Journal of Theoretical and Applied Information Technology
31st December 2020. Vol.98. No 24

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

CHARACTER PROPERTY METHOD FOR ARABIC TEXT STEGANOGRAPHY WITH BIOMETRIC MULTIFACTOR AUTHENTICATION USING LIVENESS DETECTION

NUUR ALIFAH ROSLAN[1], NUR IZURA UDZIR[2], RAMLAN MAHMOD[2],
ZURIATI AHMAD ZUKARNAIN[2], MOHD IZUAN HAFEZ NINGGAL[2], REEMA THABIT[3],
[1,2,3]Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Malaysia
E-mail: [1]nuuralifahroslan@gmail.com, [2]{izura, ramlan, zuriati, mohdizuan}@upm.edu.my, [3]rbinthabet@gmail.com

ABSTRACT

Arabic text steganography (ATS) offers a potential opportunity in hiding secret information in characters and features. The combination with any other security sub discipline such as cryptography usually will enhance its level of security. However, it is limited in its ability to optimize embedded data capacity with a high perceptual transparency level that will also not raise suspicion when written. Besides that, other concerns are active attacks by intruders which are a crucial security issue in the transmission of the shared secret key that enables the receiver to extract the secret information. Also, such attacks can be affected through a fake identity that allows the receiver to modify the secret information thus degrading its integrity. To overcome these drawbacks, we propose a hybrid ATS with biometric multi factor authentication (BMA), which uses liveness detection using fingerprints and heartbeat sensors as the authentication factors. We propose a new ATS method, the Character Property method (CPM) which uses the basic properties of the Arabic Text such as dots, calligraphy typographical proportions, and sharp-edges to hide the secret message using a table index mapping technique to optimize data capacity with high perceptual transparency to avert suspicion. The results for the biometric authentication showed that the proposed method correctly authenticates users, having a false rejection rate of only 4%, and a 0% false acceptance rate. As for liveness detection, the results were significant where the proposed method correctly detected live subjects compared to a fingerprint only biometric authentication approach, which had a high acceptance of fake inputs. BMA was implemented through a custom Arduino smartwatch with a fingerprint and heartbeat sensor as a 'proof-of-concept' device which increased the capacity in hiding the secret message up to 23.5% compared to the previous methods. Given our Arabic Character Properties method (CPM) did not affect the stego-text appearance, its 1.0 Jaro Similarity score was compared to the other methods proving high transparency of the stego-text, in addition to higher security regarding user authentication using BMA with liveness detection.

Keywords: *Arabic Text Steganography, Biometric Multifactor Authentication, Information Hiding*

1. INTRODUCTION

Steganography is defined as "covered writing" and is derived from the ancient Greek root word steganos (στεγανός) meaning "covered or protected", and graphei (γραφή) meaning "writing". Throughout history, most steganography applications were intended for military purposes that employed this covert form to communicate and to relay important secret and sensitive information.

Accordingly, steganography consists of three main components: the embedded data which is the message one wishes to send secretly; the cover-object as the media being used to hide or conceal the data; and the stego-object as the product for using the cover-object to hide the embedded data. In classic or pure steganography, the cover-object can be any medium and represented in the following formula:

Secret Message + Cover-Object = Stego-Object

In modern steganography, the sender uses the digital cover medium such as digital text, images, audio, or video, which may be transmitted via the internet publicly to the receiver. The receiver will

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

retrieve the cover-object and conceal the secret information using the designed steganographic algorithm via certain software.

Indeed, various languages that exist lead to different language styles and script writing. As such, this presents us with the beauty of text steganography, which has various types of linguistic texts whereby each language has its own form of grammar rules. Moreover, each language has its own script for writing. However, through our review, most linguistic steganography have limited capacity to be optimised given the limitation in maintaining the grammatical structure of the sentence.

Therefore, non-linguistic text steganography is not restricted to Latin alphabets but can also be implemented with other text document scripts such as Chinese [1], Tamil or Hindi [2], and Thai [3]. Indeed, each script writing has its own advantages that can be of benefit in hiding the secret information.

In Chinese script writing characters have two forms (simplified and traditional Chinese characters) and are selected and substituted in order to hide data in the carrier. Ali [4] used the Mytra in the Hindi script as the indicator to hide the secret bit. Whereas, Thai text steganography exploits redundancies in a particular vowel, compose of diacritical and tonal symbols in TIS-620, and uses the standard Thai character set as the means to hide the secret information [5].

One of the most active research areas using the scripts of writing is Arabic text steganography (ATS). However, the research in this area is still evolving through the application of better techniques and algorithms to achieve high capacity text steganography. There remain numerous potential applications using Arabic characters that can be manipulated in having a high capacity to embed secret information with a high degree of transparency. Therefore, we designed a new method of Arabic Text Steganography (ATS) with a high-security level since there could be a flaw regarding the level of security.

This paper focuses on the Arabic Text Steganography algorithm which optimized the capacity of the embedded secret information and increase the perceptual transparency of the cover text only and the robustness against the setego-text modification only.

The remainder of this paper is structured as follows: Section 2 presents related works in hybrid steganography, previous works related to hybrid biometric steganography and previous works in ATS. Section 3 describes the proposed framework, while the experiment carried out is presented in Section 4. Section 5 presents the results of the study and analysis of each conducted experiment, followed by in-depth discussion and conclusions.

2. RELATED WORKS

2.1 Previous Works in Hybrid Steganography

Hybrid steganography can be described as a combination of the steganography algorithm with other sub-disciplines in security such as encryption, hash functions, or public keys. The purpose of the hybrid depends on the nature of the issue that needs to be addressed. The following section discusses the hybrid under the various types of steganography.

The combination of RSA encryption with image steganography has been presented in Kusuma *et al*. [6]. Here, the secret image pixels apply Arnold's transformation to randomise it before being encrypted. RSA encrypts the randomised pixels, embedding it into the cover image. The LSB steganography changes the 2 bits of pixels of the carrier image by 2 bits of the secret message.

This technique makes it almost impossible for an attacker to determine the secret information and conceal the secret message.

The proposed method in Okediran *et al*. [7] improves the security of an e-voting system. The proposed method encrypts the ballot information using the merging of AES and RSA to overcome the safety, privacy, and verifiability of an e-voting system. The encrypted ballot is embedded into an image using LSB image steganography to protect the encrypted ballot.

Besides cryptography, the perfect hashing algorithm seeks to strengthen the process of hiding the secret information in image steganography [8]. Here, the used algorithm will randomly generate a hash-key which is then used by the algorithm to generate a

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

pattern of pixels where the data is to be stored. Through the combination approach, the embedded data will create new integrity of the secret message pattern that makes the hiding of data extremely efficient and, further, enhances the integrity of the secret message.

Hybrid steganography is also covered in text steganography, where the algorithm utilises Cascading Style Sheets (CSS) using End of Line (EOL) on each of the CSS style properties immediately after a semi-colon. Under this approach, and before being embedded into the cover-text, the message is initially encrypted using an RSA algorithm as a public key cryptographic algorithm and is then transmitted to the receiver [9]. Even though this approach has high security in protecting the secret message from active attacks, the capacity for hiding the secret message is limited.

Meanwhile, the image steganography algorithm utilises shared secret key steganography protocols where the secret message is encrypted and is then embedded in the cover media using the LSB substitution technique. The hybrid might be more secure as the secret message is compressed before being encrypted [10]. DNA steganography is a distinct research direction of DNA cryptography [11], where it applies an encryption technique to code a message in the DNA and then uses a simple substitution cypher to encrypt the characters in the DNA triplets.

Aside from images and text, audio steganography (AS) also increases its security via a hybrid of the AES encryption and Message Digest 5 (MD5) Hash Function. Here, the algorithm encrypts the secret message through an AES algorithm that uses a key which has been processed by MD5. Through this combination, the level of security for the secret message is improved. In other words, the combination is successfully achieved without diminishing the quality of the used mp3 files [12].
Last but not least, in video steganography which is a hybrid of RSA encryption and the key-based random algorithm employing the sequential encoding method, is used to overcome the limitation of the sequential encoding itself. The limitations in using sequential encoding are that intruders can identify the presence of hidden messages by sequentially analysing the video frames [13].

The combination of Data Encryption Standard (DES) cryptography and Discrete Cosine Transform (DCT) steganography techniques are proposed in Solichin and Ramadhan [14]. Here, the secret message with the password converted to cypher text is by using the DES algorithm. The cypher text embedded into an image using the DCT algorithm generates a stego image which is sent to the receiver via a public channel.

In summary, the review shows that a hybrid between steganography and any element of data security can be implemented with any type of steganography, such as an image, text, audio, or video steganography. Most combinations employ encryption and hashing algorithm to enhance the level of security against active intruder attacks.

Regarding our prior ATS research regarding the Primitive Structural Method (PSM) [15], we can conclude that most of the ATS algorithms are mostly concerned in having a high capacity in hiding the secret message, and most are fragile to active attacks such as using a fake identity of the sender and the receiver that can lead to intercepting or modification of the secret information.

As such, given this concern, the hybrid of text steganography with other security elements besides encryption or hashing such as biometric authentication may help to solve the problems as mentioned above. The next section will look closer at previous works in the Steganography Biometric hybrid.

2.2 Previous Works in Steganography Biometric Hybrid

Many of the approaches in the steganography biometric hybrid aim to secure the biometric information of the user employing the steganography approach since steganography offers secrecy transmission via a public network. The authentication framework with steganography will enhance the robustness of the authentication system (remote authentication station), such as the work conducted by Natilianis [16].

Natilianis proposed a robust authentication mechanism based on semantic segmentation, chaotic encryption, and data hiding. Here, the segmented video object is automatically segmented, using a head-and-body detector. Then, the biometric signals are encrypted by a chaotic cypher.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Afterwards, the encrypted signal is embedded to wavelet coefficients of the video object, using its qualified significant wavelet trees (QSWT) [16]. Finally, the inverse discrete wavelet transform is applied to provide the stego-object and authenticates the user.

Moreover, some algorithms use a face-geometry authentication system employing the distance and angle features of facial objects such as the face, nose, and eyes. The stego image, which contains the embedded face-geometry calculation information, is used for user authentication. The facial geometry extraction information is embedded in the stego image and stored in the database [17]

In another study, Das *et al.* [18] presented a Scale Invariant Feature Transform (SIFT) for extracting lip features, which includes a pre-processing step. The biometric authentication uses lip focus on both the recognition rate along with securing the templates stored in the biometric system. A spatial steganographic algorithm is employed on the lip images to ensure minimum distortion in addition to hiding the identity of the lip images in the images themselves, thus ensuring less chance of misuse of the template.

Besides the SBH for enhancing the security of the biometric authentication framework, the hybrid steganography approach offers advantages to the steganography algorithm itself. The biometric skin approach [19] uses the skin region as the cover image. Here the secret data is embedded within the skin region of the image that offers an excellent secure location for data hiding.
Image steganography integrated with lip biometric, as presented in [18] protects the secret object. The lip image is improved by applying a local interest algorithm where the lip features are extracted using the SIFT. Here, image steganography is used to hide the secret object in the lip image using a spatial modulation method.

On the other hand, Sivaranjani and Radha [20] presented biometric authentication for the confidentiality of individual data using the steganography approach employing an Electrocardiogram (ECG) as the biometric information. As such, the ECG is highly confidential and secure, making it difficult to be forged, whereby the signal is used as the host signal, which will carry the secret patient information. They applied the steganography technique, in which the secret patient information and physiological readings would be embedded inside the ECG host signal.

Furthermore, at the hospital server, the ECG signal, and its hidden information are stored. Any doctor can view the watermarked ECG signal and only authorised doctors, and certain administrative personnel can extract the secret information and gain access to the confidential patient information in addition to other readings stored in the host ECG signal [20].

In a separate study, image steganography was integrated with hand geometric [21], where the secret message is embedded into the cover message based on LSB steganography. The embedding method utilises the features of human geometrics to improve the security level of the steganography method.

Through this review, hybrid steganography and biometric authentication are mainly designed for two primary purposes: authentication for the sender and receiver of the steganography part, and to secure the captured biometric information by hiding the biometric information using the steganography approaches.

Additionally, the hybrid of the biometric authentication will provide a solution to our main concern regarding the fake identity of the sender and receiver. The following section will critically discuss the pros and cons of the ATS algorithm.

2.3 Previous Works in Arabic Text Steganography (ATS)

One of the most active research domains in using feature methods script text steganography is regarding ATS, which continues to evolve through the use of better techniques and algorithms to achieve high capacity text steganography. Most research focusing on the feature method is on Arabic linguistics given the properties of the Arabic script, such as the shape of the Arabic character, the dots, and the writing forms.

Most ATS improvises on the feature method by solving the drawbacks of this approach via Unicode encodings in designing the algorithm. This

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

approach results in a high degree of perceptual transparency of the secret information due to the undetectable changes that can be observed through the naked eyes for creating the stego-text. This section discusses the various current techniques in ATS.

The merging of small space characters with Kashida characters is to hide the confidential bits. The chosen small spaces are thin, hair-like and Six-PRE-EM, and sensitive bitstream are divided into blocks of 4-bits. The first bits in the blocks signify the Kashida character, while the remaining represent three small spaces. The existence of the character in the cover-text hides "1" while their absence hides "0". In a similar scenario, pseudo-space and the doted feature are utilised to embed the secret bits in [22], [23].

One of the most active research domains in using feature methods scripts text steganography is in ATS, which is still evolving through the application of better techniques and algorithms to achieve high capacity text steganography. Most of the research focusing on the feature method is towards Arabic linguistics given the properties of the Arabic script, such as the shape of the Arabic character, the dots, and the writing forms.

Combining Kashida with diacritics to conceal the sensitive bits was proposed in Alshahrani and Weir [24]. Here, the sensitive bits are divided into two groups; the first group embeds "1" by inserting Fathah, and the remaining diacritics are inserts to embed "0". The second group inserts two Kashida to embed "1" and one Kashida to embed "0". Also, Kashida is merged with secret sharing to improve the level of security. In the proposed study in Gutub and Alaseri [25], the secret-sharing bits were embedded into Arabic text.

The proposed study in Al-Oun and Odeh Alnihoud [26] embeds audio into a text file employing Kashida and the word "La". The proposed embedding algorithm embeds "1" by adding the next Kashida character while "0" is embedded by the untouched letter Kashida addition.

Recently, an algorithm presented by Ridzuan, and Ali [27] was introduced to hide sensitive bits into Arabic text using sun and moon letters archived in four ways. First, a Kashida is added next to a sun letter to embed the sensitive bits "00". Second, a double Kashida is added next to a sun letter to embed the sensitive bits "11". Third, a Kashida is added next to a moon letter to embed the sensitive bits "01". Lastly, double Kashida is added to embed the sensitive bits "10".

More recently, Malalla and Shareef [28] proposed a modified Fathah in ATS, in which the algorithm is based on the existence of diacritics in most of the Arabic letters. In this approach, secret bits were embedded in the text using a merging of the diacritics method and AES encryption. First, the sensitive information is encrypted with the AES algorithm, and then text steganography with modified Fathah is utilised to embed the encrypted data.

The modified Fathah resides in the same direction with the original Fathah, slightly oriented, like the original to avoid suspicion.

As the review shows, it is extremely challenging to have high capacity text steganography with high invisibility. Moreover, high perceptual transparency is important to secure the steganography communication from passive attacks in addition to active attacks which involves the modification of the secret information. Our review of ATS suggests that most of the research manages to optimise the hiding capacity of the message but with limitations remaining as well as noted disadvantages.

Most prior research can only hide or conceal a

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

maximum of 4 bits per Arabic character. To improve the hiding capacity of the secret information in the cover-text, we proposed the Character Property Method (CPM) in which a maximum of 9 bits per Arabic character can be hidden.

To acquire a high capacity for hiding, a suitable algorithm of steganography must have high perceptual transparency of the stego-object. The previous research uses a Unicode to map between the cover to create a transparency. However, this technique requires having a random mechanism to place the secret information, which makes the secret information challenging to conceal.

In this study, we designed symmetric key text steganography with maximum-order positioning mapping with the Unicode and the cover-text to create a random position of the secret information based on the secret key used. Regarding the previous research on hybrid steganography and other layers of security, the focus is mainly on strengthening the security of the secret information to prevent active attacks such as concealing or modifying the secret information. Since most of the steganography algorithm is symmetric, it is important to secure the secret key from being stolen or taken by an imposter.

Therefore, we propose a new text steganography framework; Character Properties Method (CPA) with Biometric Multifactor Authentication (BMA) with liveness detection to address and overcome the fake identity issue of the sender and receiver of the text steganography communication which might lead to active attacks such as modification of the secret information. The proposed framework is further elaborated upon in the following section.

3. CHARACTER PROPERTIES METHOD WITH BIOMETRIC MULTIFACTOR AUTHENTICATION

Our proposed CPM, along with BMA Framework, is shown in Figure 1. There are two main modules; the Multifactor Authentication Module and the ATS Module.

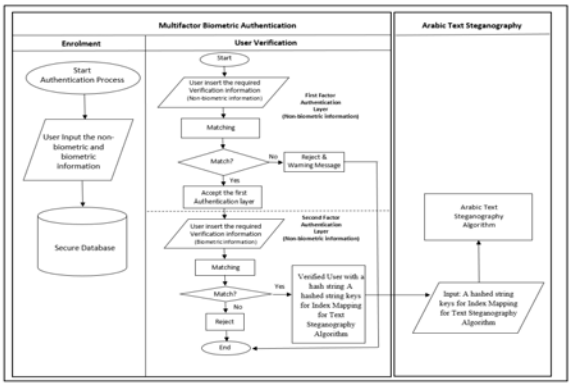

Figure. 1. Character properties method with the biometric multifactor authentication framework

3.1 Multifactor Authentication Module

Authentication is a process of verifying whether the digital identities of computers and the physical identity of people are authentic [29]. There are three types of authentications and their independent elements for verifying a user. The Proof of Knowledge refers to something that the user knows, such as a password, Personal Identification Number (PIN), mother's name, phone number, etc.

The second independent credential is Proof of Possession which refers to something that the user has, such as an ATM card, Smartcards, Tokens, PKI certificate, etc. The final independent credential is the Proof Characteristics of the user which is what the user is and refers to the physiology (biometric information) or behaviour of the user such as a fingerprint, hand gestures, facial image, iris, retina, voice, DNA, or signature pattern.

Fingerprint biometric authentication refers to fingerprints used as input for biometric authentication. Fingerprints are made of a series of ridges and furrows on the surface of the finger which has a core upon which patterns like swirls, loops, or arches are curved to ensure that each print is unique [30].

Besides the unique pattern of the fingerprint that indicates a person's uniqueness, fingerprint

authentication is easily implemented and cost-effective compared to the other biometric elements. Hence, in this study, we use fingerprints as the selected biometric element for the proposed framework.

However, the most serious issue in fingerprint authentication is spoofing by imposters who create a fake fingerprint and pretend to be the enrolled user [31]. Therefore, given these concerns, it is important to have liveness detection in the authentication stage to prove the fingerprints belong to a living individual [32]. As such, we propose multifactor authentication with liveness detection using the heartbeat to measure the liveness of the fingerprint as one of the best solutions to overcome the fake fingerprint issue.

The designed fingerprint authentication was based on hardware-oriented authentication, using the designed authentication sensor hardware with a DCP microprocessor and silicon materials that were manufactured based on the embedded fingerprint recognition system method which specifies the system's hardware and software design process [33].

Regarding Figure 1, for the Multifactor Authentication Module, there are three main algorithms involved in the framework design: the enrolment Module, First-Factor Verification Module (Non-biometric information) and the Second Factor Verification Module (Biometric Information). The authentication process begins with the Enrolment Module.
The enrolment requires all the authentication features, which are the non-biometric (e.g. username, password, and International Mobile Equipment Identity (IMEI)). Meanwhile, the biometric information is the fingerprint and the heartbeat. The following figure shows the algorithm for enrolment of the user:

INPUT: Username, password, IMEI, fingerprint & heartbeat
Read inputs
IF No Inputs Error
    Register the inputs by stored inputs into database
ELSE Displays warning message to reregister the inputs
End

Figure. 2. Enrolment algorithm.

Next is the verification in the first-factor authentication. Through this verification module, the algorithm will read the username, password, and IMEI device and proceed with matching the information stored in the database. If the inputs match, the authentication, it will proceed to the second-factor verification module, which is the biometric authentication. Figure 3 depicts the algorithm for the first-factor verification module.

INPUT: Username, Password, IMEI
First Level Authentication Verification
Read Non Biometric inputs; Username, Password, IMEI
IF inputs match with the database data
    User is authenticate and may proceed with the second level authentication
    ELSE Reject the user with warning message
End

Figure. 3. First-factor verification algorithm

Next, to have both authentications carried out concurrently, we combined the fingerprint hardware with the heartbeat sensor hardware as we designed the authentication verification based on the following algorithm to capture an authenticated user with liveness proof. The proof-of-concept of the device was designed using C++ and the Arduino as the processor. Figure 4 simplifies the second factor of the verification of the algorithm.

START
READ ZFM20 Fingerprint sensor
ZFM20 Fingerprint sensor capture the image and computes an ID_finger for the image to be compared with the created ID_num at
t
he enrollment phase
READ a Heart-beat sensor
Heart-beat sensor captures the value of the heart beat; ID_heartbeat
IF ID_finger sensor && ID_heartbeat Heart-beat sensor match in thedatabase, verification is successful
Return A hash-string keys for Index Mapping for the Text Steganography Module
ELSE User not authenticated
END

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

INPUT: Biometric inputs; Fingerprint & Heartbeat **Read Biometric inputs;** Fingerprint & Heartbeat
IF fingerprint match with the database data & detect liveness via heartbeat sensor
User is authenticate and may proceed with the steganographic communication
ELSE Reject the user with warning message
End

Figure. 4. Second-factor verification algorithm

Concerning Figure 4, the matching process of the fingerprint is carried out internally via the fingerprint sensor before the matching ID is created and compared to verify the fingerprint. As for the heartbeat verification, it is based on the medical definition between 60 bmp and 280 bmp [35], for the liveliness detection rate to be deemed accurate. The reading of 0 bmp will indicate death or a non-living entity.

Once the verification process has been verified, a return value of a hash string will be one of the embedded inputs to create a new text steganography algorithm which will proceed in the second ATS Module. The following subsection elaborates on the proposed CPM for the ATS module.

3.2 Arabic Text Steganography Module: Character Property Method.

The CPM uses the three features of Arabic characters such as sharp edges, dots, and calligraphy, proportioned as a place to hiding the secret information bits. This creates more possible places for hiding the bits per Arabic character compared to the previous research on ATS. In addition to having a high capacity for hiding, a suitable algorithm of steganography must have high perceptual transparency of the stego-object.

We designed a symmetric key text steganography with maximum-order positioning mapping with the Unicode and the cover-text to create a random position of the secret information based on the secret key used. Concerning the previous research on hybrid steganography and other layers of security, most of the focus tended to be directed towards strengthening the security of the secret information in order to prevent active attacks from occurring, such as concealing or modifying the secret information. Since most of the steganography algorithm is symmetric, it is important to secure the secret key from being stolen or taken by an imposter.

Therefore, we designed the BMA as a hybrid with our CPM with ATS to address this angle of active attacks in preventing unauthorised persons from accessing the secret information. The designed BMA concerns the liveness issue and counter fingerprint spoofing issues. As such, including heartbeat sensors with fingerprints as biometric authentication credentials will assist in improving the liveness of the authenticated users.

The CPM involves three main characteristics of the script, namely the sharp edges, dots, and the typo proportion, as illustrated in Figure 5. The sharp edges are defined as the sharp starting, and end points of an Arabic character.

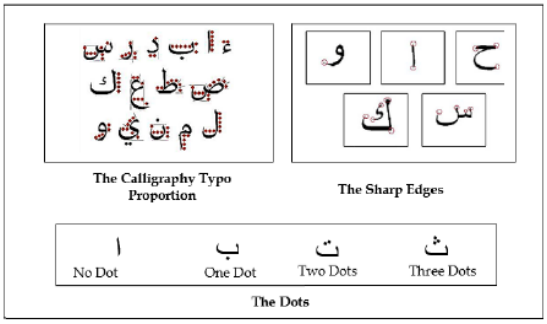

Figure. 5. Arabic script characteristics for the Character Property method

Each element of the Character Property such as the sharp edges, dots and calligraphy typo proportion indicates a possible place for hiding secret information. The dots refer to their respective number in the Arabic character, which there are 15 having one, two, or three dots. The typo proportion is adopted from the calligraphy writing method in which a rhombic dot is the shape formed when a calligrapher presses the pen to paper in one downward motion, producing a diamond shape.

In addition, each Arabic character has its own Character Property element. For example, the character ت (ta) has two sharp edges, two dots, and three lengths for the typo proportion, thus providing seven potential hiding places. Each element has its relative potential number of places for hiding the secret information, and each hiding place corresponds to a bit of the secret message.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

For example, seven bits of a secret message can be hidden in the character ت (ta) since it has seven potential places for that purpose. Therefore, each character provides more than one place to hide the secret bits. This offers a significant advantage to the algorithm in having a high capacity concerning the embedded secret bits. The main architecture of the CPM consists of two main modules: the hiding and the retrieving modules. Figure 6 schematically illustrates the main flow of the CPM.

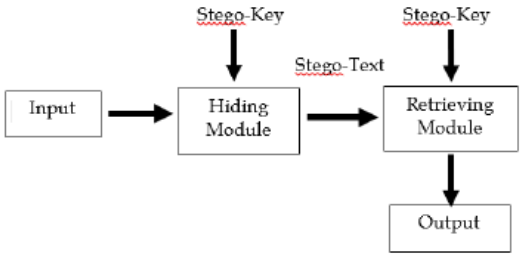

Figure. 6. Architecture of Character Property method

As shown in the above figure, the input is the secret message and the cover-text. The designed CPM requires stego-keys for the hiding modules. The hiding modules will result in a stego-text which can be published publicly for steganographic communication. The retrieving module extracts the secret message using the stego-text and requires the same text in the hiding modules. Once the key is recognised using this method, it will retrieve the secret message based on the hard-coded reference table, and the extracted secret message is the final output. The following section provides a further description of both modules and how the CPM function.

3.2.1 Character Property method Hiding Module

The hiding module is used by the sender to hide the secret message. The module starts with an input from the sender comprising the key 1 or the secret message (English alphabet) and the cover-text (Arabic text). Key 1 is a secret shared key which is agreed to by both the sender and receiver and determines the generated unique number. Key 2 is the generated key through the algorithm process, which needs to be used by the receiver to retrieve the secret message.

This hiding module consists of two major processes, (i.e., the validating input and the bit hiding). A valid process is indicated if the cover-text has a larger number of Character Property elements compared to the length of the secret bits. Once valid, it proceeds to the main process of the algorithm, which constitutes the Bit Hiding process.

3.2.1.1 Bit Hiding Process

The Bit Hiding process is the main algorithm for the CPM and is based on the mapping between the index referencing and the different lengths of the secret bit base. This main process consists of four important algorithms:

1) Character Property Algorithm;
2) Maximum Positioning Algorithm;
3) Secret Key Embedding Algorithm; and
4) Hashing Process Algorithm.

The inputs for the bit hiding process are retrieved from the previous validation process, such as Key 1, the secret message, and the cover-text. The process begins with the Character Property algorithm where the process will read the cover-text as an input, and then identify the total number of potential places for hiding each character.

For example, if the cover-text is "كتـــاب" in a word. The word "كتـــاب" has five Arabic characters, (i.e., ك, ت,ا and ب.). By referring to the Character Property table, each character presents its own number of Character Property elements as the output of the process, and the information of the output will be used in the Maximum Positioning algorithm. The maximum positioning algorithm sorts the output from the Character Property algorithm.

The sorting process begins with the maximum value and will not alter the cover-text position but instead, be a reference value for concatenating the secret bits. Then, by referring to the secret input bits, the algorithm will concatenate the bits based on the number of the Character Property for each of the Arabic characters.

Next, the alit performs the maximum positioning process, which starts with ك, ت, ب and ا based on the descending order of the number of the Character Property element. Then, the algorithm reads the secret bit, which will concatenate based on the number of Character Property elements. Since the number of the element indicates the potential place to hide the secret bits, it will begin

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

to concatenate with the first nine secret bits, then seven, and lastly the two secret bits. Once the process has been completed, it will proceed to the Secret Key Embedding algorithm.

The Secret Key Embedding algorithm is the process upon which the shared secret key is used to create a unique random number based on the positioned concatenated secret bit in the previous algorithm. The shared secret key is key 1, which is agreed to between the sender and the receiver and can be in the alphanumeric string. Next, it will be converted into a binary form and its length calculated. The binary length will be the trigger to generate a unique number for each secret message's binary pattern position where it will be stored in the Random Unique Number (RUN) table.

The RUN table consists of three main elements for each cover-text, (i.e., the sum of properties, the pattern position of the binaries, and the unique index number). A unique random number is then generated for each pattern position. The Secret Key Embedding algorithm will produce the final stego-text and the secret Key 2 for the sender to conceal the message.

The stego-text can be published publicly, but the secret key 2 only can be shared with the sender to conceal the secret message. The output from this algorithm is the stego-text, key 2 and the Lookup Table, which will be used by the receiver to conceal the secret message. The retrieving module will then converse with the hiding module, by reading Key 2 and Key 1 based on the generated Lookup table following which the secret bit binary will be concealed. Finally, it will be converted to alphanumeric to return the secret message.

The designed CPM with a BMA for ATS has the vast potential to be implemented. As the implementation of the designed architecture shows, it is mobile for steganography communication.

4.0 EXPERIMENTAL DESIGN

The conducted experimental design was divided into two parts, namely BMA experimental design and the ATS experimental design to evaluate the entire framework based on their benchmark.

4.1 Biometric Multifactor Authentication Experimental Design

There were two experiments undertaken which included user acceptance evaluation and the liveness evaluation experiment. The method employed for evaluating user acceptance is referred to as the biometric authentication device evaluation [36]. The second experiment was the liveness detection experimental design aiming to evaluate liveness detection performance against the spoofing inputs. The following subsection describes the settings for each experiment, preparation of the apparatus and materials, and performance measurement for the evaluation.

4.1.1 Biometric Authentication User Acceptance Rate Experimental Design

a) Participants

The participants in this experiment consisted of 50 students (25 of each gender) from Universiti Putra Malaysia (UPM) with ages ranging between 19 and 23 years. All participants had previous experience with passwords and using a PIN as an authentication method.

b) Apparatus and Materials

1. Hardware

The participants used the developed system smartwatch Arduino device for the biometric authentication experiment.

2. Client Software

The client software was developed using Java Language as a standalone application. The application was installed on a laptop as a localhost server, and communication between the hardware and software was via a Bluetooth connection.

3. Acceptance Criteria

The acceptance fingerprint and the heartbeat input samples were quality-checked by the designed device.

i) Password:

The participants chose a six-character alphanumeric password, including symbols indicating a strong password.

ii) IMEI:

The device was a fixed IMEI (990000862471854).

iii) Fingerprint and Heartbeat:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The selected finger of the person surface needed to be in good condition, as the sensor would only capture a perfect-image quality fingerprint. The heart rate of the participants needed to be within the normal range between 60 and 150 bpm, before conducting the experiment.

c) Procedure

The experimental setup was conducted under conducive conditions in a large open-spaced hall. Once the proof-of-concept device was set up, the participants were enrolled and registered using the designed software. They then chose their PIN number and provided the device's IMEI and the biometric information comprising the fingerprint and heart rate, which would be captured. Next, the user authenticated the system five times with true information with each successful or failed authentication recorded, thereby producing 250 samples for the results. A total of 30 attempts were wrongly authenticated to create an imposter attempt.

4.1.2 Liveness Detection Experimental Design

The objective of this experiment was to measure spoofing scores to demonstrate and prove the importance of including the liveness element for the fingerprint authentication in preventing fingerprint spoofing by fake samples. The easiest way to create a fake fingerprint is by using the printed fingerprint on transparent paper. Although, a more successful method is to create a 3D fake model with the fingerprint stamped on it which can be achieved by creating a mould that is filled with a substance (silicon, gelatin, plasticine, wax, glue, plastic) and is used to create a thick or thin mould that an intruder can use [37].

The following describes the experimental requirements and protocols in performing the liveness detection evaluation:

a) Participants

The participants comprised of 50 students (25 males and females) from UPM, aged between 19 and 23 years.

b) Apparatus and Materials

i. Hardware & Client Software

The proof-of-concept hardware device and software used in this experiment had the same setup as with the previous experiment.

ii. Fake Fingerprint Samples

The fake fingerprint was created using plasticine as the mould and silicon as the cast. Each participant created two samples using two different fingers, producing 100 samples for the experiment.

iii. Procedure

As the previous experiment setup had already registered the participants, they were authenticated in the system using their fake fingerprints.

4.2 Arabic Text Steganography Experimental Design

The main performance measurements attributed to text steganography relate to the capacity of the embedded secret information and transparency or perceptual transparency of the secret information providing the level of secrecy of the stego-text when publicly exposed [38].For this purpose, an experimental design for capacity and transparency evaluation was carried out to measure and compare the CPM. Further, we added robustness against copy, scanning and printing for the cover-text as the performance measurement determiner for text steganography

4.2.1 Text Steganography Capacity Evaluation

Two sets of experiments for capacity measurement were undertaken. The first experiment was to compare the use of the character in the cover-text via the capacity evaluation approach by Gutub [39]. The second experiment compared the capacity percentage of the embedded secret information with other previous works in ATS. The capacity experiments started by preparing the data set and then applying it using the system and calculating the embedded capacity using the standard capacity performance measurement equation.

In this context, capacity refers to the ability of a cover media to store secret data and can be measured by the amount of secret data (bytes) that can be hidden in a byte of a cover media [39]. The bitrate or capacity is defined as the size of the hidden message relative to the size of the cover [40]. Therefore, capacity (C) is determined as expressed in Equation 1:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$C = \frac{\text{bits of secret message}}{\text{bits of stego cover}} \quad (1)$$

4.2.2 Text Steganography Transparency Evaluation

To measure the perceptual transparency of text steganography, similarity metrics were used to gauge the distance between two values or sequences using the Jaro-Wrinkler metric (Jaro Score) [41].

The Jaro-Wrinkler metric was used to measure the similarity between the cover and stego files since the CPM deals with a string. The Jaro-Wrinkler metric is calculated as given in Equation 2:

$$d_j = \begin{cases} \frac{1}{3}\left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \right. \\ \end{cases} \quad (2)$$

4.2.3 Text Steganography Robustness Evaluation

Robustness refers to the ability to protect the unseen data from corruption, especially when transmitted via the internet [39]. As shown via the previous review, most drawbacks attributed to ATS are related to robustness, due to modification or tampering such as via the Optical Character Recognition system (OCR), file formatting of the text or retyping the cover-text data since most are based on feature method categories.

Therefore, the robustness experiments, the cover-text was tested against the OCR, a modification through file formatting and retyping to ensure the embedded secret information would remain intact and would not be affected.

The CPM to test the cover-text was based on the robustness compared to tampering via the OCR system, file formatting of the text or retyping the stego-text data. Through this experiment, 50 stego-texts were implemented in three robustness evaluations, as detailed in Table 1.

Table 1: Robustness Evaluation

Experiment	Experimental Steps
OCR	Step 1: Cover-Text scan through the OCR system. Step 1: Cover-Text scan through the OCR system. Step 2: Retrieve back secret message into the Character Property Algorithm.
File Formatting	Step 1: Save the Cover-Text in various type of format. Step 2: Retrieve back secret message into the Character Property Algorithm.
Retyping Cover-data	Step 1: Retype the Cover-Text. Step 2: Retrieve back secret message into the Character Property Algorithm.

The evaluation results and analysis are presented and discussed in the following section.

5. RESULTS AND ANALYSIS

The results and analysis are presented in two sections, namely, the BMA with liveness and the text steganography evaluation results and analysis.

5.1 Results and Analysis for Biometric Multifactor Authentication with Liveness

The results of the BMA evaluation consisted of two parts: BMA user acceptance rate evaluation and liveness detection evaluation. The following subsections describe the results and analysis.

5.1.1 Biometric Multifactor Authentication Evaluation

From the experiment, a total of 250 attempts by 50 participants were enacted, each making three attempts. This consisted of 30 incorrect authentication input attempts in generating imposter attempts. Table 2 shows the results of the experiments in which the system accepted 211 genuine or true individuals, nine were attempts

were rejected, no imposters accepted, and attempts were 30 rejected.

Table 1: FAR & FRR Result

Individuals	Accepted	Rejected
Genuine	**211**	**9**
Imposter	**0**	**30**

In performing the experiment, the heartbeat range was chosen as the threshold to authenticate the liveness of the users. Table 3 below displays the threshold setting for the experiment, which ranged between 50 and 120 bmp. The range between the 50 and 120 bmp is accepted range for the heartbeat rate. A low heartbeat rate indicates the possibility that it belongs to an imposter.

Table 3: Threshold

Threshold	**1**	**2**	**3**	**4**	**5**
Heartbeat (bmp range)	**50-60**	**60-70**	**70-80**	**80-90**	**90-120**

Table 4 below presents the results based on each threshold, and Figure 2 displays the result, as presented in the ROC curve.

Table 4: FFR & FAR with Threshold

Threshold	FFR	FAR
1	**0.087**	**0**
2	**0.04**	**0**
3	**0.02**	**0**
4	**0.02**	**0**
5	**0.02**	**0**

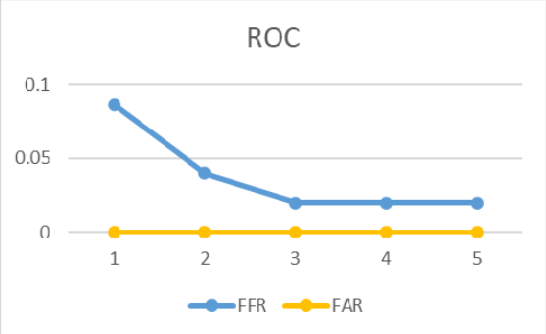

Figure 7: FFR and FAR results

The ROC curve, as shown in Figure 7, presents the FFR and FAR results based on each threshold value, with the average FFR value is 0.04 or 4% and 0% for the FAR. Through the calculation based on the FFR and the FAR equation, the FRR for the biometric device evaluation was 0.04 or 4% in which nine attempts were rejected due to correct inputs. The rejection of correct inputs might have been due to incorrect positioning of the fingerprints and heartbeat sensors, leading to errors during the experiment. Although this BMA results in 0% of the FAR, showing that the system rejected unauthorised users.

5.1.2 Liveness Detection Evaluation

The liveness detection experiment was carried out using 100 samples of fake fingerprints. The results were significant with the false acceptance of fake inputs at 0% (*FerrFake*) for BMA, indicating that no fake fingerprints were incorrectly accepted as live fingerprints. Compared to fingerprint only authentication, it resulted in 13% *FerrFakes*.

Accordingly, this suggests that authentication using only a fingerprint without liveness detection may expose the system to fake fingerprint attacks. The False Rejection of Live (*Ferrlive*) for both BMA was 0% indicating that all attempts were correctly detected as live subjects.

In summary, both experiments showed that biometric fingerprint authentication with liveness detection provided a high-security layer to the ATS system where only live users having verified access use steganography communication. As such, the issue around having a fake identity would be resolved with the added layer of security established by the framework.

5.2 Results and Analysis for Arabic Text Steganography

The evaluation of ATS involved two parts: capacity and transparency. The capacity evaluation consisted of two experiments. The first experiment was a comparison between the CPM and the PSM using the two input approaches; the fixed secret message with a different cover-text,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

and the fixed cover-text with different secret message inputs.

The second experiment compared the previous ATS methods: the Isolated Character method (ICM) and the Modified Run Length Encoding method (RLEM). After that, an experiment was carried out on the ATS evaluation and the transparency evaluation using the Jaro-Wrinkler metric (Jaro Score).

5.2.1 Capacity Evaluation Results and Analysis

5.2.1.1 Experiment 1 Results: Use of Character in Cover-Text Comparison

The first experiment involved two different approaches using a fixed secret message with a different cover-text and a different secret message with a fixed cover-text as input for the hiding process. In this experiment, the use of characters in the cover-text was compared. Figure 8 below displays the results of the first approach for the CPM, while Figure 8 displays the results for the PSM.

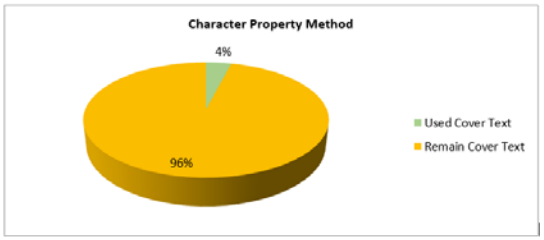

Figure 8: Different secret message with fixed cover-text for the character property method

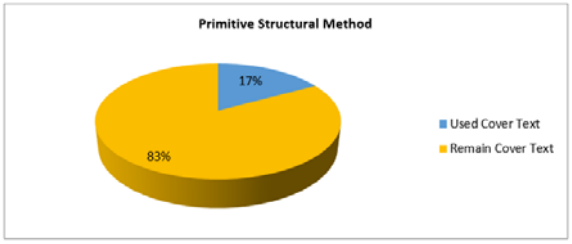

Figure 9: Different secret message with fixed cover-text for the primitive structural method

Figures 8 and 9 depict that improvements in the use of the CPM are less than 13% compared to the PSM. The results, however, still show the superiority of the CPM in hiding more secret information. As such, the experiment proves that the CPM provides more flexible places for hiding the secret bits in the cover-text compared to the PSM. The maximum positioning technique arranges the maximum secret bits for each character based on the Character Property number. Although the PSM creates a random position to hide the secret information, it still needs a large cover-text to hide more of it.

Therefore, even though both methods have the same number of potential places to hide the secret bits, the positioning of the secret bits needs to be considered to optimise the capacity of the embedded secret information.

5.2.1.2 Experiment 2-Results: Comparison with Previous Methods

The second experiment that was undertaken was a comparison with the previous method, the ICM [42] and the RLCM [38]. Figure 10 shows the results in the evaluation to determine capacity.

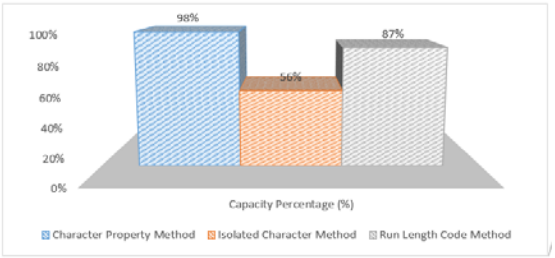

Figure 8: Character property method capacity evaluation comparison with the isolated character method and the run length encoding method

As shown in the figure above, the CPM produces a capacity of 98%, ICM (56%), and RLCM (87%). The CPM result shows a higher capacity of 42% compared to the ICM because the hiding place for the secret information in the latter is limited to the isolated characters. Due to this technique, the other potential Arabic characters are not used to hide the secret bits of information. Compared to the CPM, we utilised all the Arabic characters as places to hide the secret information.

Additionally, the RLCM provides a competitive challenge or edge since the difference is only 11% compared to the CPM. The advantage of the RLCM is in using the ASCII of non-printed characters to hide the secret information, as the CPM hides more than one secret bit per character.

However, even though the RLCM has a huge capacity, it is fragile in the OCR machine where the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

removal of empty spaces in the sentences results in the loss of hidden secret information. Therefore, the CPM results in an average optimisation of 23.5% regarding the embedded secret message compared to the other two methods.

As shown in the figure above, the CPM produces a capacity of 98%, ICM (56%), and RLCM (87%). The CPM result shows a higher capacity of 42% compared to the ICM because the hiding place for the secret information in the latter is limited to the isolated characters. Due to this technique, the other potential Arabic characters are not used to hide the secret bits of information. Compared to the CPM, we utilised all the Arabic characters as places to hide the secret information.

Additionally, the RLCM provides a competitive challenge or edge since the difference is only 11% compared to the CPM. The advantage of the RLCM is in using the ASCII of non-printed characters to hide the secret information, as the CPM hides more than one secret bit per character.

However, even though the RLCM has a huge capacity, it is fragile in the OCR machine where the removal of empty spaces in the sentences results in the loss of hidden secret information. Therefore, the CPM results in an average optimisation of 23.5% regarding the embedded secret message compared to the other two methods.

5.2.2 Perceptual Transparency Evaluation Results and Analysis

The CPM employs the Unicode character approach to hide the secret message, which is not an image. As such, the string similarity distance or the Jaro-Winkler distance is used to measure the transparency of the stego-text. The higher the Jaro-Winkler distance between two strings, the more similar they are. The result is normalised in having a measurement between 0 and 1, zero representing the absence of similarity [43]. Table 5 shows the results from the calculated similarity distance.

Table 5: Jaro Score of the String Pars of the Cover File (C) and Stego File (S)

String Pair	CS_1	CS_2	CS_3	CS_4	CS_5	CS_6	CS_7	CS_8	CS_9	CS_{10}	CS_{11}	CS_{12}
Jaro Score	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

Based on the above table, the CS_i is the ith pair of strings belonging to the cover (C) and the stego file (S) as given in Equation 3 below:

$$CS_i=(C_i,S_i) \quad (3)$$

Where C_i is the ith string of the cover (C), and S_i is the corresponding ith string of the stego file (S).

As can be seen in Table 5, the CPM results in the Jaro-Winkler distance being equal to 1.0. As such, this means that the stego-text and the cover-text are identical and provide an opportunity for the attacker to have doubts that the stego-text contains secret information. Table 6 below shows the comparison between the CPM with the previous works.

Table 6: Jaro Similarity Score Comparison with Previous Methods

Method /Approach	Jaro Similarity Score
Character Property Method	1.0
B+ Tree , DNA Coding and Arabic Diacritics Method (Khadim et al., 2014)	0.8333
Word Mapping Approach (Bhattacharyya et al., 2010)	0.8771

As can be seen in the above table, the CPM results depict the highest score indicating the highest perceptual transparency of the stego-text. The B+ Tree, DNA Coding, and the Arabic Diacritics method scores were 0.8333 while the score of the Word Mapping approach was 0.8771.

Both previous approaches display a vast difference in the score between the cover-text and stego-text because they compute another word in order to embed the secret information, whereas CPM presents the same appearance.

Likewise, the high perceptual transparency results of the CPM give it a significant advantage in allowing the stego-text to be published publicly. Therefore, it functions as high secrecy steganographic communication and reduces the suspicions of potential eavesdroppers.

5.2.3 Robustness Evaluation Results and Analysis

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The robustness evaluation involved three experiments that applied the CPM algorithm using 50 stego-texts with approximately 5 Mb bits per stego-text. The results are presented in the table below.

Table 7: Robustness Evaluation Result

Experiments:	Robustness
Stego text tested with OCR system	Robust. The secret message is retrieved.
Stego text tested using different file formats (.pdf, .doc, .txt)	Robust. The secret message is retrieved
Cover text tested using	Robust. The secret message is retrieved
Retyping Stego text	Robust. The secret message is retrieved

Based on the results as displayed in the above table, the CPM is shown to be robust compared to OCR modification and tampering against altering the file format or tampering via retyping.

6 CONCLUSION

ATS offers significant potential in hiding secret information in Arabic characters. However, most algorithms in this study shared secret keys for embedding and extracting the secret message, thus providing a vast opportunity for intruders to manipulate the communication by creating a fake identity. However, this can be addressed and overcome applying an authentication mechanism for users with proven liveness detection, which gives robustness to the identity of the users and shows the secret information not being modified. Whereas, BMA gives the 0% FAR and only 4% FRR results that show the effectiveness of the authentication system.

Therefore, it can be concluded that this hybrid method of ATS can reduce the level of vulnerability while enhancing the robustness of the authentication system. Further, a high level of user authentication can also be achieved with 0% for both ferrfake and ferrlive of the liveness detection for the fingerprint owner. By taking advantage of the independent credentials of people, namely their fingerprint and heartbeat as a defence mechanism against data spoofing, this method will provide better text steganography in data sharing.

Similarly, the CPM presents a huge potential for hiding secret information where a maximum of 12 bits of secret information per character can be hidden in each Arabic character. Our experiment compared two distinct techniques for hiding secret information through the random positioning of the basic shape groups and the maximum number of potential hiding places. Both techniques have advantages and disadvantages. Since we were more concerned with optimising the embedded secret information capacity, our CPM with maximum positioning resulted in 23.5% more optimising compared with the previous method.

In conclusion, the CPM with MBA covers both passive and active attacks for the steganography system and offers a new approach ATS. The future work of this area includes such as combining a cryptography for the shared key with this method will effectively enhance the security of the system.

REFERENCES

[1] X. Sun *et al.*, "Steganography in Chinese text," in *2010 Int. Conf. Comp. App. Syst. Modelling (ICCASM 2010)*, 2010, vol. 8, pp. V8-651-V8-654.

[2] A. E. Ali, "A new text steganography method by using non-printing Unicode characters," *Eng. Technol. J.*, vol. 28, no. 1, pp. 72–83, 2010.

[3] N. Samphaiboon and M. N. Dailey, "Steganography in Thai text," *5th Int. Conf. Electr. Eng. Comput. Telecommun. Inf. Technol. ECTI-CON 2008*, 2008, vol. 1, pp. 133–136.

[4] K. Alla and R. S. R. Prasad, "A new approach to Hindi text steganography using matraye, core classification and HHK scheme," in *2010 7th Int. Conf. Inf. Technol.: New Gener.*, 2010, pp. 1223–1224.

[5] N. Samphaiboon and M. N. Dailey, "Steganography in Thai text," In *5th Int. Conf. Elect. Eng./Electron., Comp., Telecomm. Informat. Technol.*, 2008, vol. 1, pp. 133-136.

[6] E. J. Kusuma *et al.*, "A combination of inverted LSB, RSA, and Arnold transformation to get secure and imperceptible image steganography," *J. ICT Res. Appl.*, vol. 12, no. 2, pp. 103–122, 2018.

[7] O. O. Okediran *et al.*, "Secure electronic voting using a hybrid cryptosystem and

steganography," *J. Adv. Math. Comput. Sci.*, vol. 34, no. 1, pp. 1–26, 2019.

[8] I. S. Bajwa and R. Riasat, "A new perfect hashing based approach for secure steganography," in *2011 6th Int. Conf. Digit. Inf. Manage., ICDIM 2011*, 2011.

[9] P. Vidhya, "Text steganography using public key cryptosystem in CSS," *Int. J. Comput. Appl. Eng. Sci.,* vol. 22, no. 3, 2012.

[10] K. M. Harshitha and P. A. Vijaya, "Secure data hiding algorithm using encrypted secret message," *Int. J. Sci. Res. Publ.*, vol. 2, no. 6, pp. 1–4, 2012.

[11] Z. Wang *et al.*, "Information hiding based on DNA steganography," in *Proc. IEEE Int. Conf. Softw. Eng. Service Sci., ICSESS*, 2013.

[12] R. Indrayani *et al.*, "Increasing the security of MP3 steganography using AES encryption and MD5 hash function," in *Proc. - 2016 2nd Int. Conf. Sci. Technol.-Comput., ICST 2016*, 2017.

[13] R. Kaur *et al.*, "A hybrid approach for video steganography using edge detection and identical match techniques," *Proc. 2016 IEEE Int. Conf. Wirel. Commun. Signal Process. Netw., WiSPNET 2016*, 2016, pp. 867–871.

[14] A. Solichin and E. W. Ramadhan, "Enhancing data security using DES-based cryptography and DCT-based steganography," *Proc. - 2017 3rd Int. Conf. Sci. Inf. Technol. Theory Appl. IT Educ. Ind. Soc. Big Data Era, ICSITech 2017*, 2017, pp. 618–621.

[15] N. A. Roslan *et al.*, "Primitive structural method for high capacity text steganography," *J. Theor. Appl. Inf. Technol.*, vol. 67, no. 2, 2014.

[16] K. Ntalianis and N. Tsapatsoulis, "Remote authentication via biometrics: A robust video-object steganographic mechanism over wireless networks," *Ieee Trans. Emerg. Top. Comput.*, vol. 4, no. 1, pp. 156–174, 2015.

[17] I. Banerjee *et al.*, "Biometric steganography using face geometry," *IEEE Reg. 10 Annu. Int. Conf. Proc. /TENCON*, 2015.

[18] S. Das *et al.*, "Lip biometric template security framework using spatial steganography," *Pattern Recognit. Lett.*, vol. 126, 102-110, 2019.

[19] A. Cheddad *et al.*, "A hash-based image encryption algorithm," *Opt. Commun.*, vol. 283, no. 6, pp. 879–893, 2010.

[20] B. Sivaranjani and N. Radha, "Securing patient's confidential information using ECG steganography," in *2nd Int. Conf. Commun. Electron. Syst.* 2017, pp. 540–544.

[21] Z. N. J. Al-Kateeb and M. R. J. M. Al-Bazaz, "Steganography in coloured images based on biometrics," *Tikrit J. Pure Sci.*, vol. 24, no.3, pp. 111-117, 2019.

[22] R. A. Alotaibi and L. A. Elrefaei, "Improved capacity Arabic text watermarking methods based on open word space," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 30, no. 2, pp. 236–248, 2018.

[23] R. A. Alotaibi and L. A. Elrefaei, "Utilising word space with pointed and un-pointed letters for Arabic text watermarking," *Proc. - 2016 UKSim-AMSS 18th Int. Conf. Comput. Model. Simulation*, 2016, pp. 111–116.

[24] H. Alshahrani and G. Weir, "Hybrid Arabic text steganography," *Int. J. Comput. Inf. Technol.*, vol. 6, no. 6, pp. 329–338, 2017.

[25] A. Gutub and K. Alaseri, "Hiding shares of counting-based secret sharing via Arabic text steganography for personal usage," *Arab. J. Sci. Eng.*, 2019.

[26] S. M. Al-Oun and J. Q. Odeh Alnihoud, "An efficient approach to hide compressed voice data in Arabic text using Kashida and 'La'," *J. Comput. Sci.*, vol. 13, no. 3, pp. 48–54, 2017.

[27] A. A., F. Ridzuan, and S. Ali, "Text steganography using extensions Kashida based on the moon and sun letters concept," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 8, pp. 286–290, 2017.

[28] D. S. Malalla and F. R. Shareef, "A new modified fatha method for Arabic text steganography hybrid with aes encryption," *IOSR J. Comput. Eng.*, vol. 18, no. 05, pp. 37–45, 2016.

[29] M. Karnan *et al.*, "Biometric personal authentication using keystroke dynamics: A review," *Appl. Soft Comput.,* vol. 11, no. 2, pp. 1565-1573, 2011.

[30] J. Wayman *et al.*, "An introduction to biometric authentication systems," in *Biometric Systems,* London: Springer, 2005, pp. 1-20.

[31] C. Sousedik and C. Busch, "Presentation attack detection methods for fingerprint

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

recognition systems: A survey," *IET Biometrics*, vol. 3, no. 4, pp. 219–233, 2014.

[32] J. B. P. Lee, and D. Stewart, "Deloitte TMT prediction 2014 technical report," London, United Kingdom, 2014.

[33] F. Wang and G. Gao, "Embedded fingerprint identification system based on DSP chip," *Adv. Comput. Sci. Intell. Syst. Environ.*, pp. 595–599, 2011.

[34] N. I. Zainal *et al.*, "Design and development of portable classroom attendance system based on Arduino and fingerprint Biometric," in *Inform. Commun. Technol. Muslim World (ICT4M), 2014 5th Int. Conf.*, 2014, pp. 1–4.

[35] J. Hart, "Normal resting pulse rate ranges," *J. Nurs. Educ. Pract.*, vol. 5, no. 8, pp. 95, 2015.

[36] S. Trewin *et al.*, "Biometric authentication on a mobile device: A study of user effort, error and task disruption," in *Proc. 28th Annu. Comput. Secur. Appl. Conf.*, 2012, pp. 159–168.

[37] A. F. Sequeira and J. S. Cardoso, "Fingerprint liveness detection in the presence of capable intruders," *Sensors (Switzerland)*, vol. 15, no. 6, pp. 14615–14638, 2015.

[38] S. M. Kadhem, "Text steganography method based on modified run length encoding. *Iraqi J. Sci.*, vol. 57, no. 3, pp. 2338-2347, 2016.

[39] A. Gutub *et al.*, "e-Text watermarking: Utilising 'Kashida' extensions in Arabic language electronic writing," *J. Emerg. Technol. Web Intell.*, vol. 2, no. 1, pp. 48–55, 2010.

[40] A. Desoky, "Comprehensive linguistic steganography survey," *Int. J. Inf. Comput. Secur.*, vol. 4, no. 2, pp. 164-197, 2010.

[41] M. Agarwal, "Text steganographic approaches: A comparison," *J. Int. Secur. Netw. Appl. Its*, vol. 5, no. 1, pp. 91-106, 2013.

[42] A. T. Abbasi *et al.*, "Urdu text steganography: Utilising isolated letters," in *13th Australian Inform. Secur. Manag. Conf.*, Edith Cowan University Joondalup Campus, Perth, Western Australia, 2015, pp. 37-46.

[43] S. Bhattacharyya *et al.*, "A novel approach of secure text based steganography model using word mapping," *Int. J. Comput. Inf. Eng.*, pp. 96–103, 2010.