8] T. Vidas, E. Owusu, S. Wang, C. Zeng, L. F. Cranor, and N. Christin, *QRishing: The Susceptibility of Smartphone Users to QR Code Phishing Attacks*. Pittsburgh, PA, USA: CMU-CyLab, 2012, pp. 1–12.
- [9] I. Kapsalis, “Security of QR codes,” M.S. thesis., Dept. Telematics, Norwegian Univ. Sci. Technol., Trondheim, U.K., 2013.
- [10] K. Krombholz, P. Frühwirt, P. Kieseberg, I. Kapsalis, M. Huber, and E. Weippl, “QR code security: A survey of attacks and challenges for usable security,” in *Proc. Int. Conf. Hum. Aspects Inf. Secur., Privacy, Trust (HAS)*, Crete, Greece, Jun. 2014, pp. 79–90.
- [11] K. Peng, H. Sanabria, D. Wu. *Charlotte Zhu. Security Overview of QR Codes*. pp. 1–20. Accessed: Aug. 30, 2019. [Online]. Available: <https://www.courses.csail.mit.edu>
- [12] V. Ramya and G. Gopinath, “Review on quick response codes in the field of information security (Analysis of various imperceptibility characteristics on grayscale and binary QR code),” in *Proc. Int. Conf. Adv. Eng. Technol. (ICAET)*, Singapore, Mar. 2014, pp. 1–5.

- [13] Z. Fu, Y. Cheng, and B. Yu, "Visual cryptography scheme with meaningful shares based on QR codes," *IEEE Access*, vol. 6, pp. 59567–59574, 2018.
- [14] Y. Chow, W. Susilo, J. Tonien, and W. Zong, *A QR Code Watermarking Approach Based on the DWT-DCT Technique*. Accessed: Aug. 30, 2019. [Online]. Available: <https://www.pdfsemanticscholar.com>
- [15] M. Luo, S. Wang, and P.-Y. Lin, "QR code steganography mechanism with high capacity," in *Proc. Int. Conf. Commun. Problem-Solving (ICCP)*, Piscataway, NJ, USA, Sep. 2016, pp. 1–2.
- [16] Y.-M. Wang, C.-T. Sun, P.-C. Kuan, C.-S. Lu, and H.-C. Wang, "Secured graphic QR code with infrared watermark," in *Proc. IEEE Int. Conf. Appl. Syst. Invention (ICASI)*, Chiba, Japan, Apr. 2018, pp. 690–693.
- [17] V. Sharma, "A study of malicious QR codes," *Int. J. Comput. Intell. Inf. Secur.*, vol. 3, no. 5, pp. 21–26, 2012.
- [18] N. V. Akhil, A. Vijay, and D. S. Kumar, "QR Code Security using proxy re-encryption," in *Proc. Int. Conf. Circuit, Power Comput. Technol. (ICCPCT)*, Nagercoil, India, Mar. 2016, pp. 1–5.
- [19] R. M. Bani-Hani, Y. A. Wahsheh, and M. B. Al-Sarhan, "Secure QR code system," in *Proc. 10th Int. Conf. Innov. Inf. Technol.*, Al Ain, UAE, Nov. 2014, pp. 1–6.
- [20] S. D. Degadwala and S. Gaur, "Two way privacy preserving system using combine approach: QR-code & VCS," in *Proc. Innov. Power Adv. Comput. Technol. (i-PACT)*, Vellore, India, Apr. 2017, pp. 1–5.
- [21] V. Hajduk, M. Broda, O. Kováč, D. Levický, "Image steganography with using QR code and cryptography," in *Proc. 26th Int. Conf. Radioelektronika*, Kosice, Slovakia, Apr. 2016, pp. 350–353.
- [22] M. Seredynski and P. Bouvry, "Block encryption using reversible cellular automata," in *Cellular Automata* (Lecture Notes in Computer Science), vol. 3305. Berlin, Germany: Springer, 2004, pp. 785–790.
- [23] S. Wolfram, *One-Dimensional Cellular Automata*, ch. 3, pp. 35–78. Accessed: Aug. 30, 2019. [Online]. Available: <https://www.pdfsemanticscholar.org>
- [24] S. A. Billings and Y. Yang, "Identification of Probabilistic Cellular automata," *IEEE Trans. Cybern.*, vol. 33, no. 2, pp. 225–236, Apr. 2003.
- [25] A. Scarioni and J. A. Moreno, *Border Detection in Digital Images With a Simple Cellular Automata Rule* (Cellular Automata: Research Towards Industry), S. Bandini, R. Serra, and F. S. Liverani, Eds. London, U.K.: Springer-Verlag, 1998.
- [26] C.-L. Chang, Y.-J. Zhang, and Y.-Y. Gdong, "Cellular automata for edge detection of images," in *Proc. Int. Conf. Mach. Learn. Cybern.*, Aug. 2014, pp. 3830–3834.
- [27] M. A. Lee and L. M. Bruce, "Applying cellular automata to hyperspectral edge detection," in *Proc. IEEE IGARSS*, Jul. 2010, pp. 2202–2205.
- [28] S. Djemame and M. Batouche, "Combining cellular automata and particle swarm optimization for edge detection," *Int. J. Comput. Appl.*, vol. 57, no. 14, pp. 16–22, 2012.
- [29] K. Zhang, Z. Li, and X.-O. Zhao, "Edge detection of images based on fuzzy cellular automata," in *Proc. 8th ACIS Int. Conf. Softw. Eng., Artif. Intell., Netw., Parallel/Distrib. Comput.*, Jul./Aug. 2007, pp. 289–294.
- [30] D. K. Patel and S. A. More, "Edge detection technique by fuzzy logic and cellular learning automata using fuzzy image processing," in *Proc. IEEE ICCCI*, Jan. 2013, pp. 1–6.
- [31] M. Pradipta and P. P. Chaudhuri, "Fuzzy cellular automata for modeling pattern classifier," *IEICE Trans. Inf. Syst.*, vol. E88-D, no. 4, pp. 691–702, 2005.
- [32] D. R. Nayak, S. K. Sahu, and J. Mohammed, "A cellular automata based optimal edge detection technique using twenty-five neighborhood model," *Int. J. Comput. Appl. Math.*, vol. 84, no. 10, pp. 27–33, 2013.
- [33] J. Mohammed and D. R. Nayak, "An efficient edge detection technique by two dimensional rectangular cellular automata," Dec. 2013, *arXiv:1312.6370*. Accessed: Aug. 30, 2019. [Online]. Available: <https://arxiv.org/abs/1312.6370>
- [34] S.-J. Ko and Y. H. Lee, "Center weighted median filters and their applications to image enhancement," *IEEE Trans. Circuits Syst.*, vol. 38, no. 9, pp. 984–993, Sep. 1991.
- [35] P. Maji, C. Shaw, N. Ganguly, B. K. Sikkhdhar, and P. P. Chaudhuri, "Theory and application of cellular automata for pattern classification," *Fundamenta Informaticae*, vol. 58, pp. 321–354, Jan. 2003.
- [36] K. Paul, D. R. Chaudhuri, and P. P. Chaudhuri, "Cellular automata based transform coding for image compression," in *High Performance Computing* (Lecture Notes on computer Science), vol. 1745, P. Banarjee, V. K. Prasana, and B. P. Sinha, Eds. Berlin, Germany: Springer, 1999, pp. 269–273.
- [37] C. Zhao, C. Shi, and P. He, "A cellular automaton for image compression," in *Proc. IEEE ICNC*, Oct. 2008, pp. 397–401.
- [38] W. Hong, Z. Hong-Jie, and W. Hua, "Image segmentation arithmetic based on fuzzy cellular automata," *Fuzzy Syst. Math.*, vol. 18, no. 11, pp. 309–313, 2004.
- [39] D. Safia, D. Oussama, and B. M. Chawki, "Image segmentation using continuous cellular automata," in *Proc. IEEE ISPS*, Apr. 2011, pp. 94–99.
- [40] S. Sadeghi, A. Rezvani, and E. Kamrani, "An efficient method for impulse noise reduction from images using fuzzy cellular automata," *AEU-Int. J. Electron. Commun.*, vol. 66, pp. 772–779, Sep. 2012.
- [41] K. K. V. Toh and N. A. M. Isa, "Noise adaptive fuzzy switching median filter for salt-and-pepper noise reduction," *IEEE Signal Process. Lett.*, vol. 17, no. 3, pp. 281–284, Mar. 2010.
- [42] F. Qadir, M. A. Peer, and K. A. Khan, "An effective image noise filtering algorithm using cellular automata," in *Proc. IEEE (ICCCI)*, Jan. 2012, pp. 1–5.
- [43] L. Yin, R. Yang, M. Gabbouj, and Y. Neuvo, "Weighted median filters: A tutorial," *IEEE Trans. Circuits Syst. II. Analog Digit. Signal Process.*, vol. 43, no. 3, pp. 157–192, Mar. 1996.
- [44] P. L. Rosin, "Image processing using 3-state cellular automata," *Comput. Vis. Image Understand.*, vol. 114, pp. 790–802, Jul. 2010.
- [45] A. Popovici and D. Popovici, *Cellular Automata in Image Processing*. Accessed: Aug. 30, 2019. [Online]. Available: <https://www.semanticscholar.org>
- [46] S. Uguz, U. Sahin, I. Siap, and H. Akin, "2D cellular automata with an image processing application," *ACTA Phys. Polonica A*, vol. 125, pp. 435–438, Apr. 2013.
- [47] H. Gutowitz, "Cryptography with dynamical systems," in *Cellular Automata and Cooperative Systems*. Springer, 1993, pp. 237–274.
- [48] S. R. Blackburn, S. Murphy, K. G. Paterson, S. Nandi, and P. P. Chaudhuri, "Comments on 'Theory and applications of cellular automata in cryptography' [with reply]," *IEEE Trans. Comput.*, vol. 40, no. 5, pp. 637–638, May 1997.
- [49] X. Liu, J. Lou, Y. Wang, J. Du, B. Zou, and Y. Chen, "Discriminative and robust zero-watermarking scheme based on completed local binary pattern for authentication and copyright identification of medical images," *Proc. SPIE*, vol. 10579, Mar. 2018, Art. no. 105791I.
- [50] A. V. Durga and S. Srividya, "A new algorithm for QR code watermarking technique for digital image using wavelet transformation," *Int. J. Eng. Comput. Sci.*, vol. 3, pp. 776–778, Aug. 2014.
- [51] T.-Y. Fan, H.-C. Chao, and B.-C. Chieu, "Lossless medical image watermarking method based on significant difference of cellular automata transform coefficient," *Signal Process., Image Commun.*, vol. 70, pp. 174–183, Feb. 2019.
- [52] M. Seredynski and P. Bouvry, "Block cipher based on reversible cellular automata," *New Generation Computing*, vol. 23, no. 3, pp. 245–258, 2005.
- [53] L. Mariot, "Cellular automata, Boolean functions and combinatorial designs," Dept. Autom. Control Eng., Univ. Côte d'Azur, Azur, France, Tech. Rep. NNT: 2018AZUR4011, 2018.
- [54] M. Szaban and F. Seredynski, "Cryptographically strong S-boxes based on cellular automata," in *Proc. 8th Int. Conf. Cellular Automata Res. Ind. (ACRI)*, Yokohama, Japan, Sep. 2008, pp. 478–485.
- [55] J. L. Schiff, *Introduction to Cellular Automata*, pp. 1–226. Accessed: Aug. 30, 2019. [Online]. Available: <https://www.psoup.math.wisc.edu>
- [56] *Tables of Cellular Automaton Properties*. Accessed: Aug. 30, 2019. [Online]. Available: <https://www.stephenwolfram.com>
- [57] S. Das and B. K. Skidar, "Analysis and synthesis of nonlinear reversible cellular automata in linear time," Nov. 2013, *arXiv:1311.6879*. Accessed: Aug. 30, 2019. [Online]. Available: <https://arxiv.org/abs/1311.6879>
- [58] S. Nandi, B. K. Kar, and P. P. Chaudhuri, "Theory and applications of cellular automata in cryptography," *IEEE Trans. Comput.*, vol. 43, no. 12, pp. 1346–1357, Dec. 1994.
- [59] P. P. Choudhury, B. K. Nayak, S. Sahoo, and S. P. Rath, "Theory and applications of two-dimensional, null-boundary, nine-neighborhood, cellular automata linear rules," Aug. 2008, *arXiv:0804.2346*. Accessed: Aug. 30, 2019. [Online]. Available: <https://arxiv.org/abs/0804.2346>
- [60] D. R. Nayak, P. K. Patra, and A. Muhapatra, "A survey on two dimensional cellular automata and its application in image processing," Jul. 2014, *arXiv:1407.7626*. Accessed: Aug. 30, 2019. [Online]. Available: <https://arxiv.org/abs/1407.7626>
- [61] V. Seenivasagam and R. Velumani, "A QR code based zero-watermarking scheme for authentication of medical images in teleradiology cloud," *Comp. Math. Methods Med.*, vol. 2013, Jun. 2013, Art. no. 516465.

- [62] S. Yokota. *QR Code Overview & Process of QR Code Applications*. pp. 1–50. Accessed: Aug. 30, 2019. [Online]. Available: <http://www.gs1jp.org>
- [63] C. Carlet, “Vectorial Boolean functions for cryptography,” in *Boolean Models and Methods in Mathematics, Computer Science, and Engineering*, vol. 134. London, U.K.: Cambridge Univ. Press, 2010, pp. 398–469.
- [64] D. K. Bhattacharyya and S. Nandi, “CA based password-only authenticated key exchange,” in *Proc. IEEE Workshop Signal Process. Syst. (SIPS)*, Lafayette, LA, USA, Oct. 2000, pp. 820–827.
- [65] L. Mariot and A. Leporati, “A cryptographic and coding-theoretic perspective on the global rules of cellular automata,” *Natural Comput.*, vol. 17, no. 3, pp. 487–498, 2018.
- [66] L. Mariot, S. Picek, A. Leporati, and D. Jakobovic, “Cellular automata-based S-boxes,” *Cryptogr. Commun.*, vol. 11, no. 1, pp. 41–62, 2019.
- [67] *What is a QR Code?* Accessed: Aug. 30, 2019. [Online]. Available: <https://www.keyence.com>
- [68] T. Toffoli and N. H. Margolus, “Invertible cellular automata: A review,” *Phys. D, Nonlinear Phenomena*, vol. 666, nos. 1–3, pp. 229–253, 1994.
- [69] A. M. Ilyasu, Roadmap to talking quantum movies: A contingent inquiry,” *IEEE Access*, vol. 7, pp. 23864–23913, 2019.
- [70] Y. Wu, J. P. Noonan, and S. Aghaian, “NPCR and UACI randomness tests for image encryption,” *J. Sel. Areas Telecommun.*, vol. 1, no. 2, pp. 31–38, 2011.
- [71] S. Geeetha, P. Punithavathi, A. M. Infanteena, and S. S. S. Sindhu, “A literature review on image encryption techniques,” *Int. J. Inf. Secur. Privacy*, vol. 2, no. 3, pp. 42–59, 2018.
- [72] *QR Code Masking*. Accessed: Oct. 1, 2019. [Online]. Available: <http://blog.kangaderoo.nl/2013/04/qr-code-masking.html>



**ABDULLAH M. ILIYASU** (aka Abdul M. Elias) received the M.E., Ph.D., and Dr. Eng. degrees in computational intelligence and intelligent systems engineering from the Tokyo Institute of Technology (Tokyo Tech.), Japan, where he is currently a Research Faculty with the School of Computing. Concurrently, he is also the Principal Investigator and Team Leader of the Advanced Computational Intelligence and Intelligent Systems Engineering (ACIISE) Research Group at the College of Engineering, Prince Sattam Bin Abdulaziz University (PSAU) in the Kingdom of Saudi Arabia. He is also a Professor with the School of Computer Science and Technology, Changchun University of Science and Technology, China. In addition to being among the pioneers of research in the emerging quantum image processing (QIP) subdiscipline, he has to his credit more than 100 publications traversing the areas of computational intelligence, quantum cybernetics, quantum image processing, quantum machine learning, cyber and information security, hybrid intelligent systems, health informatics, and electronics systems reliability. As well as being the Managing Editor of Fuji Technology Press, Japan, he is a member of editorial board of the *Journal of Advanced Computational Intelligence and Intelligent Informatics* (JACIII) and the *Quantum Reports Journal*. He is also an Associate Editor in many other journals, including *IEEE ACCESS* and *Journal of Medical Imaging and Health Informatics* (JMIHI).

• • •