Research article

Enhancing QR code security: Exploiting hidden message mechanisms and machine learning classification

Intelligent Decision Technologies
1–15

DOI: 10.1177/18724981241302039
journals.sagepub.com/home/idt

Mirsat Yeşiltepe[1], Muhammet Kurulay[1], Akram Bennour[2], Jawad Rasheed[3,4] and Shtwai Alsubai[5]

Abstract

The degree of utilization of Quick Response (QR) codes is sharply increasing due to the wide availability of smart devices. The primary purpose of the QR code is to ensure that an extensive message is fully transferred in a compact data format. Like any environment, security is an essential issue where QR codes are utilized. Such problems include the lack of signing information in a QR. This study aims to exploit the QR code hiding mechanism without spoiling the value of the code in the QR code while determining it using several machine learning algorithms. Consequently, several new QR image datasets are generated with varying sizes and variations to examine the classification of the proposed message-hiding scheme. This study used state-of-the-art models (VGG16, Xception) and a CNN-based model for QR code classification but only achieved 50% accuracy across four QR code dataset variants. Unsatisfied with these results, the study then employed the histogram feature density technique with various machine-learning (Logistic Regression (LR), Decision Tree (DT), and Random Forest (RF)) and deep learning (DL) models. The experimental results reveal that adapting the histogram density method in the proposed scheme for feature creation achieved an overall success rate of approximately 99.98%. Moreover, the study further aims to simulate single-layer QR codes from hackers' perspective that pretends to look like two-layer QR code systems. As a result of this simulation study, the performance was tested using different classification algorithms. In most cases, except for one, the DL model performed better by attaining a success rate above 90%.

Keywords

quick response, scale, VGG16, histogram, random forest

Received: 30 March 2024; accepted: 7 November 2024

1 Introduction

QR code is a two-dimensional barcode that transmits complete and without-defect data to the other party. Since this type of code is standardized, the content of the code is the same in all media and devices. In other words, the user who wants to transmit data creates a QR code with the data value within the capacity of the code and sends it to the other party. The degree or data contained in this code is accessed on the receiving end.

[1]Department of Mathematics Engineering, Yildiz Technical University, Istanbul, Turkey
[2]Laboratory of Mathematics, Informatics and Systems (LAMIS), Echahid Cheikh Larbi Tebessi University, Tebessa, Algeria
[3]Department of Computer Engineering, Istanbul Sabahattin Zaim University, Istanbul, Turkey
[4]Department of Software Engineering, Istanbul Nisantasi University, Istanbul, Turkey
[5]Department of Computer Science, College of Computer Engineering and Sciences in Al-Kharj, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia

Corresponding author:
Jawad Rasheed, Department of Computer Engineering, Istanbul Sabahattin Zaim University; Department of Software Engineering, Istanbul Nisantasi University, Istanbul, Turkey.
Email: jawad.rasheed@izu.edu.tr

Usually, a QR code is created from black patterns on a white background, but this is not always the case. Even if the originality of the code deteriorates to a certain extent (at varying rates according to the parameters of the code), it can be used with different background and pattern colors as long as it preserves the value. It has many usage areas, such as document verification,[1] presenting website link information for promotion,[2] and advertising various other details.

Generally, the primary purpose of the QR code detection method is to investigate whether the relevant image contains a QR code. Scientists exploited various methods, which can be divided into two groups: local features-based techniques and advanced feature extraction techniques such as convolutional neural networks (CNN).[3] For instance, authors in[4] used pixel similarity analysis with the help of filter matrices on the image. In,[5] researchers exploited local binary patterns to trace valuable features. Similarly, individuals in[6] utilized local gray pixel values as features by applying a median filter, binarization (converting to an image that only contains black and white pixels), and image expansion for better results. More local feature determination methods, such as scanner line features, Hough transformation, and statistical features, can be found in.[7]

On the other hand, methods based on CNN are also employed extensively, such as authors in[8] proposed a CNN-based architecture to determine the number of layers. Unlike prior studies, authors in[9] exploited CNN architecture to eliminate the angular noise of the QR code. Researchers in[10] incorporated an advanced DL-based model called Darknet19 as a transfer learning approach. Faster-RCNN, one of the newer CNN architectures, was used in.[11] Another study[12] used a local feature detection method called histogram density, aiming to convert images to grayscale and use histogram density values as features. Usually, this method was used to identify QR-coded images.

The primary purpose of two-layer QR codes is to display QR codes with open and hidden messages in a single QR code image. In the remaining part, studies on classification processes made on QR codes created for various purposes are included. In,[13] the authors performed identification verification by exploiting regional and global features using three-dimensional QR code images. They tested the resistance of these generated QR codes against various method attacks (image analysis with different camera types, distance between image and camera, ambient light, etc.). In general, they observed a success rate of 96%.

In,[14] the goal is not to copy the QR code obtained by adding the created image so that the original value in the second layer is not copied in the first layer. They examined the datasets in two groups, scanning by scanner and mobile phone. They used a customized CNN model for feature determination and DL as a classification algorithm. According to the study, false QR code classes and their subclasses were created, and the reliability of two-layer QR codes was observed to be 98% and above. The researchers in[15] developed a noise removal-based multi-layer QR code classification system. Their classification process aims to correctly classify and increase the length of the message that QR codes can contain. For testing, they did not read the QR codes in the virtual environment but scanned with a mobile phone in printed form. As a result of the classification process, the performance is over 97%, and the capacity increase is between 10% and 56%.

In,[16] the researchers classified the image as a simple QR code or an information-hidden QR code image by scanning the digital and print media. Besides classification, they aimed to increase the size of the hidden message. Their study employed that while the QR code is hidden within the QR code, the aim is not to spoil the value of the public-level QR code but to perceive it as a standard QR code. They achieved a success rate is 100% in the digital environment and over 98% in the printed environment. In,[17] the researchers classified the original and fake data matrices in different environments (mobile environments, scanners, etc.). They first extracted features in the spatial and frequency domain and later applied Support Vector Machine and other DL methods to these global features for classification. The difference from the main work is that the fake QR code class is used instead of the two-level QR code. As observed in the literature, for multi-layer QR code classification, most studies examine QR codes generally of two-level. Rarely, three-layer[18] or more are studied. Scientists in,[19] emphasized that a fake file can be produced with a steganographic picture, thus the other party cannot understand. However, a man-in-the-middle attack may obtain the actual documents. They used machine learning methods such as CNN to determine whether the file was steganographic.

To the best of our knowledge, the prior works discussed in the literature do not address two-layer QR code detection. As modern two-layer QR codes contain images and hidden data in their layers, most of the advanced DL methods used previously fail to classify the codes and retrieve the data correctly. Therefore, this study contributes the following to the scientific research:

- Created several new QR image datasets of varying sizes and variations for examining the classification capabilities of the proposed message-hiding scheme.
- For better evaluation, the study creates each image in a dataset with different scaling parameters, including version, scale, code mode, and error correction level.
- Exploits modern DL-based pre-trained models to classify QR codes as two-layer and single-layer codes
- Compared the performance of the simulated QR codes using various classification algorithms.

- Adapted the histogram density method for feature creation within the proposed scheme to impressively improve the overall success rate of approximately 99.98% in the experimental results.
- Conducted a simulation study from hackers' perspective by simulating single-layer QR codes designed to mimic the appearance of two-layer QR code systems.
- Provides insight into how the histogram density method, along with machine learning techniques, can be leveraged for improved classification and security measures.
- The study contributes to understanding and enhancing QR code security by exploring message-hiding schemes and potential vulnerabilities.

The rest of the paper is structured as follows: Section 2 details the creation of a two-layer QR code dataset and elaborates on various machine learning and DL algorithms. In contrast, Section 3 outlines different test scenarios along with obtained results. Section 4 briefly discusses the obtained results whereas Section 5 concludes the study.

2 Materials and methods

This study creates a new dataset using four scaling parameters described in this section. Later, to obtain useful features, it exploits two feature extraction techniques, including CNN and histogram density. Once features are extracted, the study trains and tests three different machine learning (RF, DT, and LR) and a DL model (based on two hidden layers) to classify two-layer QR code images. Besides these, the scheme is then evaluated based on the attacker's activities (see Figure 1). The attacker understood that communication could not be done with a QR code with normal standards by looking at the histogram density values of the QR code images used in communication. From the attacker's point of view, the gray colors next to the black and white color values of different normal QR codes were used. Thus, he tried to produce similar QR codes using different methods so that the system would think that these were its own codes, and he wanted to keep the system busy. The system checks the codes produced as per its own algorithm according to the classification model. Tests are carried out to see if the codes produced by the attacker can be understood by these algorithms.

2.1 Dataset creation

As an aside, a QR code containing the same data may look different depending on its generator by setting four scaling parameters, including *version*, *scale*, *code mode*, and *error correction level*. To examine the effect of four scaling attributes in QR code formation, a single scaling parametric value is changed at a time while keeping others fixed (default). The content contained in all QR codes to be examined in this section is "yildiz".

The *version* parameter controls the standard size and width of the QR code, which ranges between 1 to 40. Figure 2 shows different versions of the QR code, all having the same content but different sizes and shapes. It is evident in Figure 2 that the standard size and width of the code increase as the version value of the QR code increases. Therefore, for higher efficiency, it is suggested to use the smallest possible *version* for fixed content, as the larger version value increases the size of the QR code which eventually results in high transmission time.

Figure 3 depicts the effect of the *scale* parameter in QR code formation. The contribution of the scale value makes the QR code appearance larger, like the version parameter, but it does not increase the capacity at scale, and shapes also remain unchanged. In the *version*, the character length of the content (value) it contains increases with an increase/change in the structure of the code.

Figure 4 shows QR codes with different code mode parameters. When the mode is set to Alpha Numeric, Byte, or Numeric, the QR codes have different visual representations while code sizes remain the same. This may cause different values to be obtained by scanning via various QR code readers. To avoid this situation, it is considered better to use *Alpha Numeric* mode because of its default setting in widely used QR code readers.

Figure 5 illustrates four different *error correction level* parametric effects in QR code formation. Like others, it only changes the QR code's visual appearance, with each one adding different amounts of backup data depending on how much damage the QR code is expected to suffer in its intended environment and, hence, how much error correction may be required. The QR code becomes less dense as the error correction decreases from Level H to L even though it contains the same data/information (as a percentage: H = 30, Q = 25, M = 15, L = 7). A higher error correction level can make the QR code sustain more damage, but results in a larger number of columns and rows of modules required to store the original data plus the increasing amount of backup code words.

Similar to image steganography, a two-layer QR code is a normal QR code that hides another QR code. In such QR codes, the upper layer contains public information. In contrast, the lower layer (hidden layer) encompasses content inaccessible by the standard QR code reader, such as the confidential information verification code.[20] It contains information that should not be obtained by anyone, such as a secret message[6,21,22.]

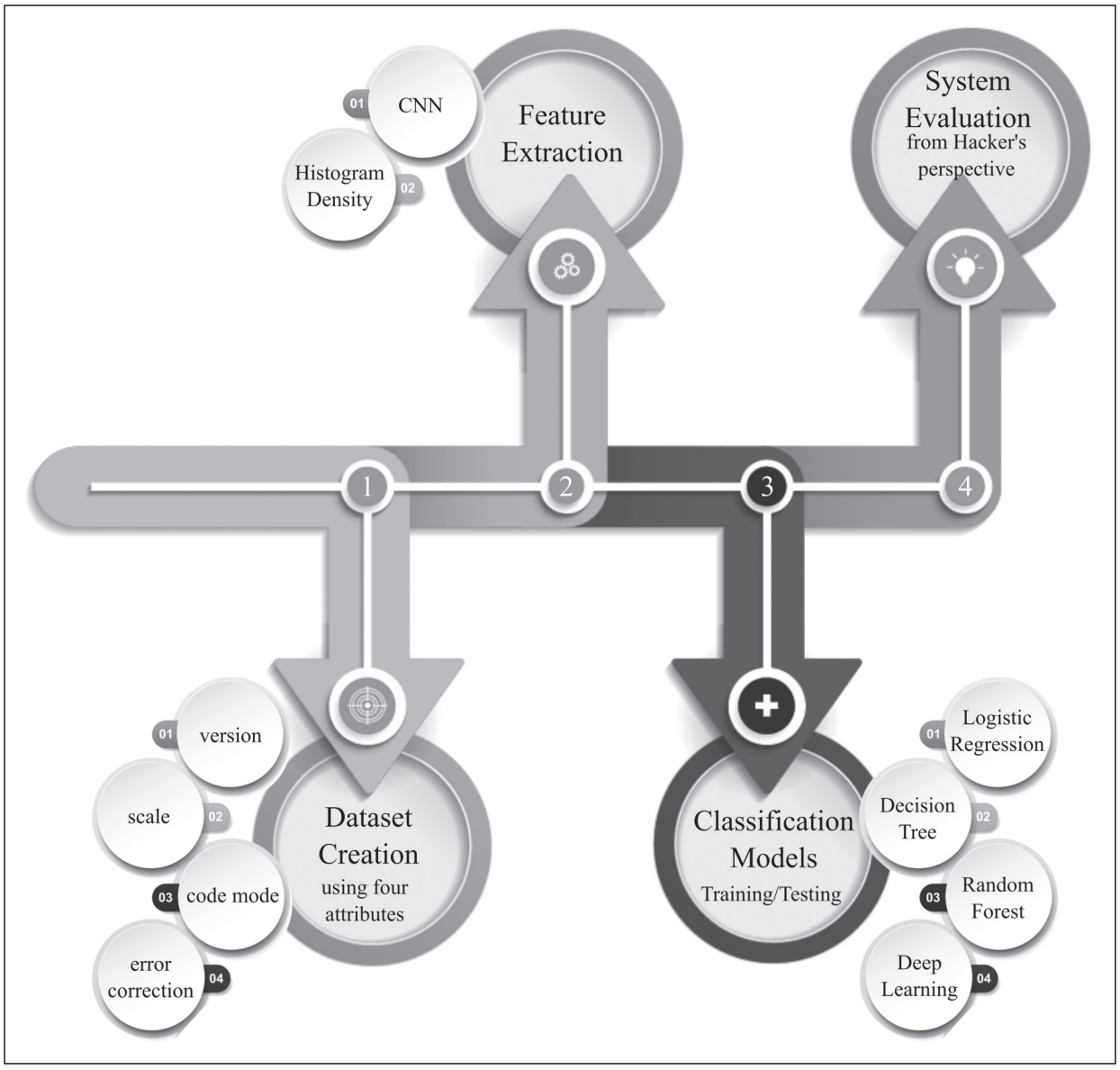

Figure 1. The workflow of the proposed scheme.

Figure 6 shows a sample of a two-layer QR code. The value (data) of the two-layer QR code is the same as that of a single-layer code. The reverse layer image shows the recovered QR code (hidden layer) contained in the two-level QR code, with preserved hidden layer value. Generally, QR codes are on a white background, but the reverse layer usually has a light gray background, which is generally not visible to users.

The following steps are involved in the creation of a two-layer QR codes dataset:

1. QR codes with features are created according to the relevant data set rules (variable or fixed scale and version parameters) for the formation of the hidden (lower) and public (upper) layers.
2. If the height and width dimensions of the QR codes as images are not equal, the height and width size of the image of the general QR code is equalized with the dimensions of the lower and upper ones.
3. The hidden bit value is determined and set to 1 in the test processes. The shift bit value is calculated by subtracting the hidden bit value from 8. This value was used as 7 in the test.
4. Visible and hidden mask values are calculated. The visible mask value 0XFF is found by shifting the hidden bit value 1 bit to the left, whereas the hidden mask value is found by shifting the 0XFF value with the shift of bit value, that is, by shifting 7 bits to the right.

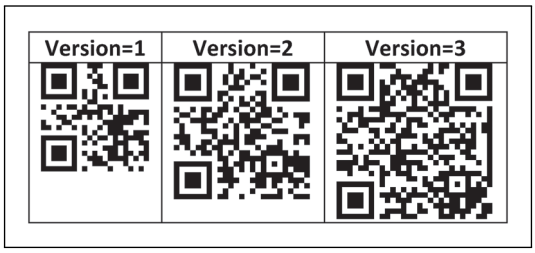

Figure 2. Samples of QR codes having the same content but different versions.

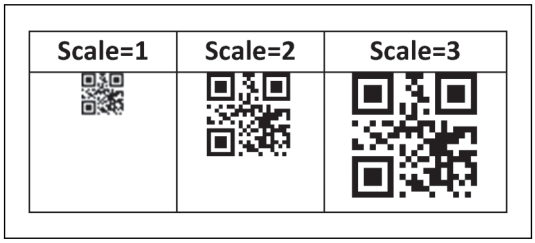

Figure 3. Samples of QR codes having the same content but different scales.

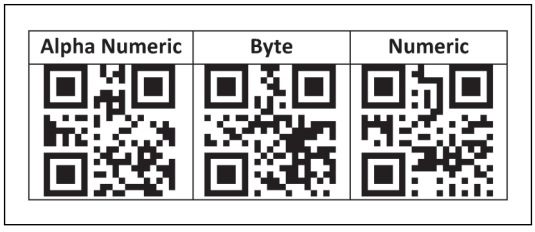

Figure 4. Samples of QR codes having the same content but different code modes.

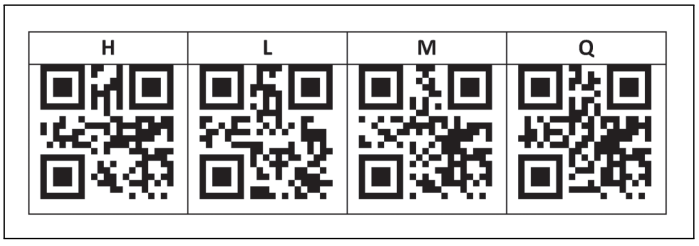

Figure 5. Samples of QR codes having the same content but different error correction levels.

5. The operations in this section are repeated as the general QR image height and width dimensions. The corresponding bit of the visible and hidden image is read. For the red, green, and blue channels of the relevant pixel, the following operation occurs sequentially. The corresponding channel value of the visible bit value and the visible mask value are treated with il and (and). The corresponding channel value of the hidden bit value is shifted to the right by the shift bit and is subjected to the hidden mask and operation with this value. The last found value is added to the previously found values.

2.2 Algorithms and models for classification

For this purpose, methods such as CNN and global and local feature extraction methods are used. The main purpose of CNN methods is to obtain important representation from the image under examination by utilizing various filters and using them as features. For initial analysis, ready models were exploited to the images, and the features were obtained rather than the customized CNN.

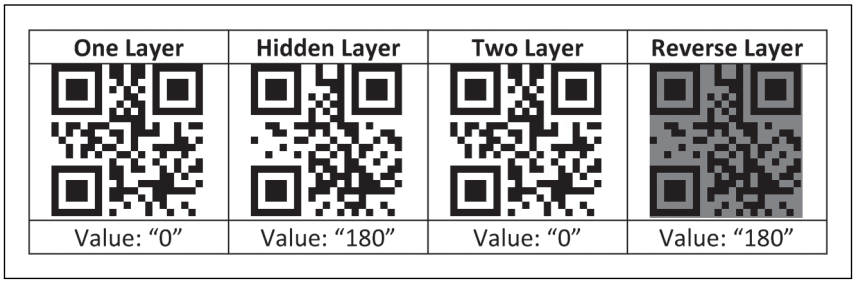

Figure 6. Example of two-level QR code creation for the data set.

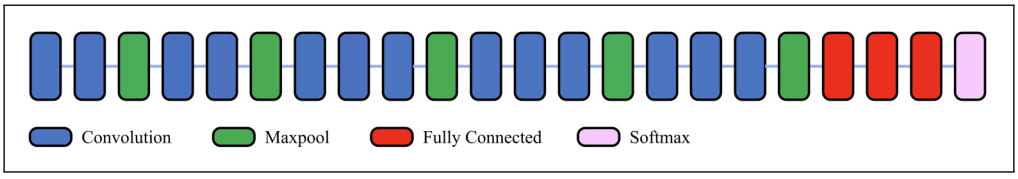

Figure 7. VGG16 model architecture.[23]

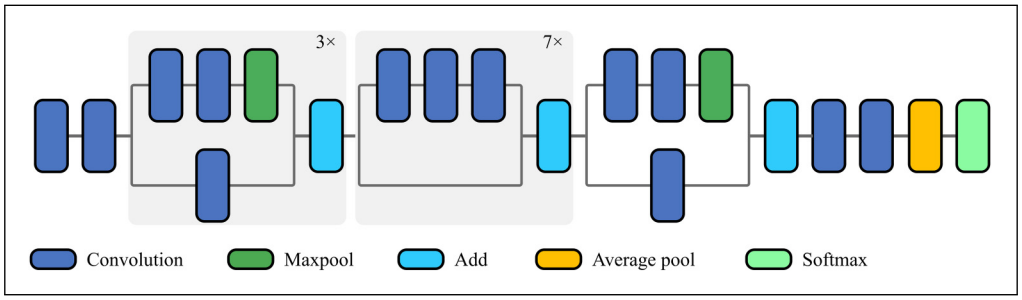

Figure 8. Xception model architecture.[23]

	Pixels							
	248	249	250	251	252	253	254	255
0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.496329
1	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.514395
2	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.493439
3	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.503555
4	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.496329

Figure 9. Example of histogram density feature extraction dataset.

This study employed CNN-based pre-trained models since models such as VGG16 and Xception are ready and designed specifically for transfer learning (Figure 7, Figure 8). The VGG16 model[15] accepts pixel values of colored images as input and performs 3×3 convolution operations using several layers having 64, 128, 256, and 512 filters. Moreover, it performs 3×3 maximum pooling (extracting only the value with the highest pixel value of the region that the filter is interested in) and the 'softmax' function at the output. It is a model that aims to achieve value within the XCeption model[16]; although operations are performed on various layers (but by returning some layer values in between), the operations are performed over one color channel, not 3 color channels.

In the histogram density feature extraction method, the histogram density values in the gray-scaled images are used as features. As a feature, the ratio of the number of pixels with each gray tone to the total number is used as a value. Thus, it provides 256 features for each image of varying sizes, as shown in Figure 9.

Attribute	Coefficient	Standard Coefficient	Standard Error	z-Value	p-Value
att1	-0.872	-0.092	0.527	-1.653	0.098
att2	-3646.506	-3.248	1013.465	-3.598	0.000
att3	7462.014	5.998	2659.748	2.806	0.005
att4	-2662.217	-2.098	1619.117	-1.644	0.100
Intercept	0.074	0.283	0.062	1.198	0.231

Figure 10. Logistic regression model example.

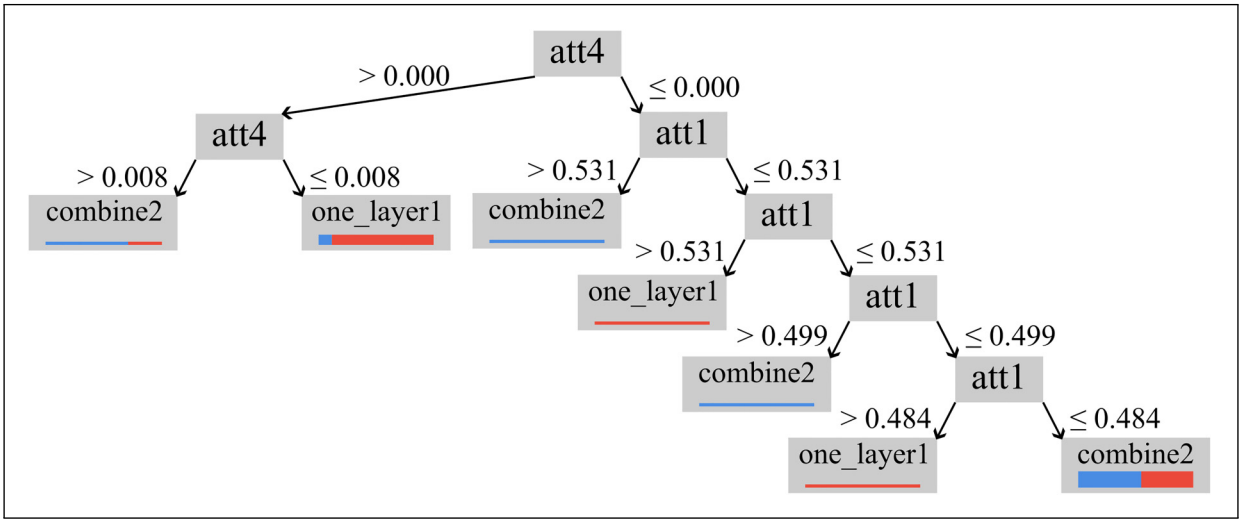

Figure 11. Decision tree model example.

The purpose of the classification process is to predict whether a QR code is 2-layer or 1-layer. For this, this study exploited various machine learning algorithms and DL approaches. The following section outlines the brief workings of these algorithms.

LR is an algorithm that aims to separate classes with a curve, as in linear regression. While linear regression uses a straight line for this curve, LR targets to draw a smoother curve between data points to separate classes using a logit function curve (S-shaped). An equation is produced as a model (see Figure 10). In this way, the created model can be integrated faster than other algorithms, as a model transfers in a different environment or by transmitting it to codes without transfer.

DT is a classification method that creates a model in the form of a tree structure consisting of decision nodes and leaf nodes according to features and targets.[24] The fundamental feature of DR is that a single tree can represent the created model. Thus, it can be expressed as a tree; therefore, at the test stage, the relevant tree nodes are traversed to determine the class of the test sample (see Figure 11). After several experiments and different combinations, for this study, the maximum depth of the DT is set to 10, with pruning activated, while the criterion is set to *gain ratio*. Moreover, it has a confidence of 0.1 with pre-pruning activated. The minimal gain and minimal leaf size are set to 0.01 and 2, respectively.

RF resembles the DT algorithm in terms of the models it creates; however, it creates a multi-tree instead of a single tree (see Figure 12). RF is preferred, especially when the other model created with a single tree does not perform well for the problem at hand. For this study, we repeated experiments with various RF parameters. However, the RF that has 100 number of trees, a gain ratio criterion, and a depth of 10 with the confidence voting strategy performed best than other exploited RF models.

The DL algorithm is usually preferred more than other algorithms due to the increased processing power of computers. Generally, it is an artificial neural network with hidden layer(s). Its strength is that these hidden layers can vary and possess the desirable number of neurons to handle complex classification problems. For this study, various DL models were exploited. However, the model with two hidden layers, each with the size of 50×50, having a rectifier as an activation function and an automatic loss function, performed better than others. The suggested model is trained over 10 epochs.

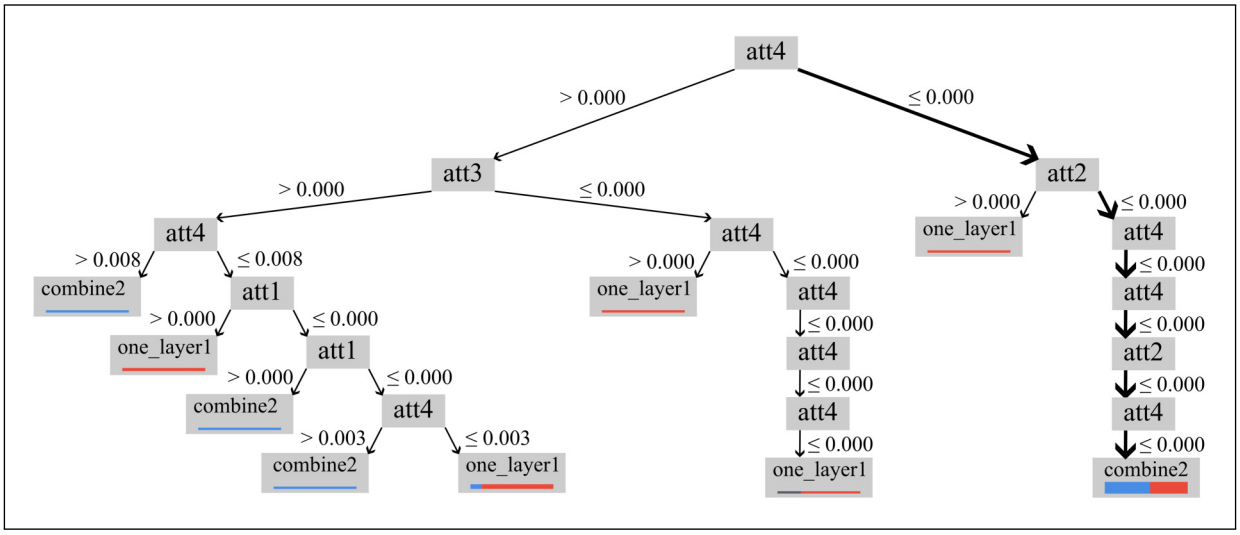

Figure 12. Random forest model example.

Various parameters measure the classification result in the classification process. The most common of these is the truth value. Therefore, performance is analyzed in terms of accuracy against each class by dividing the correctly predicted samples by the total number of samples. Since the sample numbers of the classes examined in the article are equal, no other evaluation criterion is used.

3 Test environment

This section mentions the steps for creating the test environment, which are generally shown below.

- **Creating the main datasets to be examined:** 4 main datasets were created for this step. The characteristics of the created master datasets are as follows: 1 - with the same version and scale values, 2- with the same version value but with different scale values, 3 - with different version values but with the same scale values, and 4 - with QR codes with different versions and scale values generated datasets. The values of the QR codes are also composed of random words with increasing length according to the version value in the selected condition. A random value between 1 and 10 was used as the same value, and 5 as the different value. In all the main data sets, the default values of the QR codes and other features of the studied library were used. There are 2 classes in each created data set, single and double-layer QR codes. Each class contains 50,000 images. By checking the content values of all QR codes created in all classes, it was tested whether the QR codes are valid QR codes and whether both the upper and lower QR codes are valid QR codes in double-layered QR codes.
- **Feature extraction from the images to be tested:** For this step, this study initially exploited ready-made Keras models such as VGG and Xception. However, later, the histogram density feature selection method was applied for better accomplishment.
- **Division of each main dataset into training and testing:** In the CNN method, it is divided into 60% training, 20% testing, and 20% validation. In the other method, the training and test sets are divided by 70% and 30%, respectively, in each main data set.
- **Application of classification processes:** In the classification process, pre-defined DL models were employed in the feature determination step along with CNN of 3 layers of size 512 for classification. The Keras library was used in the CNN directional classification process. On the other hand, with the histogram feature-specific method as a feature selection, LR, DT, RF, and DL classification algorithms were used.
- **Evaluation of classification processes:** This study marks accuracy as the evaluation criteria for better understanding and distinguishing. The results are analyzed in terms of attacker and defender. The roles of the attacker and defender are as follows.

For this study, the following scenario was created regarding the attacker and the defender. Let an organization use two-layer QR code transmission in an insecure environment for message transmission. The attacker first works with machine

Table 1. The performance comparison of several convolutional neural network-based state-of-the-art models to classify the QR codes.

	Accuracy (%) of model	
Dataset	VGG16	Xception
1	50.00	50.00
2	49.99	50.00
3	50.02	50.00
4	50.00	50.00

Table 2. The performance comparison of several classification models when exploiting the histogram density feature method to classify QR codes.

	Accuracy (%) of classification model			
Dataset	Logistic regression	Decision tree	Ramdon forest	DL-based neural network
1	99.98	99.98	99.98	99.98
2	99.29	99.98	99.98	99.98
3	99.98	99.99	99.99	99.98
4	99.99	99.99	99.99	99.99

learning to understand whether the QR code used by the system is original or steganographic. The defender does not want this situation to occur. In other words, the attacker wants the performance to be high, and the defender wants it to be low.

3.1 Experimental results for classification of layers based qr code

This section compares the classification results of two feature extraction techniques, including the CNN-based method and the histogram feature method. The experimented results of exploited classification models to distinguish the QR code based on the number of layers for all the four sub-datasets created with different parametric values are evaluated in terms of accuracy scale.

For QR code classification, this study first incorporated state-of-the-art models (VGG16 and Xception) along with a CNN-based model. Even though we exploited various combinations of CNN layers, we only succeeded in achieving an accuracy of 50% for all four different variants of QR code datasets. Table 1 shows the accuracy achieved by the CNN model for VGG16 and Xception nets. It is evident from Table 1 that the models achieved unsatisfactory results even when the dataset has an equal number of samples for each class. Thus, next, this study exploits the histogram feature density technique along with various machine learning classification algorithms to secure better results.

This study extends the experiments by utilizing the histogram feature extraction technique. These extracted features are later fed to LR, DT, RF, and DL-based neural network models. Table 2 outlines the performance of these four classification models when trained and tested over features extracted by the histogram density feature technique. It is notable from Table 2 that all suggested models performed well and attained an accuracy of 99.98% or above, except for LR, which secured 99.29% when trained with a second sub-dataset (QR codes with the same version value but with different scale values). In other words, in this case, the defender will be able to use the system's different classification algorithm for QR codes with different features or with the same feature that has completed the process for this condition. In the same situation, the attacker tried to produce a new version of QR codes to reduce the system's success. For a better understanding of the types of main and sub-datasets created, readers are encouraged to read Section 2.1 and the beginning of Section 3.

When the tables are examined for the attacker and the defender, the attacker was successful with the histogram density feature extraction method. In other words, it understands that the transmitted code in all datasets is not the original (single-layer) QR code.

3.2 Systems evaluation from hacker's/defender's perspective

The attacker understood that the QR code used in the transmission (described in the previous section) was a special QR code. As the next step, the attacker examined the histogram intensity values of the original and system QR codes (see Figure 13). The attacker determined that single layers have only 0th and 255th gray intensity values, while bilayers have 0th, 1st, and 254th, 255th gray intensity values.

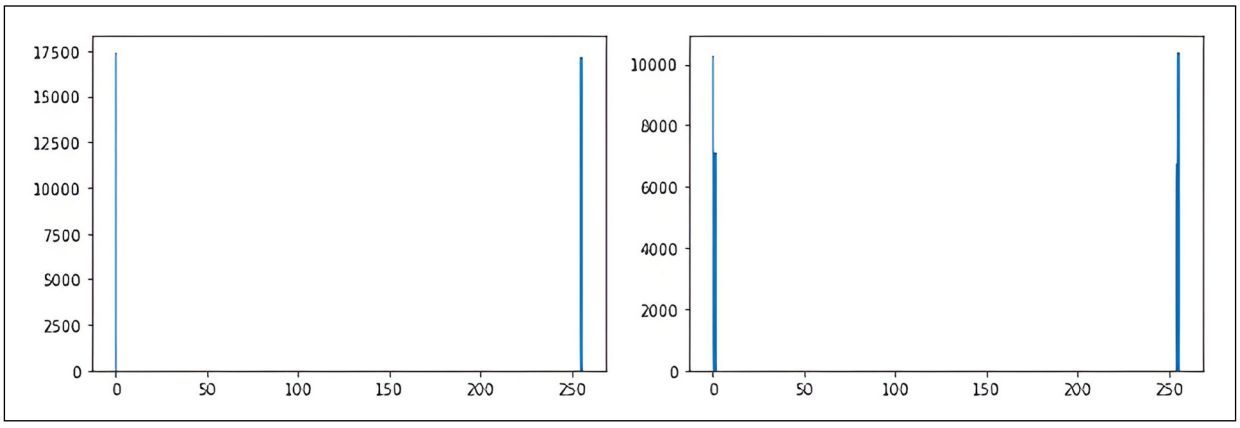

Figure 13. Example of single-layer (left) and two-layer (right) QR code histogram chart.

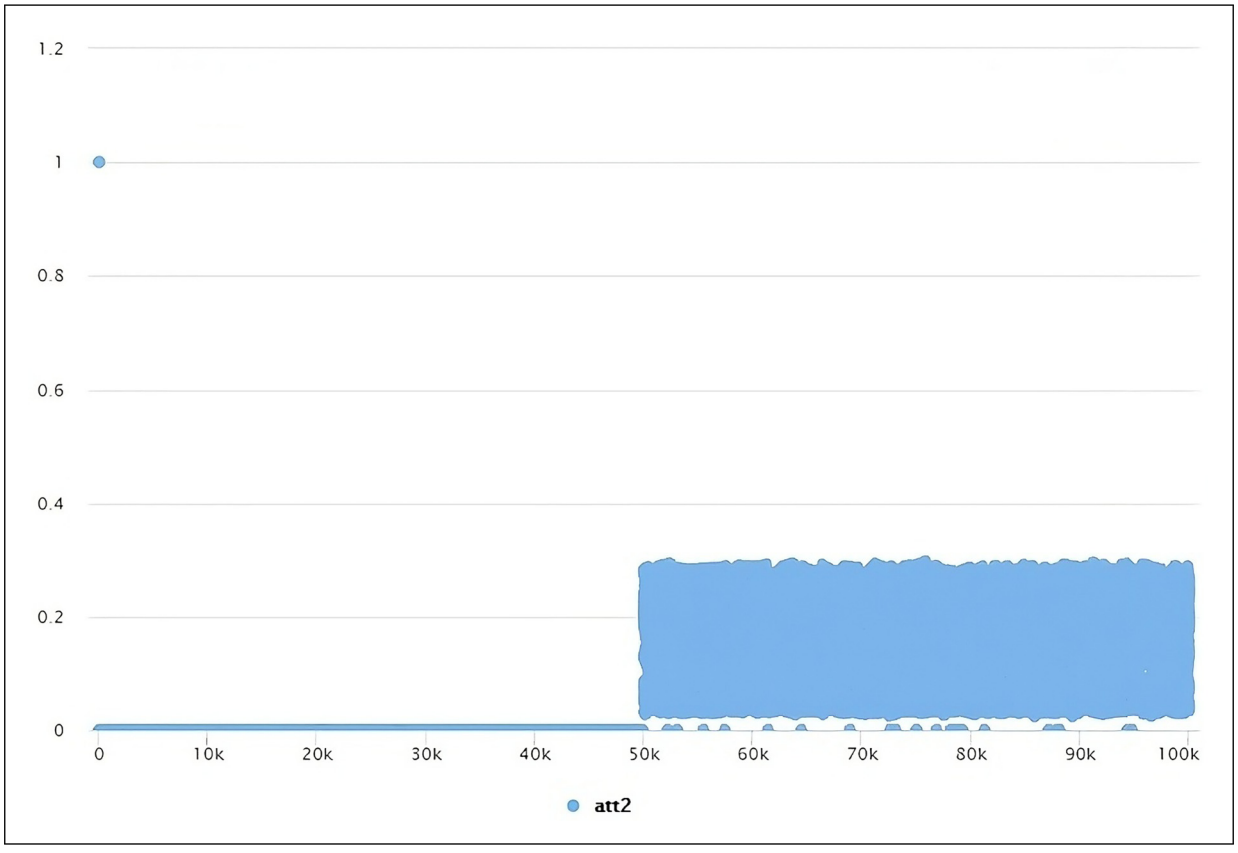

Figure 14. For the first data set gray level density graph (left Side one layer–right Side two layer QR code).

In the next step, the attacker wants to generate QR codes that are close to the type used by the system. For this, firstly, QR code strategies to be produced for the main four data sets have been determined so that the QR codes have the 1st and 256th gray levels. For this step, as an example, the distribution of the gray levels of the intensities of the QR codes used by the system, as in Figure 14, was examined. As listed in Tables 3 to 6, the arithmetic mean, minimum value, maximum value, standard deviation, and visual limit value of the intensity values were used as offensive examination parameters. The visual limit value and the limits where the values have observed high intensity were determined as the observation value. For example, this situation is determined as 0.05 and 0.25 in Figure 14. The parameter determination process has been determined for all main data sets specific to those sets. The main purpose of this step is to successfully generate the histogram density values of the fake system QR codes to be produced (see Figure 15).

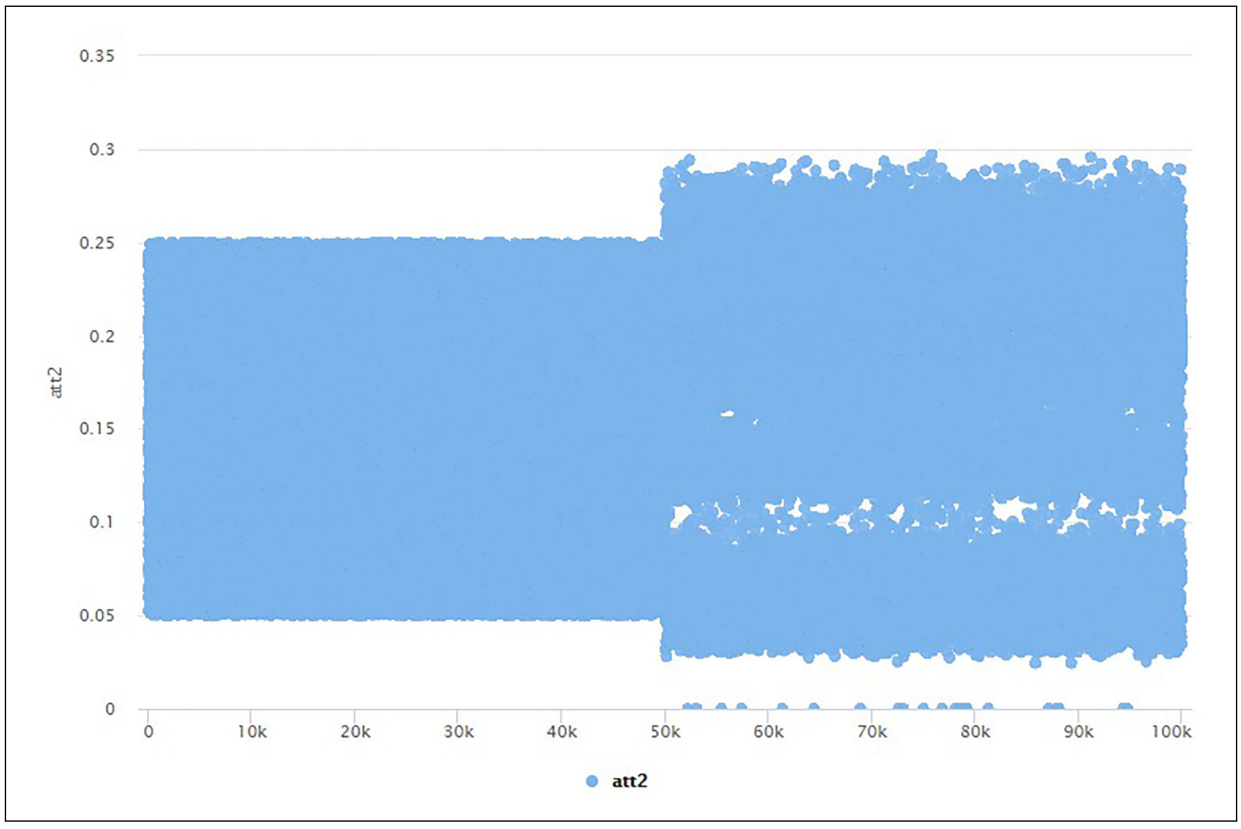

Figure 15. For the first v5 data set gray level density graph (left Side one layer–right Side two layer QR code).

Table 3. The density properties of the dataset containing QR codes having the same version and the same scale.

	Attribute				
Pixel	Mean	Minimum	Maximum	Standard deviation	Visual limit
0	0.312208	0.198722	0.512897	0.052389	0.25–0.45
1	0.184821	0	0.296999	0.053299	0.05–0.25
254	0.184841	0	0.296277	0.053333	0.05–0.25
255	0.318249	0.203665	0.520175	0.054280	0.25–0.45

Table 4. The density properties of the dataset containing QR codes having the same version but varied scale.

	Attribute				
Pixel	Mean	Minimum	Maximum	Standard deviation	Visual limit
0	0.265815	0.181610	0.518077	0.033734	0.25–0.35
1	0.235511	0	0.316227	0.033602	0.15–0.25
254	0.235344	0	0.319054	0.033653	0.15–0.25
255	0.263327	0.190058	0.523066	0.033527	0.25–0.35

Table 5. The density properties of the dataset containing QR codes having the varied version but same scale.

	Attribute				
Pixel	Mean	Minimum	Maximum	Standard deviation	Visual limit
0	0.305911	0.201444	0.502225	0.0479778	0.25–0.4
1	0.191927	0	0.295670	0.048967	0.05–0.25
254	0.191142	0	0.295842	0.048614	0.05–0.25
255	0.311019	0.19621	0.520778	0.049644	0.25–0.4

Table 6. The density properties of the dataset containing QR codes having the varied version and varied scales.

	Attribute				
Pixel	Mean	Minimum	Maximum	Standard deviation	Visual limit
0	0.303148	0.198030	0.498496	0.048327	0.25–0.40
1	0.195005	0	0.303811	0.049393	0.05–0.25
254	0.194225	0	0.300047	0.049228	0.05–0.25
255	0.307620	0.198639	0.521137	0.050359	0.25–0.40

Table 7. The performance comparison of several classification models to classify QR codes with the same version and scale.

	Accuracy (%) of classification model			
Dataset version	Logistic regression	Decision tree	Random forest	DL-based neural network
1	99.98	99.98	99.98	99.98
1-v2	94.88	97.14	96.95	99.03
1-v3	62.99	100.00	96.89	75.63
1-v4	68.20	75.54	75.98	98.54
1-v5	69.58	75.47	76.83	87.48
1-v6	55.46	73.35	72.85	95.73

In the next step, the attacker wanted to generate QR codes that would not be noticed in transmission by the above parameters. The attacker will create five datasets. The definitions of these data sets are as follows: The following explanation describes how to pass the value of the first attribute to the first two attributes. The exact process is done for the last attribute's value transfer to the last two attributes.

v2: The value of the first feature is randomly distributed to the first two features.

v3: The average of the 2nd feature values of the double-layered QR codes was examined. The first feature value is randomly distributed to the first and second features up to the maximum mean value of the 2nd feature. When the mean value is exceeded, the remaining value is distributed over the first feature.

v4: The difference from v3 of the minimum and maximum values instead of the mean.

A random value is chosen from

v5: It is not sensitive to set the maximum and minimum values different from v4; a certain value is selected, and those values are determined. For example, the value of 0.27954 is set to 0.25. While determining this value, the value distribution was examined, and the maximum and minimum values of the area where the values were clustered were selected. In this way, it is desired to avoid contradictory situations.

v6: In this method, the maximum, minimum, mean, and standard deviation values of the 2nd feature values are used. With these values, random values were taken over the normal curve, which became the maximum value for the 2nd feature. The aim here is to ensure that the values to be selected are the closest to the real values, as the objective is to select the frequencies that are seen more frequently and less frequently in the data.

The test results are explained as follows. In this section, the acceptable accuracy value is determined as 95% and above. Between 95% and 75%, the situation that needs improvement is accepted as six worse situations. For the attacker, the accuracy values in these tables are expected to be low (because it is not noticed by the system), while for the defender, these values are expected to be high (because the system notices it). Among these values, the value is used as the maximum distribution value for that sample in the value distribution.

When Table 7 is examined, it is seen that the algorithm that performs best is DL. From the attacker's point of view, the v6 state has the best. In this case, only DL secured a good response, while LR performed worst. The success of the models produced by DT and RF is less than usable. LR has been observed to adapt worse to all versions. Because the sizes of both the upper and lower QR codes of the data sets in this table are equal. The fact that LR is the most suitable algorithm for linear problems has become a disadvantage in suitable attacks. DT and RF models can be used for v2 and v3 cases, however, not for other cases.

When Table 8 is examined, it is observed that LR performed better on this dataset (same version but variable scale) than the previously discussed dataset (same version, same scale). However, it still couldn't outperform the other models and its performance even degraded for the v3 case (worst case). On the other hand, the DL-based model outperformed other models (LR, DT, RF) for all versions of the dataset (v2, v4, v5, and v6) by maintaining an accuracy of 92.76% or

Table 8. The performance comparison of several classification models to classify QR codes with the same version but varied scale.

	Accuracy (%) of classification model			
Dataset version	Logistic regression	Decision tree	Random forest	DL-based neural network
2	99.29	99.98	99.98	99.98
2-v2	82.47	81.38	81.57	92.76
2-v3	52.75	99.74	97.78	84.80
2-v4	78.04	79.40	79.31	93.47
2-v5	80.37	82.61	84.08	96.01
2-v6	82.49	81.63	82.55	92.85

Table 9. The performance comparison of several classification models to classify QR codes with varied versions but the same scale.

	Accuracy (%) of classification model			
Dataset version	Logistic regression	Decision tree	Random forest	DL-based neural network
3	99.98	99.99	99.99	99.98
3-v2	95.16	97.06	96.67	98.9
3-v3	45.69	99.99	98.70	99.71
3-v4	68.82	76.81	76.59	97.36
3-v5	71.21	74.16	76.32	95
3-v6	95.15	97.00	97.11	99.03

Table 10. The performance comparison of several classification models to classify QR codes with the varied version scale.

	Accuracy (%) of classification model			
Dataset version	Logistic regression	Decision tree	Random forest	DL-based neural network
4	99.99	99.99	99.99	99.99
4-v2	93.55	95.59	95.69	98.66
4-v3	57.39	99.94	97.53	50.32
4-v4	69.78	75.84	76.10	97.66
4-v5	72.27	73.98	74.60	97.7
4-v6	93.56	95.69	95.81	98.74

above, but it performed average for v3 dataset with an accuracy of 84.80% only. For v3, DT secured the best accuracy of 99.74%.

Similarly, Table 9 notes that the suggested DL-based model achieved the highest accuracies for all versions (v2, v3, v4, v5, and v6) of the dataset having QR codes with variable versions but the same scale. DL-based model attained a minimum accuracy of 95% for v5, while for other versions, the accuracies are better. For v5, other exploited models (LR, DT, and RF) hardly secured accuracies between 71% and 77%. For the dataset containing a QR code with variable version and variable scale, the performance of all models varies. For v3, DT outperformed other models, however, for other versions DL-based model performed well. Similarly Table 10 presents results with varied version scale.

Of the data sets, v4 and v5 were observed as the most effective methods to reduce the attacker's classification accuracy. In other words, working with minimum and maximum values as a tabular or visual boundary has been the most aggressive method. As a result, a solution was found for the defender by changing to a better classification algorithm in each case.

4 Discussion

It has been observed that the CNN feature generation method is not efficient in the classification of layer-based QR codes, while the histogram density feature extraction method is effective. In the histogram density feature extraction method, the suggested DL-based neural network model has the highest accuracy value in most of the cases. It is noted that the most defensive classification algorithm against sub-datasets in various versions is LR due to its more effectiveness against linear classification, and because the attack is linear simulation, its defense against attacks is lower. The results of the DT and RF algorithms are close to each other. Among them, a small amount of accuracy is generally in favor of RF. If the dimensions of the hidden and open QR codes are the same, the histogram density feature selection and DL model perform better than others.

The most general information extracted; The more similar the features of the QR codes to be examined, the more likely the attacker will achieve his goal. It is recommended to work with QR codes with variable features in the systems studied. The same results may not be obtained with the two-layer QR codes used by all the different systems obtained from this study. In other words, two-layer QR codes created with different algorithms can be successfully classified by the CNN method, or different results can be obtained.

5 Conclusion

In conclusion, the escalating utilization of QR codes propelled by the widespread availability of smart devices underscores the importance of exploring their security vulnerabilities. This study delved into the intricacies of QR code encryption and decryption, aiming to develop a message-hiding mechanism while preserving the code's integrity. Through the utilization of various machine learning algorithms and the creation of diverse QR image datasets, the efficacy of the proposed scheme was thoroughly assessed. Impressively, adapting the histogram density method with a DL model yielded a remarkable success rate of approximately 99.98%, showcasing the potential of this approach in fortifying QR code security. Additionally, the simulation of single-layer QR codes mimicking more complex systems provided valuable insights into potential vulnerabilities, with the DL model demonstrating notable effectiveness in classification tasks. These findings underscore the importance of continued research and innovation in securing QR code technology as it permeates various aspects of modern life. However, there is still a need for work to classify the QR code images with a single dataset that contains more versions of QR codes having different parametric values.

ORCID iDs

Muhammet Kurulay https://orcid.org/0000-0002-9276-9989
Jawad Rasheed https://orcid.org/0000-0003-3761-1641

Statements and declarations

Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

Declaration of conflicting interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Dataset availability

The dataset will be provided on request.

References

1. Wibiyanto A and Afrianto I. QR Code and transport layer security for licensing documents verification. *IOP Conf Series: Mater Sci Eng* [Internet] 2018; 407: 012069. Available from: https://iopscience.iop.org/article/10.1088/1757-899X/407/1/012069.
2. Vazquez-Briseno M, F I, Sanchez-Lopez D, et al. Using RFID/NFC and QR-Code in Mobile Phones to Link the Physical and the Digital World. In: *Interactive Multimedia [Internet]*. InTech; 2012. pp.219–242. Available from: http://www.intechopen.com/books/interactive-multimedia/using-rfid-nfc-and-qr-code-in-mobile-phones-to-link-the-physical-and-the-digital-world. https://doi.org/10.5772/37447
3. Jain V, Jain Y, Dhingra H, et al. A systematic literature review on qr code detection and pre-processing. *Int J on Tech Phys Prob of Eng* 2021; 13: 111–119.
4. Lopez-Rincon O, Starostenko O, Alarcon-Aquino V, et al. Binary large object-based approach for QR code detection in uncontrolled environments. *J of Electr Comput Eng* [Internet] 2017; 2017: 1–15. Available from: https://www.hindawi.com/journals/jece/2017/4613628/.
5. Tong L, Gu X and Dai F. QR Code detection based on local features. In: Proceedings of International Conference on Internet Multimedia Computing and Service [Internet]. New York, NY, USA: ACM; 2014. pp.319–322. Available from: https://dl.acm.org/doi/10.1145/2632856.2632860
6. Yuan T, Wang Y, Xu K, et al. Two-Layer QR codes. *IEEE Trans on Image Proces* [Internet] 2019; 28: 4413–4428. Available from: https://ieeexplore.ieee.org/document/8709989/.
7. Bodnár P. *Image analysis methods for localization of visual codes* [Internet]. Szeged: Szegedi Tudományegyetem; 2016. Available from: http://doktori.bibl.u-szeged.hu/2825/
8. Chou T-H, Ho C-S and Kuo Y-F. QR Code detection using convolutional neural networks. In: 2015 International Conference on Advanced Robotics and Intelligent Systems (ARIS) [Internet]. IEEE; 2015. pp.1–5. Available from: http://ieeexplore.ieee.org/document/7158354/

9. Kurniawan WC, Okumura H, Muladi , et al. An improvement on QR code limit angle detection using convolution neural network. In: 2019 International Conference on Electrical, Electronics and Information Engineering (ICEEIE) [Internet]. IEEE; 2019. pp.234–238. Available from: https://ieeexplore.ieee.org/document/8981449/
10. Hansen DK, Nasrollahi K, Rasmusen B. C, et al. Real-Time barcode detection and classification using deep learning. In: Proceedings of the 9th International Joint Conference on Computational Intelligence [Internet]. SCITEPRESS - Science and Technology Publications; 2017. pp.321–327. Available from: http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0006508203210327
11. Peng J, Yuan S and Yuan X. QR Code detection with faster-RCNN based on FPN. In Artificial Intelligence and Security. 2020: pp.434–443. Available from: http://link.springer.com/10.1007/978-3-030-57884-8_38.
12. Ciążyński K and Fabijańska A. Detection of QR-codes in digital images based on histogram similarity. *Image Proces Commun* [Internet]. 2015; 20: 41–48. Available from: https://www.sciendo.com/article/10.1515/ipc-2015-0033
13. Song C, Li Z, Xu W, et al. My smartphone recognizes genuine QR codes! *Proce of the ACM on Interact, Mob, Wearable and Ubiquitous Technol* [Internet]. 2018; 2: 1–20. Available from: https://dl.acm.org/doi/10.1145/3214286
14. Cui Z, Li W, Yu C, et al. A new type of two-dimensional anti-counterfeit code for document authentication using neural networks. In: Proceedings of the 2020 4th International Conference on Cryptography, Security and Privacy [Internet]. New York, NY, USA: ACM; 2020. pp.68–73. Available from: https://dl.acm.org/doi/10.1145/3377644.3377651
15. Zhang L, Chen C and Mow WH. Accurate modeling and efficient estimation of the print-capture channel with application in barcoding. *IEEE Trans on Image Proces* [Internet] 2019; 28: 464–478. Available from: https://ieeexplore.ieee.org/document/8453836/.
16. Zhang X, Duan J and Zhou J. A robust secret sharing QR code via texture pattern design. In: 2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC) [Internet]. IEEE; 2018. pp.903–907. Available from: https://ieeexplore.ieee.org/document/8659559/
17. Chen C, Li M, Ferreira A, et al. A copy-proof scheme based on the spectral and spatial barcoding channel models. *IEEE Trans on Inf Forensics and Secur* [Internet] 2020; 15: 1056–1071. Available from: https://ieeexplore.ieee.org/document/8794824/.
18. Liu S, Fu Z and Yu B. Rich QR codes with three-layer information using hamming code. *IEEE Access* [Internet] 2019; 7: 78640–78651. Available from: https://ieeexplore.ieee.org/document/8735663/.
19. Fu H, Zhao X and He X. Improving anticompression robustness of JPEG adaptive steganography based on robustness measurement and DCT block selection. Wang J, editor. *Secur Commun Netw* [Internet]. 2021; 2021: 1–15. Available from: https://www.hindawi.com/journals/scn/2021/9153468/
20. Tkachenko I, Puech W, Destruel C, et al. Two-Level QR code for private message sharing and document authentication. *IEEE Trans on Inf Forensics and Secur* [Internet] 2016; 11: 571–583. Available from: http://ieeexplore.ieee.org/document/7349185/.
21. Warang A and Patankar A. QR Code Based Image Steganography. 2017;3:1100–1105.
22. Singh N and Sharma D. An efficient multiple data hiding technique for medical images using QR code authentication. *Int J of Scic Res in Sci Eng Technol* 2017; 3: 135–139.
23. Aggarwal A, Das N and Sreedevi I. Attention-guided deep convolutional neural networks for skin cancer classification. In: 2019 Ninth International Conference on Image Processing Theory, Tools and Applications (IPTA) [Internet]. IEEE; 2019. pp.1–6. Available from: https://ieeexplore.ieee.org/document/8936100/
24. Liu C, Lin B, Lai J, et al. An improved decision tree algorithm based on variable precision neighborhood similarity. *Inf Sci* [Internet] 2022; 615: 152–166. Available from: https://linkinghub.elsevier.com/retrieve/pii/S002002552201163X.