Computers, Materials & Continua
DOI:10.32604/cmc.2021.015770
Article

Tech Science Press

Tamper Detection and Localization for Quranic Text Watermarking Scheme Based on Hybrid Technique

Ali A. R. Alkhafaji*, Nilam Nur Amir Sjarif, M. A. Shahidan, Nurulhuda Firdaus Mohd Azmi, Haslina Md Sarkan and Suriayati Chuprat

Razak Faculty of Technology and Informatics, University Teknologi Malaysia, Kuala Lumpur, 54000, Malaysia
*Corresponding Author: Ali A. R. Alkhafaji. Email: aliraheem1983@gmail.com
Received: 06 December 2020; Accepted: 23 January 2021

Abstract: The text of the Quran is principally dependent on the Arabic language. Therefore, improving the security and reliability of the Quran's text when it is exchanged via internet networks has become one of the most difficult challenges that researchers face today. Consequently, the diacritical marks in the Holy Quran which represent Arabic vowels (أ. و. ي) known as the kashida (or "extended letters") must be protected from changes. The cover text of the Quran and its watermarked text are different due to the low values of the Peak Signal to Noise Ratio (PSNR), and Normalized Cross-Correlation (NCC); thus, the location for tamper detection accuracy is low. The gap addressed in this paper to improve the security of Arabic text in the Holy Quran by using vowels with kashida. To enhance the watermarking scheme of the text of the Quran based on hybrid techniques (XOR and queuing techniques) of the purposed scheme. The methodology propose scheme consists of four phases: The first phase is pre-processing. This is followed by the second phase where an embedding process takes place to hide the data after the vowel letters wherein if the secret bit is "1", it inserts the kashida but does not insert the kashida if the bit is "0". The third phase is an extraction process and the last phase is to evaluate the performance of the proposed scheme by using PSNR (for the imperceptibility), and NCC (for the security of the watermarking). Experiments were performed on three datasets of varying lengths under multiple random locations of insertion, reorder and deletion attacks. The experimental results were revealed the improvement of the NCC by 1.76%, PSNR by 9.6% compared to available current schemes.

Keywords: Text watermarking; tamper detection; authentication; quranic text; exclusive-or operation; queuing technique

1 Introduction

Nowadays, the tamper detection of text documents has increased significantly for secure data transmission over the internet [1]. The importance of efficient and secure multimedia rights regarding digital watermarking techniques has been realized for the privacy preserved information communication [2,3]. The watermark scheme is categorized into two primary stages: First is the

stage where watermark embedding is carried out in the initial information; second, is the stage for extracting the watermark from watermarked information that has been assaulted.

Capacity, imperceptibility, robustness and safety are the watermarking criteria. Capacity is the total number of hidden bits while an imperceptibility location is used to measure the difference between the original location and the watermarked location, noting any additions to the original location. Robustness is the capacity to remove or detect a watermark after attacking the watermarked location. Finally, the security requirement is the challenge of removing a watermark without destroying it [4].

Text watermarking is one of the media's criteria of digital watermarking and has become crucial due to developments in information concealment. It refers to the process of embedding the watermark inside a text document as a way of protecting in terms of the authentication of the document. The protection of such information is necessary to preserve the watermarked text so that it is the same as the original text without change. In the domain of text watermarking, there are three major issues which must be resolved: imperceptibility, capacity, and security [5]. Due to the obvious sensitive nature of the text of the Holy Quran, many methods have been proposed to prevent people from tampering with these invaluable and sacred scripts. The proposed method here is the intended use of a fragile watermark which takes both the wavelet domain and the spatial domain into consideration. Therefore, focused attention is needed to develop some watermarking schemes, especially when it comes to the Arabic language text.

The Quran was written in Arabic, so in order to conceal information within it, one needs to know the features of the language. Text watermarking has many duties, some of which include the use of the "kashida" letter extension (one of the features of the Arabic language) and the Arabic vowels ـٱ. ـو and ـي. Both of these tasks are used to embed secret data and determine the order in which the words and conjunctions are placed. Even a hyper mix of two or more features may be used to hide secret data within the Quran's text. Each of these methods has its advantages and disadvantages; for this reason, a new algorithm must be produced to overcome all of the disadvantages of the existing methods [6,7].

In this paper, the author presents a scheme for improving the embedding imperceptibility and security through the use of tamper detection and location authentication of a Quranic text watermarking scheme based on vowels with kashida using the exclusive-or (XOR) and queuing techniques. The proposed scheme utilizes the Queuing technique which was chosen in order to check the embed phase for the tamperage of words forms the text of the Quran's text. As a result, when any of type of attacks such as deletion, insertion, or reordering is detected one can then truly say whether or not the text was, indeed, tampered with, check by extract phase. The main contribution is the development of an improved tamper detection scheme. Thus, this contribution is different from other similar approaches of existing literature due to the addition of the location of authentication in Quranic text watermarking. The proposed scheme was implemented to detect tamperage by a hacker that tried to manipulate the watermarked Quranic text. The use of the hybrid technique refers to using both the XOR and the Queuing techniques which make the proposed scheme more secure in terms of fast and accurate tamper detection. The proposed scheme was able to detect the location of the insertion, deletion and reordering of the tamperage accurately.

This paper contains five more sections. Section 2 provides related work. Section 3 presents the proposed methods and evaluation. Section 4 discusses the comparison and results, and finally, Section 5 offers the researcher's conclusions.

2 Related Work

Obtaining the authentication and credibility of users and information through the internet is vital. However, ensuring the Quran's text remains authentic is even more vital. The authenticity and originality of verses in digital media can be challenging to preserve especially as it is prerequisite for accomplishing this feat in a quick manner. The proposed method for implementing a Quranic authentication system is based on information retrieval techniques and hashing algorithms. The evaluation of different characteristics may impact the accuracy of the authentication process. Results showed that hashing verification is a good candidate for automatic authentication with high confidence. As a result, rather than focusing on encryption, the primary focus is on authenticating information. However, finding proper Quranic documents online for this research is difficult; consequently, the relationship between information providers and readers is not typical e-commerce, one-to-one relationship in which typical hash algorithms can be sent to the known receiver. Nevertheless, private keys intended for a receiver will not be applicable in our scenario where the tool is expected to quickly check the Internet for authentic Quranic documents that have not been tampered with [8–10].

Tamper detection in plain text papers uses a zero-watermarking algorithm. The algorithm produces a watermark dependent on the quality of the text which can later be obtained using the extraction algorithm to define the tampering position in the document. Watermark matching patterns and watermark distortion rates on several text samples of varying duration are used as measurement parameters. Findings show that this algorithm still detects tamperage even when the amount of it is limited. With the limited volume of attack, the presence of a watermark and the reliability of the details are, therefore, difficult to determine [11].

A zero-watermarking method used for authentication utilizes the characteristics of the Arabic characters without changing the file. The watermark key focuses on the characteristics of the original verses For each verse of the Quran in the initial stage, a key is produced which checks the name and number of the surah along with its *ayat* or verse numbers. The key is then matched with one that is stored at the certification authority. The findings demonstrate the success of this method in detecting random tampering attacks that include 100% detection of any distortions or modifications rendered to the Quran's text whether intentionally or unintentionally. Thus, the text should be examined in the event of any discrepancies as shown in Fig. 1 [12].

Some researchers suggest using a tamper locating algorithm for authentication of the content found in .docx documents. The authentication information that is not connected to the text content is stored in the key setting file called document.xml through the segmentation of the display characters. Identifying the quality of the text content can detect whether or not the embedded watermark is similar to the authentication watermark. Tests show that the algorithm is very sensitive to any change and can very well identify the manipulated places [13].

Authentication of image content is primarily used to determine the validity of image content and can easily locate the tampered area in order to determine the motive for actions. Therefore, for content-preserving processes such as noise, compression, and replacement methods, the authentication algorithm must be robust and made vulnerable to malicious attacks. Furthermore, semi-fragile digital watermarking techniques can, to some degree, differentiate between content-preserving activities and malicious tampering attacks [14].

Referring to multiple watermarking systems to secure ownership and tamper detection, this scheme robustly embeds the watermark details used for authenticating copyright by the synonymous replacement process; thus, it uses a double fragile watermark to achieve the detection

and position of manipulations. The suggested system has a strong benefit when it comes to its implementation; however, are already some shortcomings such as the watermark's robustness, the precision of finding the tamperage when dealing with a more complicated attack, both of which need further developments [15].

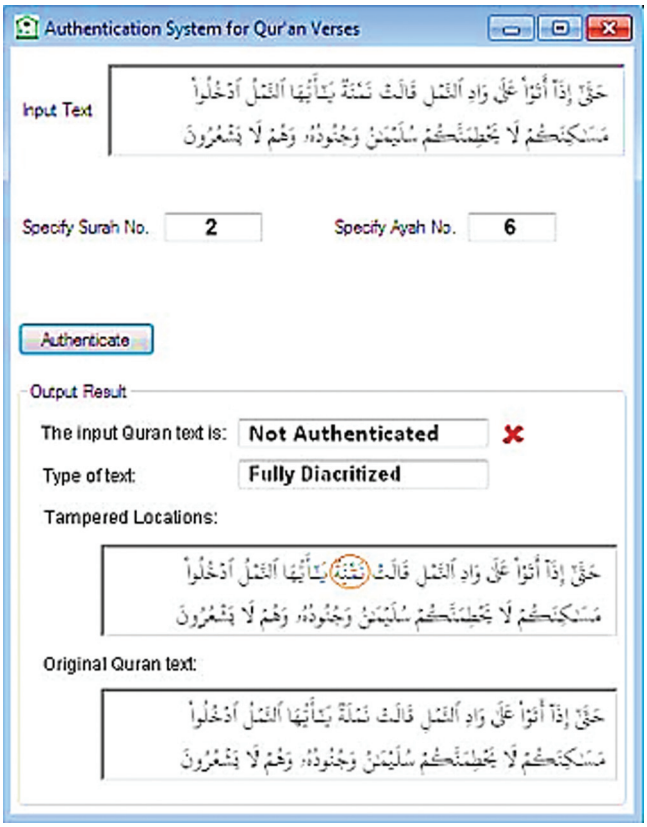

Figure 1: The authentication system of Quran verses [12]

According to the novel CNN-based security-guaranteed image watermarking generation scenario for smart city applications, the content-based watermark synchronization scheme recognizes watermark embedding positions via stable image feature points in the present anti-geometric attack watermarking algorithm, embeds the watermark in a local neighbourhood feature point, and uses the feature points to locate the watermark. These methods demonstrate increased robustness. The gradient path distribution of the neighbouring pixels of a feature point is used to determine the direction information for each feature point. In practice, in a neighbourhood window based on the feature points, samples are generated, and a histogram is used to measure the gradient directions of the adjacent pixels. Thus, with the main direction of the feature point, the peak of the histogram represents the primary direction of the neighbourhood gradient. The model achieves enhanced robustness with the incorporation of CNN [16].

The quaternion Fresnel transform (QFST), computer-generated hologram (CGH) and two-dimensional Logistic-adjusted-Sine map (2D-LASM) techniques, a novel four-image encryption scheme is proposed. Four images are in the proposed process, Holistically processed in a vector manner by first using QFST. Then, using virtual RPM, phase-shifting interferometry (PSI) and

Burch's coding process, the encrypted CGH of the input complex amplitude, which is constructed by the components of the QFST-transformed images, is generated [17].

The RPMs are in order to avoid using the entire RPMs as the decryption keys 2D–LASM produced. LASM's initial values are used as decryption keys. CGHs have the advantage of digital fabrication, storage and transmission compared to traditional optical holography. Subsequently, the obtained CGH is permitted based on LASM to improve security. Numerical simulations show the feasibility and performance of the proposed system. The encrypted hologram is scrambled based on 2D-LASM to enhance the security of this paper and weaken the correlation. The validity of the proposed technique of image encryption is demonstrated by experiments [17].

The 2D Fuzzy C-Means (FCM) algorithm has been widely used for segmenting medical images. Throughout the years, various extensions of it were proposed. A modified version of FCM was therefore proposed for segmenting 3D medical volumes, which has rarely been implemented for 3D medical image segmentation. We present a parallel implementation using the Graphics Processing Unit of the proposed algorithm (GPU). Researchers state that when working with 3D models, reliability is one of the primary problems when using FCM for medical imaging [18].

A hybrid parallel implementation of FCM is therefore suggested for removing volume objects from medical files. Using actual medical data and virtual phantom data, the proposed algorithm has been validated. The key factors for the device validation were segmentation accuracy of predefined datasets and actual patient datasets. To demonstrate the efficiency of each implementation, the processing times of both the sequential and the parallel implementations are calculated. The results obtained conclude that the parallel implementation is 5x faster than the sequential version of the same operation. Concentrate on automatically segmenting oblique slices from 3D volumetric data. To minimise the processing time here as much as possible, an acceleration mechanism using the GPU will be considered [18].

Watermarking systems have been implemented before, but authentication and tamper detection are still a major research issue. A double image-based watermarking scheme was introduced using the Local Binary Pattern (LBP) to protect multimedia documents from unauthorized alteration. The suggested approach involves the following procedure: the host image is partitioned into non-overlapping blocks during watermark encoding. This is followed by the generation of the process vector (S) using LBP and secret watermark bits to execute XOR. A two-bit authentication code is created from the (S) vector and embedded in a dual image based on a shared secret key (ÿ). The findings are compared to state-of-the-art approaches to show the usefulness of our suggested system. It is noted that the proposed scheme is secure and robust against different standard attacks whereas it can detect the integrity of messages inside the watermarked item [19].

Different scenarios have recently been proposed by the researchers depending on the insertion of variable numbers of kashida per word. Such strategies have produced better results in terms of capacity and security than the previous methods; however, they also reveal a noticeable weakness in the process of retyping [20]. These two forms are designed to provide the working memory for the cover sharing. This can be very useful because the secret posts are created without giving any preference to the user. These two suggested patterns for the secret stocks are hidden within the texts using the Arabic script features to conceal the information based on the extension of the kashida. Moreover, two optimization models are suggested that used the kashida to hide the secret stocks in various scenarios. The enhancement is focused on using the bilocation of the kashida possibilities for the embedding of hidden data within the text. The kashida locations of the first

form are considered to lead to the second form, then to the third one, the fourth one and so on as depicted in Fig. 2.

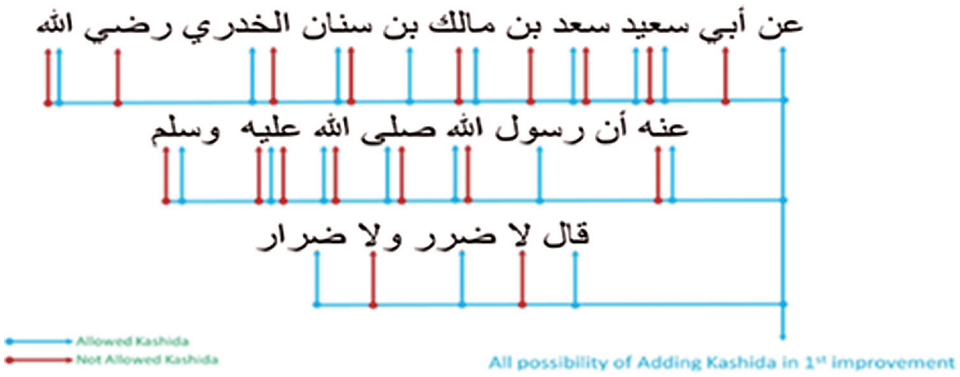

Figure 2: The counting-secret sharing process-based first approach [20]

Previous literature shows research has been conducted on text watermarking approaches and methods that were proposed for several purposes of information security. Some research proposes a robust content authentication method for paper text documents, solving the problem of paper document content integrity verification, tamper identification and tamper position locating. Furthermore, in this method, the watermark information does not depend on the additional carrier, and the text document is embedded with the watermark visible. The watermark is embedded during the process of a printout, which avoids the risk of malicious tampering of the watermark information. As a result, this method is seen to possess high-security performance. The watermarking algorithm is robust against the print-and-scan attack and has a high capacity with low accuracy. In addition, the minimum string edit distance algorithm is used to compare the difference between the authentication watermark signal sequence extracted from the scanned image and the one calculated in real-time, through which it determines whether or not the paper document's text content has been changed, and if so, it accurately locates the place where it was tampered with [21].

The Intrusion Detection System (IDS) is an application that detects malicious attacks or data breaches within a network. As a vital network security element, IDS has been frequently used in recent years. This analysis aims to define the best-fit solution, that would reduce the number of features substantially. Furthermore, with less processing time, the method would lead to high classification accuracy. We propose a hybrid feature selection model that combines the strengths of the legacy filtered and wrapper selection mechanism to reach this aim. This proposed amalgam of approaches is supposed to efficiently pick the optimal set of intrusion detection features. Using correlation feature selection (CFS) along with three different search techniques known as best-first, greedy stepwise and genetic algorithm, the suggested hybrid model was carried out [22].

In order to evaluate each of the features that were first selected by the philtre process, the wrapper-based subset evaluation uses a random forest (RF) classifier. On both the KDD99 and DARPA 1999 datasets, the reduced feature range was checked in a supervised setting using an RF algorithm with ten-fold cross-validation. The outcome shows that, in terms of detection

time, accuracy, and detection rate, choices made by the hybrid feature had better performance improvements. A low false alarm rate was reported, as well [22].

The suggested zero watermarking utilized the material features of the text to create a watermark instead of embedding the watermark in the text using the structural variable and word length, that is common to all kinds of texts. It consisted of two stages: text embedding and extracting. The watermark was generated with the data holders, and the extraction was done with the certifying authority (CA) [23]. The architecture of the proposed tampering detection scheme is shown in Fig. 3.

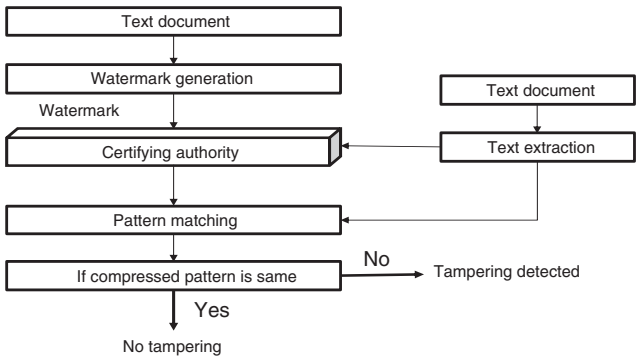

Figure 3: The overview of HSW technique [21]

A technique was developed that used a text document as an input for manipulation detection, and the watermark was created based on the HSW (hybrid structural component and word length) approach. A watermark extracted pattern was registered with the (CA) where the attacker could change the document's contents. During the tampering identification, the extraction method was used to retrieve the watermark and the template fitted the recorded (CA) model. A decision on the interference was made that depended on the degree of pattern matching with MD5 compression. The evaluated attacks including the deletion, insertion, and reordering using this method achieved high accuracy than other works and. However, this approach could not update the detailed ownership of the source in the generated watermark information [24].

The proposed zero-watermarking technique known as watermark arrangement is based on the Markov Model Level 4 Word Mechanism (ZWAFWMMM) and is used for authenticating information and detecting tamperage within the Arabic text. It is an effective model as ZWAFWMMM adopts a hybrid system. Nevertheless, due to the complicated nature and structure of the Arabic language, the basic curriculum uses conventional techniques which lack the capacity to provide effective solutions to the Arabic text. The findings of the experiment reveal that ZWAFWMMM is more sensitive to all forms of tampering and highly accurate when it comes to the low capacity of tamper detection [25].

As mentioned before, the Holy Quran is a religious book that plays an extremely vital role in the life of Muslims as the main decisions of Islamic jurisprudence and, indeed, the everyday life of Muslims are based on the instructions written in the verses of the Holy Quran. Ordinary Muslims cannot judge the authentication of the verses of the Holy Quran if the verses have been tampered with. In fact, it requires intense attention, extensive knowledge, and dedicated efforts to be able to tell the difference between true Quranic verses and ones that have been tampered with, especially when one or more words are missing from the recitations. Typically, the authenticity

of an online quote attributed to the Quran can be confirmed by making a comparison between the online quotations and the original text of the Quran [26]. Furthermore, the Holy Quran is written in Arabic language and with various styles, such as plain text, Uthmanic, Koufi, Kaloon, and other such styles [26]. These styles are also used in the Middle East and all Muslim countries. However, Tab. 1 mentions the current comparison studies with other baseline approaches.

Table 1: The summary of comparison with existing work

References	Domain	Methods	Type of Attacks	PSNR (dB)	NCC/%	Remarks
[27]	Text document image	The Pascal Triangle	Text Modification Attack	54.95	0.98	Increased the imperceptibility and security with less capacity
[28]	Quran text	XOR diacritics of the special characters	Message Modification Attack	56.34	NA	Increased the capacity and solve the complexity
[29]	Quran text	The moon and sun letters	Message Modification Attack	61.16	NA	Increased capacity with low security
[30]	Text document	LSB algorithm	Text Modification Attack	62.46	NA	Increased imperceptibility with less security
[31]	Arabic text	Inserts the pseudo-space and other three small or zero-width spaces	Text Modification Attack	NA	NA	High imperceptibility and capacity with low security
[32]	Arabic text watermark	Adding diacritics into any Arabic letter	Text Modification Attack	NA	NA	High capacity with less imperceptibility and security
[33]	Arabic text Hadiths	Counting based secret sharing with kashida	Text Modification Attack	52.05	NA	Increased the imperceptibility and security with less capacity
[34]	Arabic Text documents	Kashida with diacritics	Text Modification Attack	59.04	NA	Considered the authentication with a high degree of capacity

3 Proposed Watermarking Method

The methodology used in this research is divided into four phases: The first phase is the pre-processing phase which, as its name suggests, is responsible for the preparation of the hosting media such as the Quranic text in this case, and a secret bit (data that was hidden in the text). The second phase includes the embedding of the secret bit within the Quranic text. The third stage involves the extraction of the data including the attack process and the final phase is the performance evaluation of the scheme through the use of various measures. The details of each phase are discussed below.

3.1 Pre-Processing Phase

In this phase, the watermark is converted from the binary image into the sequence of bits by decoding it with certain conditions (e.g., inserting a kashida in the next letter after the vowel).

Next, the binary image is scanned to sequentially find the byte. If the byte value is 255, then the secret bit is 0. After finishing the binary image scan and byte assignment, vector 1 and 0 are produced. During this phase, the original Quranic text is called the cover text. This text consists of various characters that are given a unique ASCII code with a decimal value. This decimal's value is useful for determining the condition of embedding which is known in advance. Each location is stored in the vector-based condition accordingly and produces the vector for the position as shown in Fig. 4. After this process, the pre-processing phase was completed that covered the Quranic text with a secret bit ready for the embedding in the next phase.

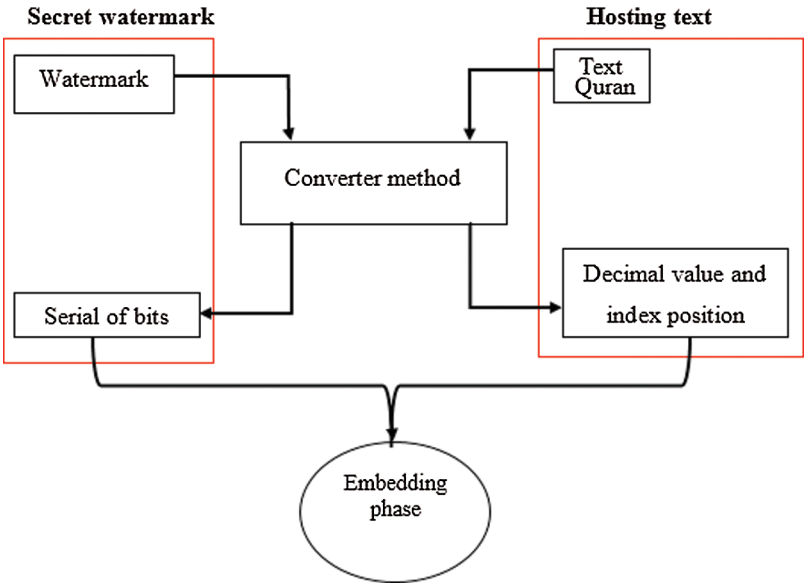

Figure 4: The converting the secret watermark and host text into binary bits

3.2 Embedding Phase

The proposed scheme is based on the vowels to be embedded in the Arabic text where the three vowels are used. These letters are selected because they are the most redundant characters in the whole text of the Quran. The embedding process is comprised of the preparation phase for both watermark and covers of the Quranic text. For watermarked text, all the bytes are converted into bits and then stored in the vector. The cover of the Quranic text should scan first for counting several vowels and check the condition (inserting a kashida in the next letter after the vowel) that is appropriate with the embedding protocol.

The embedding strategy is accomplished in the following six steps:

- Prepare both watermark and cover Quranic text. The preparation step is necessary for any watermarking scheme which is comprised of two elements.
- Prepare the watermark for conversion into binary decomposition in the form of 1 and 0 that results in secret input of the system as a serial of bits.
- Prepare the Quranic text including manipulating the text as the UTF file format before embedding to make it compatible with the ASCII code and further control.
- Open text file to check if there are vowel letters and then save them in vector.

- Match secret bits with the condition (inserting a kashida in the next letter after the vowel). If matching greater than secret bits then invert the secret bits that are obtained from the watermark ($0 \rightarrow 1$ & $1 \rightarrow 0$).
- Apply the embedding process by adding the kashida character when the secret bit is 1 otherwise do not add it.
- Embed to satisfy the condition of the first vowel letter to hold the secret bit (as kashida).
- After finishing the cover text, prepare the watermark text to send the receiver.

Embedment in the proposed scheme with the condition is necessary to fulfil the agreement between the sender and receiver as shown in Fig. 5.

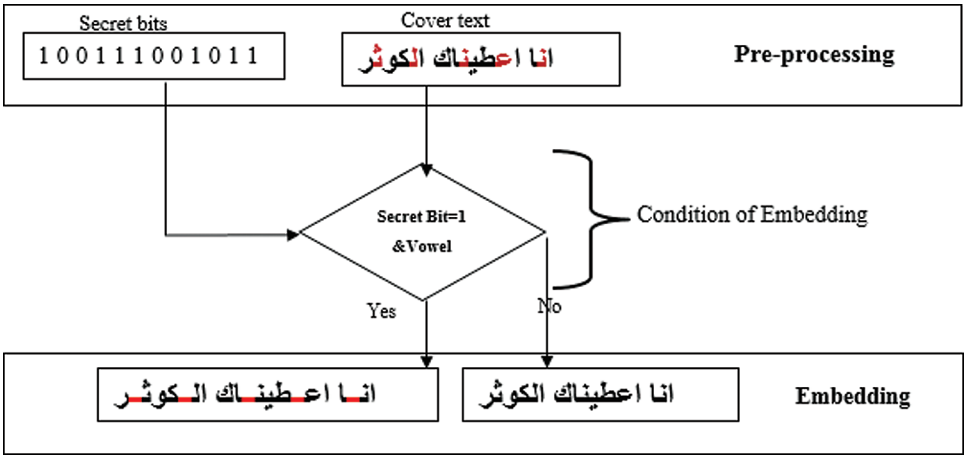

Figure 5: Embedding process with the condition of the proposed scheme

There are two cases when inserting the bits from the watermark into cover Quranic text which is 1 and 0 depending on the presence or absence of kashida. These four cases control the most embedding issue as summarized in Tab. 2.

Table 2: Four cases of the embedding process

Secret bit	Kashida within the cover text	Results
1	1	1
0	1	0
1	0	1
0	0	0

Results can be interpreted as 1 if the kashida is present in the watermarked text; otherwise, if it is absent, it is represented as 0. The kashida in the original text (cover text) may or may not appear inside. Each byte in the binary image is represented by bit 1 (white) and 0 (black) and these bits are stored in vector to embed it sequentially into the cover text. After locating the vowel letters and insertion into the cover text, the technique checks the condition of the embedding of the letter in the next vowel letter.

3.3 Extracting Phase

Although the extraction process includes many special cases, it is the reverse process of the embedding technique. The enclosed complex data inside the watermarked text needs to be arranged according to the embedding and extracting techniques which may be changed by the external effects. The extracting technique is responsible for the action of all these details. The main contribution of the proposed embedding or concealing technique of the secret bits is the insertion of the kashida at the next letter after a vowel. For extracting the hidden information, one needs to find the vowel letter to locate the next letter. To find the appropriate kashida position for extracting, the condition for the embedding needs to be checked where the presence or absence of the word's kashida is represented by the secret bit 1 or 0 as shown in Fig. 6. After this phase, a certain letter is converted from the ASCII into the binary bit. These bits are stored in a vector for the next process tamper detection attack as explained below.

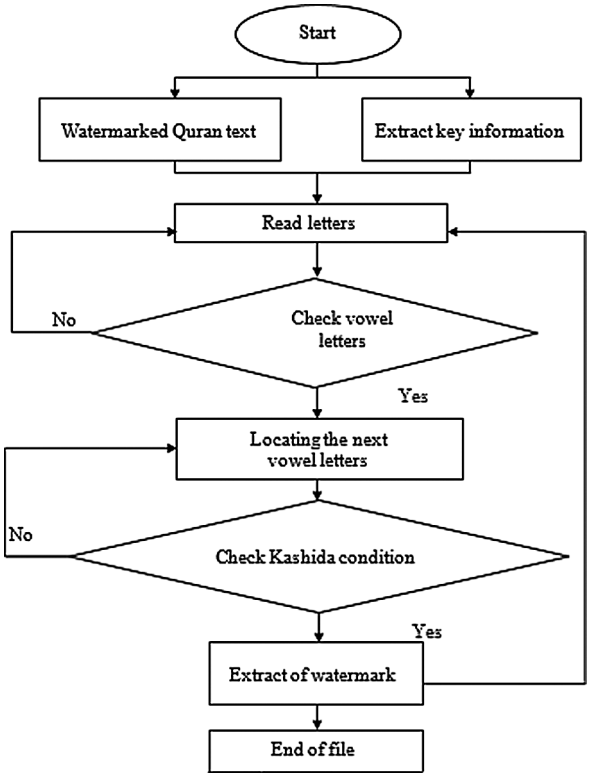

Figure 6: Extraction proposed scheme

3.3.1 Tamper Detection Attack

During the transmission, the watermarked Quranic text suffers from many kinds of attacks and tamper detection is one of them. Tamper detection in the proposed scheme aims to find any external manipulation that is unauthorized. The recipient does not know whether the watermarked text was tampered with or not. Thus, the proposed scheme detects whether or not the watermarked Quranic text has been tampered with by reading each letter or word throughout the document. Therefore, that which the watermarked text is converted into consists of the decimal value and vector of the decimal value and is examined by taking the highest value found in the

text. Next, by examining and analyzing it with the remaining text characters, one of its values is obtained. A file must be opened to store these values where, afterwards, they are read and combined with different techniques to both stores in a vector and also determine their location as well.

Furthermore, the detection of a tampered-with watermarked Quranic text usually shows that the text has been manipulated by the intruder in one or all of three ways. First, the intruder adds either a single letter to the word or adds a single word to the statement to confuse the recipient of the watermarked Quranic text. Second, the intruder deletes one letter in the word or one word in the sentence, which affects the meaning of the sentence. However, quite often, it is an unimportant letter in the text that the intruder seeks to modify. Moreover, the deletion or shifting of such a letter does not change the whole meaning of the word or statement. Depending on the nature of the language, it is important to work with some letters that have no meaning or are often aesthetically located within the text. Third, the intruder often manipulates the letters in the word or sentence to trick the receiver and becomes successful due to the similar-looking shape and size of the original text in the watermarked Quran. If detected the file must be read again, otherwise, the scheme is claimed to be successful.

In the present study, the proposed scheme finds there are three types of tampering and resolves the drawback of the existing methods for tamper detection inside the program as mentioned above. Thus, two techniques are included in the proposed scheme for controlling tamper detection. These include the queuing technique with the FIFO process and the XOR operation to check changes in the watermarked Quranic text as discussed below.

Queuing Technique The main aims of watermarking schemes are to conceal the secret bit in the cover Quranic text. This is first done on the sender's end to produce the watermarking of the text. It is then sent through the trusted channels to the receiver for extracting the original secret text. The receiver's side is responsible for receiving the data and is not aware of the missing part in the received data. For this reason, some missing information may contain a secret bit; therefore, it is important to keep an eye out for any information which might be lost during this time. Sensitive data should be integrated and there is no way to proceed without every bit of information in its wholeness. Thus, the queuing technique is proposed to track each letter and word between the sender and receiver. The main issue here is to detect the presence of any tamperage in the text of the Quran wherein any text represented in the digital world as the ASCII code (hexadecimal) must be converted into a binary representation (a serial of bits). These bits involve the words and statement (whole text). Logically one can depict this compensation as a queue Fig. 7.

The queue looks like a 2D array where the rows represent a place of words with each cell consisting of a letter. Every word located in a single row, along with the number of rows in this array, considers the whole text. The last bit in the row specifies the results of the XOR operation, where the last bit in the vector that denotes the XOR of all the bits in the end or rows mentioned before. On the receiver side, the whole text is received as one package and the receiver reads it word for word or even letter for letter. Consequently, the architecture of the text of the Quran looks like queue letter by letter representing the word (raw of the queue) and word by word to represent the statement (verses) or the whole text (column of the queue). The receiver manipulates the text of the Quran as a serial of bits. Therefore, the absence of one letter or word immediately gets discovered due to the change in the sequence of the queue. The receiver side can catch the original text queue by comparing with the key that contains an index for this queue. However, the tampering or changing of the text of the Quran is easy to discover with its secret information.

The secret key that sends to the receiver should contain the indexing of the queue, wherein the indexing queue consists of the manipulated bits generated from many processes based on the XOR operation. Generally, half of the words (50%) are taken from each row and the XOR with another half of the same word is performed to produce the bits considered as the indexing bits. The same procedure is followed for the rest of the queue and applies to all of the text of the Quran as illustrated in Fig. 8.

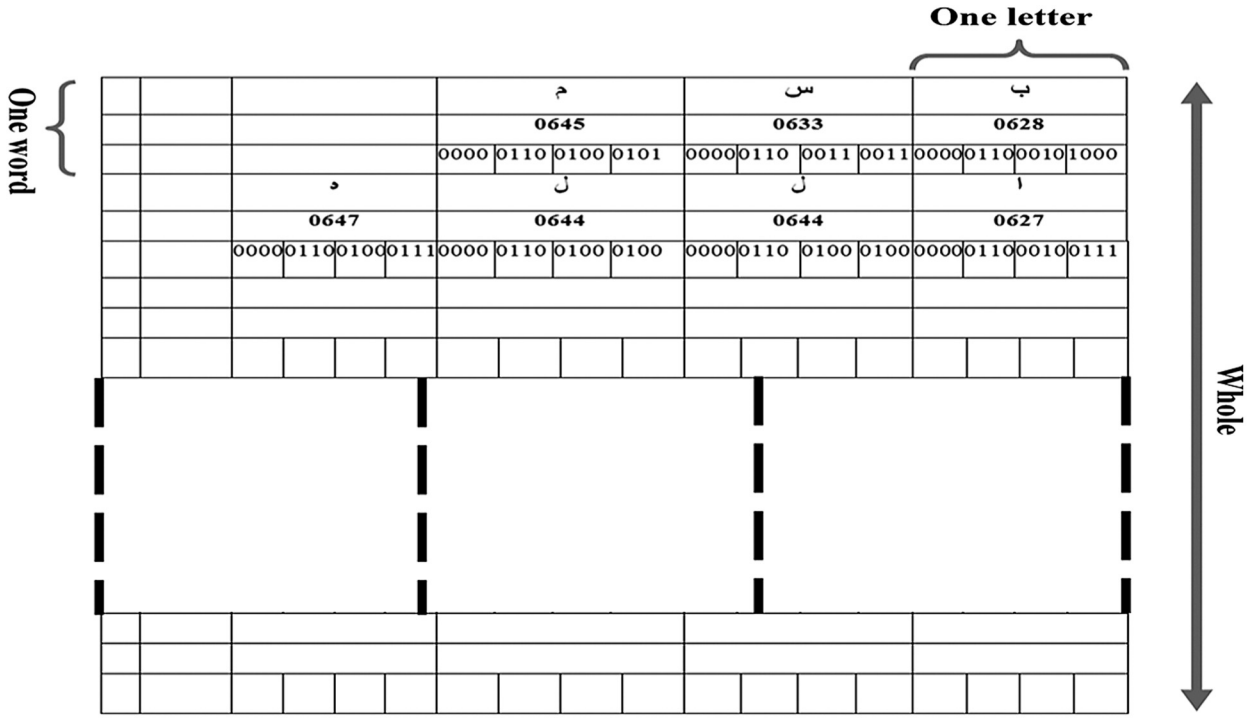

Figure 7: Structure of the queue in the proposed scheme

In the queue, the bits of the word XOR results in the bit store at the end of the row which is checked later by the receiver. Changing or modifying any letter of the word leads to an error in this bit. The same XOR operation is applied to all these bits for all the text (queue) and the results are stored in the secret key.

XOR Operation After embedding, the watermarked Quranic text is sent to the receiver wherein all the embedded secret information is extracted from the watermarked text that is stored in the watermarked key. The watermarked key can carry very little information in this case, and as such, there must be some technique to retrieve as much information as possible. It is known that a proposed scheme that produces one decimal value can be used by another partner to find the real or original file. This value is considered as the maximum value in the text file extracted from the vector of the queuing technique. After producing this value, the receiver performs the XOR of all the values in the queue with this number inside the key to find the original text.

3.3.2 Tamper Detection Process

The proposed scheme detects the tampering in the watermarked file by reading each letter or word throughout the document. First, the watermarked text is converted into ASCII and the

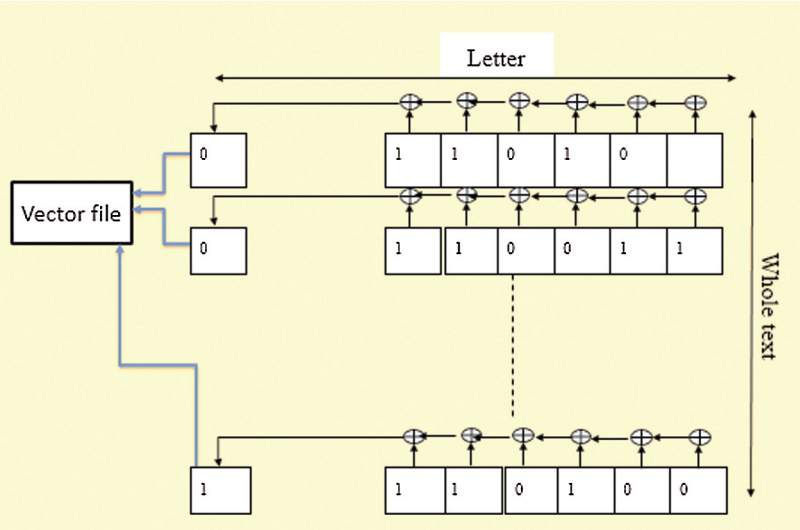

Figure 8: Process of the queue for the tampering

examined by taking the highest value found in the text. Next, by examining and analyzing it with the remaining text characters one of its values is obtained. A file must be opened to store these values and then they are read as well as combined with the XOR to store in a vector. In this event if the tampered text is detected then the file must be read again; otherwise, the scheme is claimed to be successful. Thus, Fig. 9 shows the basic architecture of the proposed tamper detection scheme.

Any text file consists of a group of letters and even space is considered a letter under a certain value. Thus, each text file considers a series of letters that can be viewed as a long vector. The letter or character is coded in the computer in a unique ASCII code which is shown as the decimal value. In fact, a text file is a group of decimal value and each value matches one character in the text file. Consequently, for decoding the letters within text file the decimal value file is created similar to the 2D array or matrix with rows and columns as depicted in Fig. 10

The conversion of the watermarked Quranic text file into the corresponding decimal value is necessary. The decimal values are easy to generate using logical operations such as the XOR, OR, AND, and so on. The basic insight can be obtained from the decimal file which is necessary to secure the process and information itself. If conditions are favourable, the logical process is started. In the proposed scheme, this term refers to the selection of the maximum number or value from the decimal file. This maximum number that is used for the important logical operation is called the Exclusive OR process ($\oplus$).

Moreover, it is worth explaining the details of the XOR operation here. This operation is considered to be the logical operation that accepts two arguments including the digital input and produces one argument result. For more than two arguments one must combine many operations. The XOR operation is mainly applied due to its ease of use, understanding, and ability to acquire input as a result in the second iteration. It implies that if the output result is XOR-ed again with one input argument then one can get the second input argument. On other hand, if only one of p or q is T/1, then the result will be T/1. Otherwise when both $p \oplus q$ is (F/0) or (T/1) then the result will be F/0. Thus, considered to symbol F = false, and T = true as shown in Fig. 11.

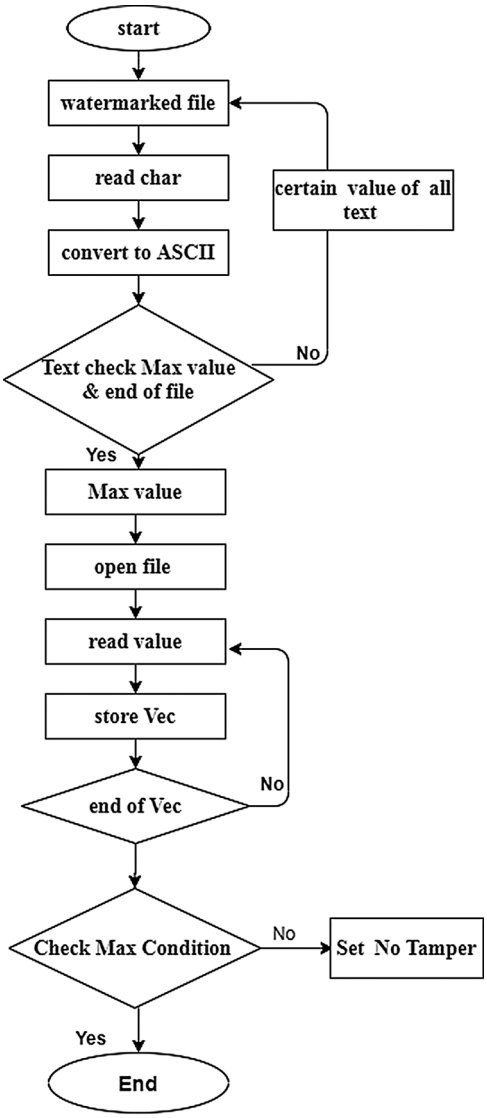

Figure 9: Tamper detection process

After the embedding phase, the authentication process starts with the decimal file and the maximum number of some procedure via the XOR operation. Every number in the decimal file is XOR-ed with the maximum number that is derived from the decimal file for tracking. Consequently, the Queue vector is designed to store the results (decimal file $\oplus$ MAX value) using the queuing technique. The first decimal value obtained from the decimal file is XOR-ed with the max value and the results stored in the first position of the queue vector. The same procedure is used for the next second decimal value and its result is store in the second position of the queue vector. The queue vector generation strategy followed the First In First Out (FIFO) or on a first-come, first-served basis. On the sender's side, a vector is produced after embedding which is further used by the receiver to verify the presence or absence of tamperage in the watermarked file before extracting to. After the embedding process, to keep track of the queue vector the reverse procedure is followed in the sender side as displayed in Fig. 12.

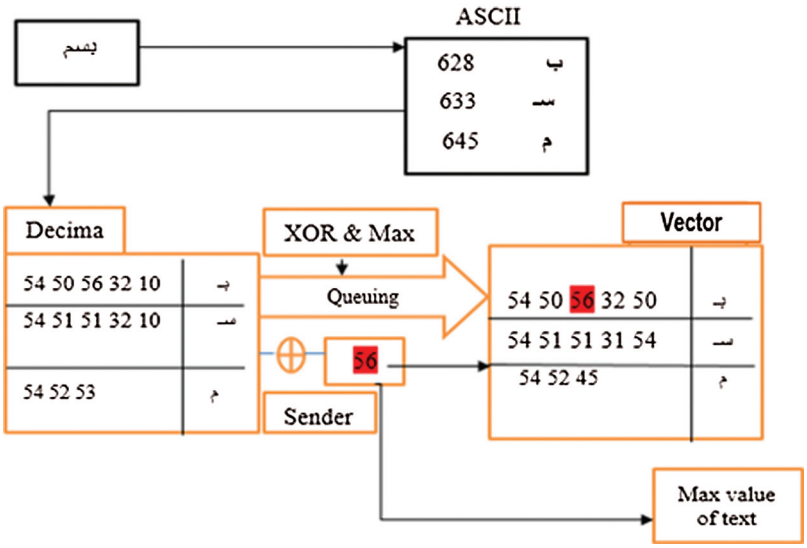

Figure 10: Decoding scheme of the watermarked text

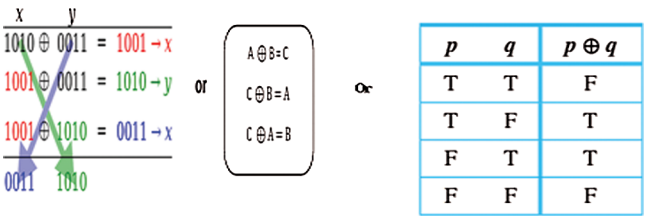

Figure 11: Features of the XOR operation

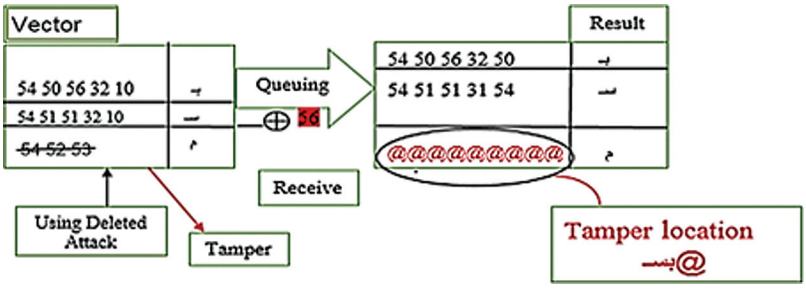

Figure 12: The tamper detection strategy of the proposed scheme

After the receiver obtains the necessary information from the watermarked key, the XOR operation is performed to logically generate the queue vector and max value number. This operation results in a decimal file identical to the one derived from the watermarked file. Thus, the procedure of authentication in the system operates like $A \oplus B = C$ at the sender side and $B \oplus C = A$ at the receiver side (or vice versa).

After finishing the XOR process, the resultant file is compared with the decimal file derived from the watermarked text to identify the presence (or absence) of any differences, indicating the presence (absence) of tampering. This method is very secured, less time consuming and makes it more difficult for the hackers or intruders to guess the secret key. Using this technique, three main threats can effectively be detected such as the insertion of the outlandish character or deletion of

one of the specific existing characters, or the modification or reordering of characters in the text file as explained underneath.

Fig. 13 shows the basic design of the proposed tamper detection scheme that reads the secret key and watermarked file on the receiver website. Next, the method examines the vector file from the key in the text. Consequently, it reads both vector and watermarked characters until the entire file is processed. The examination and analyses for the remaining text characters are obtained until the end of the file is processed. Then, it reads again to rectify the presence of any issue related to the processing. In addition, it normalizes the length of files by reading both vector and watermarked characters. When the vector and watermarked characters are combined with the XOR of different values, it implies the successful detection of the tampered text. Thus, the file must be read again. Finally, the scheme writes the file with the position of a mark to detect the location.

In addition, Fig. 14 shows the whole identification tamper pseudocode scheme used in the proposed technique.

3.4 Evaluation Phase

Every watermarking system aims to evaluate the performance of the resultant watermark media on the sender's side before sending it to the receiver, thereby ensuring the robustness of the proposed scheme against attacks. In this paper, two types of assessment evaluation are used to check the performance of the proposed scheme: First, the Peak Signal-to-Noise Ratio (PSNR) is evaluated to determine the imperceptibility of the watermarked version. The PSNR value reflects the similarities between the watermarked Quranic text with the original. If the PSNR is high then it is easy to deceive the intruder or attacker, implying the absence of secret bit involvement in this media.

The distribution of the secret bit embedded to cover the Quranic text reflects the PSNR value, while the PSNR detect the frequency of the bits within the cover Quranic text. When the bits become heterogeneous or more chaotic, then the PSNR will get high value. Thus, for embedding is introduced. The high PSNR value indicates good quality of the Quranic text, thereby all the previous methods tried to enhance the PSNR [26]. Subjective methods depend on the human eye and assessment without using any reference evaluation. The PSNR value that is used to measure the quality of the text of the Quran after embedding can be defined as Eqs. (1) and (2):

$$PSNR = 10.log_{10}\left(\frac{MAX_1^2}{MSE}\right) \tag{1}$$

With,

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2 \tag{1.1}$$

where the variable integer MAX is the maximum number of possible letters value of the input text, then MAX is considered one or 255 bytes which means adding the kashida into embedding process when it contains an unsigned 8-bit integer data type. While MAX considers 0 when it does not have any kashida, m: integer value is the range of the whole text, and n: integer variable considers the number of verses; in the other part of the Eq. (1): character variable refers to the original text and K: character variable is the noisy text (watermarked text).

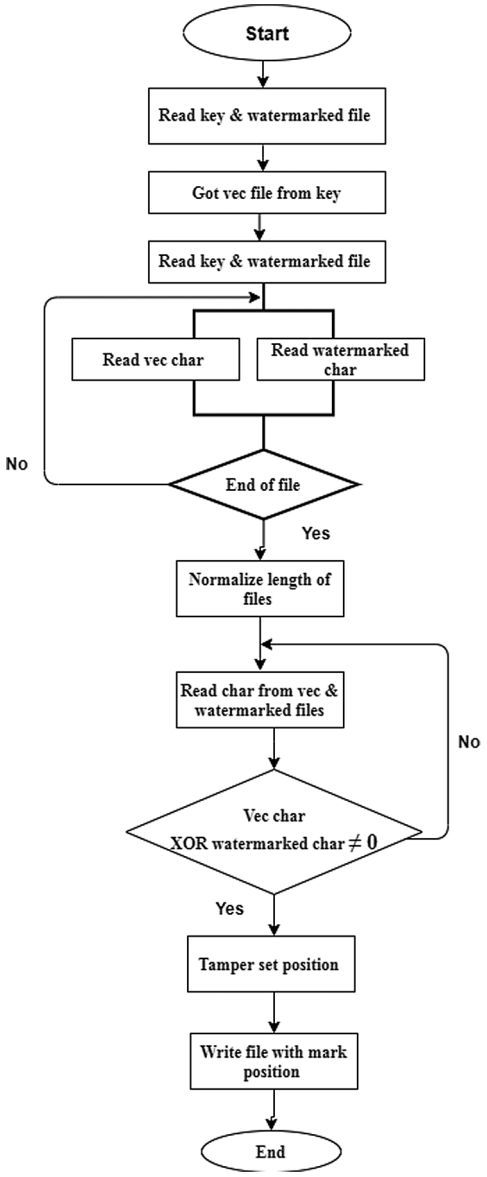

Figure 13: Identification tamper process

Since the MSE double variable adversely affects the double variable PSNR, then there will surely be some inevitable harmful effects. The PSNR parameter allows for the normalizing of the equation for all methods and text types. However, measured either as a ratio or as a(log_{10}) base−10 (decibel) or base−2 (doublings orbits), the logarithmic value of the difference lies between the smallest and largest signal values. For example, the decibel (dB) is a unit used to express the ratio as logarithms, mostly for signal power and amplitude [27].

The second evaluation of the effective performance evaluation of system security for the Quranic text watermarking is the calculation of Normalized Cross-Correlation (NCC) measure. The double variable NCC value reflects the ability of the watermark to withstand against any

```
Input: Vector & Text file
Output: output file
Begin
Declaration parameters;
Get Vec file;
Get text file;
Read both vec & text file simultaneity;
For  I = 1 to length of vec file
       Read char from vec;
       Read char from text;
Set length of vec = length vec;
End for
For I = 1 to length of text file
Read char from text file;
Read char from vec file;
Set length of text = length text;
End for
If length vec > length text
 then
          Set difference = length text;
Else
         Set difference = length vec;
End if
For I = 1 to difference
 Read vec file char by char;
 Read text file char by char;
If vec char [I] XOR text char [I] ≠ 0
  Then
Tamper detection = I;
End if
End for
//write output file
I = 0;
While (not end of text file)
         Read text char;
         I = I + 1;
If I = I
Then   write '' @@ text char @@''
Else
Read text char
End while
End begin
```

Figure 14: The whole identification tamper pseudocode proposed scheme

attack that happened to the watermarked text [28]. The NCC can calculate the similarity between the original watermark $W(i,j)$ and the extracted watermark $W'(i,j)$ when using Eq. (2):

$$NCC = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}(W_{ij} * W'_{ij})}{\sum_{i=1}^{m}\sum_{j=1}^{n}\left(W_{ij}\right)^2} \tag{2}$$

where n and m are the integer variables represents the range of the text (verses or whole text), i *and* j are the integer variables refer to the location of the text and letter in such word.

The range of the NCC value is in between 0 and 1 and the watermark is said to be the best when it is very close to 1, indicating the closeness of the extracted watermark character variable (W') to the original one (W). The NCC value obtained from the DCT or frequency domain is often lower than the spatial domain because the coefficients in the frequency domain always affect the extracted watermark (W'), thus making it lesser [29]. In image watermarking, the NCC very clear but in the proposed text watermarking scheme a substantial difference from the existing scheme must be achieved. The extracted text of the Quran is first separated into verses and then the words take characters. By comparing it with the original text of the Quran (W),

if the watermarked text shows a slight difference then it is considered as an abutter method than other approaches; else, the embedding process must be repeated.

However, the material of data used in the Quranic text dataset was acquired from the public domain via the Internet. The Arabic text was obtained from different reported art-of-the-studies for benchmarking. This dataset was selected due to the enclosure of the sensitive data related to Islam. In addition, surahs of different sizes were chosen for the compression of the host text, benchmarking, and evaluation as explained below.

4 Results and Discussion

The implementation of the proposed scheme has been achieved via the use of MATLAB software. In order to evaluate the performance of the proposed approach, it has been considered the standard and authentic version of Quran datasets from tanzil.net [30]. Likewise, these standards have been used in previous research. This dataset was further verified by experts. Moreover, the standard Quran datasets which have been utilized in this work are selected based on the payload capacity and watermark (Logo). The performance evaluation of this work is performed through using four different payload capacity (Al-Kursi verse 4 Kelo bytes (Kb), Surah Al-A'raf 32 Kb, Surah Al-A'raf and Al-Anbiy 43 Kb, Surah Al-Baqara 58 Kb). Consequently, Watermarking of the Four logos in the binary images of such as the H, UTM, Nike, and لألح(للاح). Are chooses randomly dependent on the sizes. Fig. 15 shows the system interface of the proposed scheme based on tamper detection with the location:

The system is tested by modifying the output file of the watermarked Quranic text to see if the system can detect this case as illustrated in Fig. 16. This includes the modification of the Text of the Quranic text without any sense wherein the text can easily be altered by editing it using any software application.

In Tab. 3, we compare the performance of the proposed scheme in terms of PSNR and NCC values with the existing state-of-the-art methods in the literature. The results revealed that the proposed method outperformed the existing ones in terms of quality and robustness. For a fair comparison, the watermark was used for the testing. The PSNR value for watermark host text was 68.20 dB, displaying about 16 dB quality enhancement compared to the highest reported PSNR value (52.05 dB). In addition, the PSNR value for watermark chess host text was 67.43 dB, indicating a quality increase of 14.5 dB compared to the existing reports (62.94 dB). In short, the improvement was acceptable due to the enhancement of the PSNR by 16 dB over other literature reports.

The NCC value of the proposed method for the watermark host text was 0.9976%, showing the security enhancement of 1.76% compared to the latest related work 0.98%. In (H) host text, the NCC value for the proposed method was higher than the reported finding which is 0.9976%, for the still (H) host text. Thus, the proposed scheme achieved about 1.76% security and accuracy enhancement for the NCC value compared to that reported in the literature.

The results of the measurement performance using PSNR and NCC in this study to improve the accuracy and security are shown in Figs. 16 and 17.

Fig. 16 shows the illustrated results of different payload capacity with a fixed secret watermark of 25 bytes. The difference (83.06 to 86.12) of about 3 dB in the PSNR at the beginning is due to the small text file size where the reversing mechanism played a remarkable role. However, larger text file sizes (which made up the rest of the cases) allowed for the use of the random

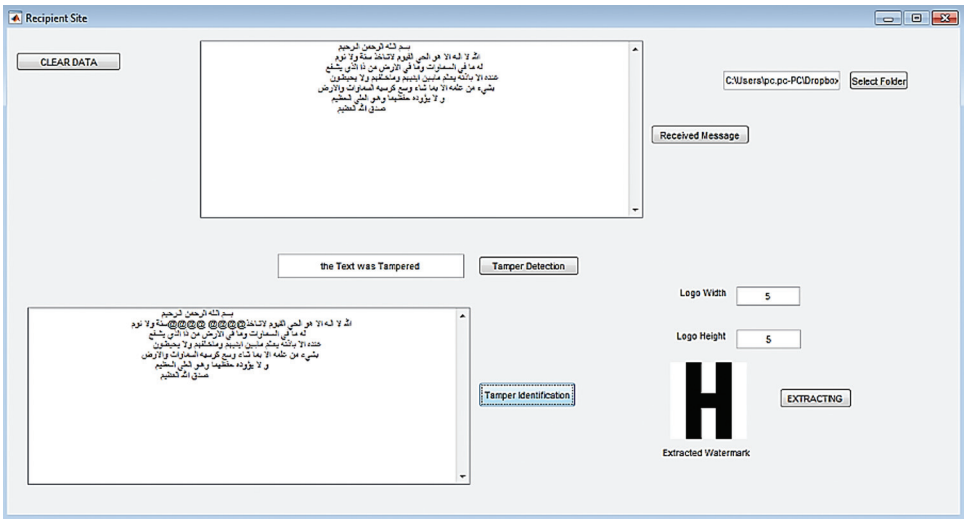

Figure 15: The tamper detection with the first location in the watermarked Quranic text using the proposed technique

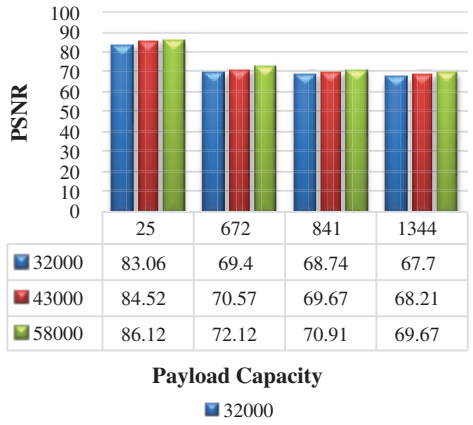

Figure 16: Imperceptibility evaluation using

function among the satisfying conditions; thus, the reversing mechanism played an insignificant role. For large text files as hosting media, more vowels are available to host a secret bit and thus randomly chosen from these letters. The PSNR value is increased with the increase in the file size. The observed sudden increase from one limit capacity to another is because of the excess available space for hosting the secret bits, indicating the embedment of many secret bits and thus making the free space untapped.

Fig. 17 shows the varying payload dependent NCC values. Overall, 1344 bytes has the highest NCC value compared to (25,672, and 841) watermarks. This figure also shows that, although there are three different host text file sizes (32000, 43000, and 58000 bytes) used to embed the host text file and test the security of the proposed scheme, 58000 bytes (grey color) has a better result in all host text file size.

The results of NCC as shown in Fig. 18, posit that the Quranic text processing tools have been distributed over the Internet. Because of this easy and free handling, a sensitive document found on the internet such as the Holy Quran is always under the threat of being tampered with

Table 3: Comparison of security evaluation using PSNR and NCC

Watermark/byte	 6 bytes	 20 bytes
Host size	1000 bytes	1000 bytes
Evaluation scheme	PSNR	NCC
[31]	52.05 dB	NA
[32]	62.46 dB	NA
[33]	NA	0.98%
Proposed scheme	68.20 dB	1.00%
Improvement scheme	9.60%	1.76%

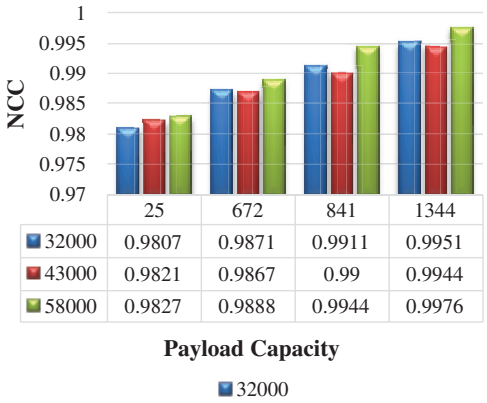

Figure 17: The values of NCC as a function of the different logos with payloads capacity payload capacity obtained using the proposed scheme

by intruders or malicious users, causing it, thereby, to lose its authenticity and originality. For many applications, the verification of the integrity of such data is important. Therefore, many methods have been proposed over the years to verify the authenticity and integrity of the text of the Quran's content. In this spirit, to keep the text of the Quran far from distortion or tampering a robust watermarking scheme has been proposed to detect all kinds of tampering in advance before proceeding to extract the information.

The benchmark of tamper detection of the proposed scheme compared to other researchers work in the tamper detection attack of the Quranic text-domain is shown in Tab. 4.

Tab. 4 shows the benchmark work with existing researchers' work that mentioned there is tamperage or not. All the host apply by all researchers are from Arabic text. The previse work as shown in Tab. 4. Only detect the status of tamper. However, our proposed scheme improved the ability to show the location tampered as shown here in Fig. 18:

Tamper detection is a different procedure from other evaluation criteria in terms of the measures where detection gives the condition of true or false. In the simulation program, the notification is received to ascertain the presence or absence of the tamperage. In addition, can

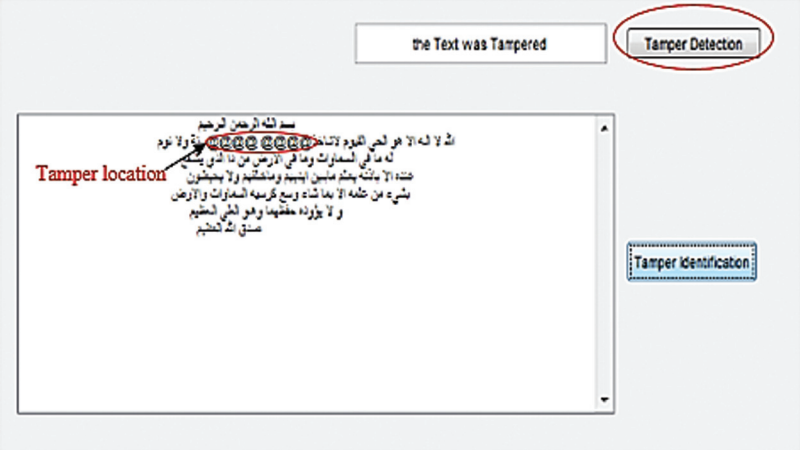

Figure 18: System response against the tamper attack detection

Table 4: Compression evaluation of tamper detection attack

Reference	Host	Tamper detection	Tamper location
[34]	Quran	✓	NA
[35]	Arabic	✓	NA
[36]	Quran	✓	NA
Proposed scheme	Quran	✓	✓

be determined the location of the tamper detection inside the word or statement called tamper identification so this process is working when the text has been tampered with.

Indeed, the efficiency of this scheme that we are focusing on the authentication of the tamper detection and location inside the text watermark than existing works. Therefore, we used 3 datasets to get the best experimental analysis results compared with the benchmark of the currently existing work.

5 Conclusion

The purpose of this paper is to enhance the watermarking scheme of the text of the Quran based on a hybrid technique (XOR and queuing techniques). The methodology consisted of four phases: The first phase was the pre-processing phase followed by the embedding process phase which is performed in order to hide the data after the vowel letters where if the secret bit is "1", it inserts the kashida but does not insert the kashida if the bit is "0". The third phase is an extraction process and the last phase is an evaluation of the performance of the proposed scheme by using PSNR for the imperceptibility and NCC for the security of the watermarking. However, in terms of tampering detection, accuracy falls below deletion, insertion, and reordering attacks. The experimental results revealed the improvement of the NCC by 1.76%, PSNR by 9.6% compared to available current schemes. For the future work, researchers should consider the watermarking of the Quranic text when manipulating the text in different formats (.docx, .rtf, and so on) which was out of the scope of this research. Again, some measures that are not mentioned in evaluating the present scheme might also be considered. Many attacks which are not considered in the proposed scheme must be also be looked into. In sum, it is worthwhile to enhance the security and capacity of the scheme for its further betterment.

Acknowledgement: The authors would like to thank those who contributed to the article and who support them from Universiti Teknologi Malaysia (UTM) for their education.

Funding Statement: The authors extend their appreciation to the Ministry of Higher Education (MOHE) and Universiti Teknologi Malaysia (UTM) for their educational and financial support. This work is conducted at Razak Faculty of Technology and Informatics, under cyber-physical systems research group and funded by MOHE (FRGS: R.K130000.7856.5F026), Received by Nilam Nur Amir Sjarif.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] G. Xin, Qi Xitong and C. Ding, "An improved tamper detection and location scheme for DOCX format documents," *Cloud Computing and Security*, vol. 11, no. 66, pp. 242–251, 2018.

[2] M. Fujimura, K. Imamura and H. Kuroda, "Application of saliency map to restraint scheme of attack to digital watermark using seam carving," in *IEEE Int. Conf. on Consumer Electronics-Taiwan*, Taipei, Taiwan, IEEE, pp. 347–348, 2017.

[3] N. S. Prajwalasimha, A. N. Sowmyashree, B. Suraksha and H. P. Shashikumar, "Logarithmic transform based digital watermarking scheme," in *Int. Conf. on ISMAC in Computational Vision and Bioengineering*, Cham, Springer, pp. 9–16, 2018.

[4] J. Abraham and V. Paul, "An imperceptible spatial domain color image watermarking scheme," *Journal of King Saud University-Computer and Information Sciences*, vol. 31, no. 1, pp. 125–133, 2019.

[5] J. Mayer, P. V. Borges and S. J. Steven, *Fundamentals and Applications of Hardcopy Communication*. Brazil: Springer, pp. 1–5, 2018.

[6] R. A. Alotaibi and L. A. Elrefaei, "Improved capacity Arabic text watermarking methods based on open word space," *Journal of King Saud University-Computer and Information Sciences*, vol. 30, no. 2, pp. 236–248, 2018.

[7] S. Hakak, A. Kamsin, O. Tayan, M. Y. Idris, A. Gani *et al.,* "Preserving content integrity of digital holy Quran: Survey and open challenges," *IEEE Access*, vol. 5, no. 10, pp. 7305–7325, 2017.

[8] A. Alwan, M. Shahidan, N. N. Sjarif, M. Hashim, M. S. Mohd *et al.,* "A review and open issues of diverse text watermarking techniques in spatial domain," *Journal of Theoretical and Applied Information Technology*, vol. 96, pp. 5819–5840, 2018.

[9] A. A. Alwan, M. S. Abdulah and N. N. Sjarif, "A survey on combined various data hiding techniques," *Open International Journal of Informatics*, vol. 7, no. 2, pp. 31–44, 2019.

[10] A. A. R. Alkhafaji, N. N. Sjarif and M. S. Abdulah, "A review of comparative spatial domain techniques of steganography and watermarking," *Journal of Technology Reports of Kansai University*, vol. 62, no. 5, pp. 1–17, 2020.

[11] I. Alsmadi and M. Zarour, "Online integrity and authentication checking for Quran electronic versions," *Applied Computing and Informatics*, vol. 13, no. 1, pp. 38–46, 2017.

[12] Z. Jalil, A. M. Mirza and M. Sabir, "Content based zero-watermarking algorithm for authentication of text documents," *arXiv preprint arXiv*, vol. 7, no. 2, pp. 212–217, 2010.

[13] Y. M. Alginahi, M. N. Kabir and O. Tayan, "An enhanced Kashida-based watermarking approach for Arabic text-documents," in *Int. Conf. on Electronics, Computer and Computation*, Ankara, Turkey, pp. 301–304, 2013.

[14] H. Hai, X. D. Qing and Q. Ke, "A watermarking-based authentication and image restoration in multimedia sensor networks," *International Journal of High-Performance Computing and Networking*, vol. 12, no. 1, pp. 65–73, 2018.

[15] C. Liao, H. Zhong, S. Zhu and A. Squicciarini, " Server-based manipulation attacks against machine learning models," in *Proc. of the Eighth ACM Conf. on Data and Application Security and Privacy*, Tempe, USA, pp. 24–34, 2018.

[16] D. Li, L. Deng, B. B. Gupta, H. Wang, C. Choi *et al.,* "A novel CNN based security guaranteed image watermarking generation scenario for smart city applications," *Information Sciences*, vol. 479, no. 20, pp. 432–447, 2019.

[17] C. Yu, J. Li, X. Li, X. Ren, B. Gupta *et al.,* "Four-image encryption scheme based on quaternion fresnel transform, chaos and computer generated hologram," *Multimedia Tools and Applications*, vol. 77, no. 4, pp. 4585–4608, 2018.

[18] S. Alubi, M. Shehab, M. Al-Ayyoub, Y. Jararweh, B. Gupta *et al.,* " Parallel implementation for 3d medical volume fuzzy segmentation," *Pattern Recognition Letters*, vol. 130, no. 1, pp. 312–320, 2020.

[19] P. Pal, B. Jana and J. Bhaumik, "Watermarking scheme using local binary pattern for image authentication and tamper detection through dual image," *Security and Privacy*, vol. 2, no. 59, pp. 1–16, 2019.

[20] A. Gutub and K. Alaseri, "Hiding shares of counting-based secret sharing via Arabic text steganography for personal usage," *Arabian Journal for Science and Engineering*, vol. 45, no. 25, pp. 2433–2458, 2019.

[21] W. Qi, W. Guo, T. Zhang, Y. Liu, Z. Guo *et al.,* "Robust authentication for paper-based text documents based on text watermarking technology," *Mathematical Biosciences and Engineering*, vol. 16, no. 4, pp. 2233–2249, 2019.

[22] M. H. Kamarudin, C. Maple and T. Watson, "Hybrid feature selection technique for intrusion detection system," *International Journal of High-Performance Computing and Networking*, vol. 13, no. 2, pp. 232–240, 2019.

[23] A. D. Saeed, "Integrity verification & temper detection of English documents using hybrid structural component and word length," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 1, pp. 7073–7078, 2020.

[24] F. N. Al-Wesabi, K. Mahmood and N. Nemri, "A zero watermarking approach for content authentication and tampering detection of Arabic text based on fourth level order and word mechanism of Markov model," *Journal of Information and Applications*, vol. 52, no. 10, pp. 1–15, 2020.

[25] M. Almazrooie, A. Samsudin, A. A. Gutub, M. S. Salleh, M. A. Omar *et al.,* "Integrity verification for digital holy Quran vs. using cryptographic hash function and compression," *Journal of King Saud University-Computer and Information Sciences*, vol. 32, no. 1, pp. 24–34, 2020.

[26] A. A. Shaker, F. Ridzuan and S. A. Pitchay, "Text steganography using extensions Kashida based on the Moon and Sun letters concept," *International Journal of Advanced Computer Science and Applications*, vol. 8, pp. 286–290, 2017.

[27] A. F. Al-Azzawi, "Multi-layer Arabic text steganographic method based on letter shaping," *International Journal of Network Security & Its Applications*, vol. 11, pp. 1–19, 2019.

[28] S. M. Al-Nofaie and A. A. Gutub, "Utilizing pseudo-spaces to improve Arabic text steganography for multimedia data communications," *Multimedia Tools and Applications*, vol. 79, no. 1–2, pp. 19–69, 2020.

[29] A. R. A. Alkhafaji, N. N. Sjarif and M. A. Shahidan, "Digital text watermarking techniques classification and open research challenges: A Review," *Journal of Technology Reports of Kansai University*, vol. 62, no. 5, pp. 1–22, 2020.

[30] H. Z. Zadeh, "Tanzil Quran navigator," 2013. [Online]. Available: http://tanzil.net/#1:1.

[31] M. Elhoseny and K. Shankar, "Optimal bilateral filter and convolutional neural network based denoising method of medical image measurements," *Measurement*, vol. 143, no. 263–2241, pp. 125–135, 2019.

[32] S. Hakak, A. Kamsin, J. Veri, R. Ritonga, T. Herawan *et al.,* "A framework for authentication of digital Quran," *Information Systems Design and Intelligent Applications*, vol. 672, no. 74, pp. 752–764, 2018.

[33] S. Hakak, A. Kamsin, P. Shivakumara and M. Y. Idris, "Partition-based pattern matching approach for efficient retrieval of Arabic text," *Malaysian Journal of Computer Science*, vol. 31, no. 3, pp. 200–209, 2018.

[34] R. G. Sharma, "A vision on text steganography with proper investigation report to identify the associated problem," *International Journal of Computer Trends and Technology*, vol. 54, no. 5, pp. 1–12, 2017.

[35] S. K. Khudhair, *Watermarking Text Document Image using Pascal Triangle Approach*, vol. 5. Malaysia: Universiti Teknologi Malaysia, pp. 1–55, 2014.

[36] M. M. Islam, M. N. Kabir, M. S. Sadi, M. I. Morsalin, A. Haque *et al.,* "Novel approach towards tamper detection of digital holy Quran generation," *InInECCE2019*, vol. 632, pp. 297–308, 2020.