Journal of Information Hiding and Multimedia Signal Processing ©2023 ISSN 2073-4212
Ubiquitous International Volume **14**, Number **3**, September **2023**

Arabic Text Steganography based on Arabic Astrology

Assist Prof. Dr. Farah R. Shareef Taka

Ministry of Education
Baghdad, Iraq
sabahaliraq2014@gmail.com

Received November 2022; revised July 2023

Abstract. *Steganography is a significant strategy for data security that can protect data by concealing a message within a cover message. Most steganography research uses cover media, such as text, videos, pictures, and sounds. Even so, text steganography is generally not prioritized, because it can hide little data compared to other types of steganography. This paper proposed a new strategy of Arabic text steganography, which used the nature of Arabic letters for Astrology science for the embedding process, to increase the capacity of embedding, which is based on the nature of letters for Astrology science. Two secret bits have been embedded into Arabic text by using (kashida) and Unicode letters, which are, zero width no joiner (ZWNJ), Hair, Right-To-Left (RTL), and zero width no-break space (ZWS). These letters are classified into four types: fiery, watery, earthy, and airy. The names and the letters they contain are clear evidence of their individuality. The results show that; this method has reached two strengths. One is the great ability to conceal large messages in Arabic letters compared to traditional stego techniques based on "kashida" as it can hide two secret bits rather than a bit. Secondly, it is difficult to notice by the watcher's "Visibility". As well as the difficulty of detecting the decryption process since it uses eight cases to hide secret bits through four scenarios for each case, in addition, it can work with any language. This method represents a new high-capacity stego method for hiding a large secret message in an Arabic text cover.*
Keywords: Steganography, Text Steganography, Astrology, Stego, Embedding.

1. Introduction. In general, the web is the most well-recognized and commonly used medium for interpersonal communication and the transmission of messages (cryptic data) to the perfect recipient. While communication passes across the internet, the web does not provide end-to-end security [1]. This is a serious problem that can easily lead to trade-offs for unknown data. Steganography is, therefore, such an important topic for experts, advocacy organizations, and government organizations.

Steganography and Cryptography [2] are information security techniques, which have been used for data protection for many years. Because computer capabilities have been quickly increasing in recent years, cryptographic technology may be vulnerable. Furthermore, the availability of cipher texts mitigates this flaw because eavesdroppers can apply cryptanalysis tactics on the system to crack it. This shortcoming, however, can be significantly minimized by employing a hiding strategy, which is a type of clandestine communication. Steganography [3] is the method utilized to implement this concept.

Data hiding is a standard term covering several sub-disciplines. The most significant sub-discipline is the steganography technique. In steganography, the suitable cover is the traditional-looking cover that can be (text, audio, images, video, or various other digital consultant code) that can hold the hidden data. The secret message (which may be images, plaintext, ciphertext, or any data that need to disappear by being hidden within other messages. In steganography, both the embedded message and cover carrier make the stego-carrier. The information hiding may need a stego key, which is another secret information, like a password, wanted for embedding the information [4].

Text steganography is generally grouped into three types [5]: Randomly and statistically generated based on format and language methods. Techniques used to integrate data within a text, and their characteristics need to be improved. These types may be one of the different strategies including the size of the font, insert of not-viewed spaces or characters, prepared misspellings allocated throughout the text, and such like. There exist three most important factors when making use of steganography, which are capacity (the highest amount of concealed data, Security (which is the protection and privacy snooper), and the amount of change on the coverage that can be held before removing the private data. (Robustness) [6].
Historically, hiding secret messages has been used for a long time to cloak methods in settings that can be in any other event properly public, whether it is poetry, music, botanical drawings, and even star charts. Astrology, likewise, is also used as a cover for secret communication. Essentially, as lately as 1996, a hidden secret message was found that it has been hidden in the astrological tables within a well-known occult manuscript belonging to the 1500s. The one who developed this scheme is the monk "Johannes Trithemius", whose astrological prediction relates to the political prospects of the Jews. Incredibly, [7].
In this work, we intend to utilize astrology to propose Arabic text steganographic method that can in addition to being a new hiding method, it can get better capacity by raising the embedding data within the text of the cover.

2. Literature Survey. The Arabic language is consisting of twenty-eight characters, which can be written in any cursive style matching to Farsi and Urdu. Based on it is the location of the word, the letter of Arabic adjusts shape. It could appear firstly, mid, or latest position or could even be separated. Every word generally includes more than two letters joined with each other. Several Arabic letters involve 1, 2, or 3 dots located either on top of the character or under the character. In comparison to the English language, which has no multi-point characters, the Arabic characters have FIFTEEN pointed letters, five of which are multi-points [8].
Arabic words, in addition, have diacritics known as "Harakat" which are put to the frame of the vowel sound. The 8 Arabic content diacritics are Damah (ُ), Tanwin Damah (ٌ), Fathah (َ), Tanwin Fathah (ً), Kasrah (ِ), Tanwin Kasrah (ٍ), and Shaddah (ّ). These types of diacritics are necessary for realizing the Islam Holy Quran, historical texts, religious scripts, and Arabic studying books [9].
Even so, almost all other Arabic text doesn't include diacritics. The text of Arabic likewise involves an additional character named kashida, which is utilized as a justification for the words, and also white spaces, that justify the text of Arabic. Kashida is placed right after a letter depending on its position in a word [10].
It may make use of Arabic steganography text for handling secret bits in different algorithms. Depending on the Arabic characteristic described previously. However, this work is depended on the nature of the letter for Astrology science, where (kashida) can be served for this goal. Hence, several Arabic Text Steganography algorithms that utilized (kashida) are explained, as follows:

2.1 Kashida variation [11 This method was created to increase robustness. A Text cover message segments into blocks, and then hides the bits in each block depending on the following scenario: the first is to insert Kashida after Arabic dotted letters to represent "1" in another case, it should be "0". The second case is to insert Kashida after not dotted Arabic letters to represent "1" in any other case it will be "0". The third case is to insert Kashida after letters to represent "1"; in any other case, it will be "0". The fourth case is to insert Kashida after letters to represent "1"; in any other case, it should pass "0", and inserting Kashida after letters to represent "1"; in any other case, it will pass "1".

2.2 Inserting Kashida using specific characters [12] In this method, Kashida is a place to be a bit 1 and it is deleted for a bit 0 within a watermarking key, which is preset. The approach for inserting Kashida depends upon inserting Kashida before a certain set of the

characters (ئ،آ،إ،أ،ؤ،و،ا،ز،ر،ذ،د) until the key reaches the end. This method is a high level of imperceptibility encoding. Moreover, this stego method is of small capacity, so it can be useful and better to use with applications that required low data sizes such as in document validity and copyright protection.

2.3 Kashida for hiding two bits [13] In this method, two bits can be hidden by using one Kashida rather than slandered approaches that hid only one bit in each Kashida. This can be achieved depending upon the qualities of letters, such as whether they are (unpointed or pointed letters). Hence, the cover message (the text message) should be divided into two similar chunks by using this manner. Each block is dealt with in a specific way. However, in some situations, the sequences of secret bits and the best-adapted appearances from Kashida can have an impact on the capacity ratio in this approach, which can be deemed negative and guide to unfavorable results.

2.4 Utilization of the Sun and Moon letters [14] The message in this method that contains secret bits is hidden in sun and moon letters (letters classification in the Arabic language). To achieve this goal, Kashida be inserted by using four scenarios: one is to add Kashida exactly after the sun letter to represent "00" secrets. In the second scenario, Kashida; is added next a letter from the sun and it stands for "11" secret bits. In the third scenario, Kashida; is added after a letter from the moon and it represents "01" secret bits. The final scenario is to insert 2 Kashida involves to conceal the ten secret bits.

2.5 Kashida with Small Space Characters [15] In this method Kashida and small space characteristics in addition to Arabic language attributes, in this method each Kashida can hide one bit while there is enough space to hide three bits, so it can increase the secret messages length.

2.6 Using ZWC and ZWJ [16] This method can be increasing the capacity and security of secret data by combining steganography and encryption. To conceal data in the carrier of Arabic text, the recommended algorithm employs linguistic steganography. To disguise the secret information, the current approach employs Unicode characters for example Zero-Width-Character (termed ZWC) and Zero-Width-Joiner (ZWJ). Before embedding confidential information, bit inversion utilize to encrypt secret data, resulting in a high-security technique. The suggested algorithm's findings were successful in achieving a high capacity of the cover medium, security, and resilience.

2.7 Kashida with Darkness and Noorani letters [17] In this method, Kashida is used with Unicode space characters, in which the embedding process depends on the Arabic letter properties called "Darkness and Noorani" letters. The Noorani characters are located in (Al-Fatiha) surah in addition to the starting of every Al-Quran surah. while Darkness letters aren't seen in (Al- Fatiha) surah and also it couldn't be found at the beginning of all Al-Quran surahs. There are five scenarios in every case for incorporating secret elements into the cover of the Arabic text. Firstly, case, 00 secret bits, and 11 secret bits are concealed by inserting Kashida1 and Kashida2 just after the Noorani letter. Kashida1 and Kashida2, are inserted directly after the Darkness letter, Secondly, case, to conceal the 01 and 10 secret bits, respectively. In the third case, Unicode space characters (Narrow space and Hair space) are placed exactly next to the Isolated_ Noorani char to conceal the secret bits 00 and 11. Unicode space characters (Hair space and Narrow space) are placed exactly after the Isolated_ Darkness letter to hide 01 secret bits and 10 secret bits. In the last case, the space existence is replaced by (ZWNJ/00, PRE space/01, Thin/10, PRE+Thin/11).

3. Proposed Method The Arabic language involves twenty-eight alphabets and every alphabet owns its unique number, which is its numerical value. By using these numerical values, several computations can be made, that is the most appropriate of which any Astro - Science can provide and can be utilized in several fields using many applications. These twenty-eight alphabets are also divided into seven pairs and every pair include four alphabets. They are also divided into similar Abjad series, then they will give values for each elemental. Abjad's alphabet has been divided into Four Elements: Earth, Air, Fire, and Water. In addition, every element has seven alphabets. These were then likewise based on the Zodiacal and planets' Signs [18].

3.1 Compression to a secret message While the secret message is important, the algorithm of (gzip) provides a better compression ratio for a secret message that fits into the text of the Arabic cover. (Fig. 1)

Figure 1: The Model for the compressing process of secret messages

3.2 The Process of Embedding The hiding process is depending on the nature of Arabic letters for Astrology science; there are four types of Arabic letters shown below in Table 1.
The **fiery letters**: أ ـ ذ – ش – ط – ف – م ـ ـه, and its effect on the person whose name contains the largest number of letters, including that he is classified by a group of characteristics such as the tendency to collect money, control, intensity, pride, disclosing the secret, the tyranny of opinion, and activity.
As for the **Airy letters**:ث ـ ج – ز – س – ظ – ق – ك , in addition, it affects the individual whose name bears the largest number of them, he may be impulsive, with instability, nervousness, love of life, and reaching the goal.
Moreover, the **Earthy letters**: ب ـ ت – ص – ض – ن – و – ي, its effect on the one who carries the largest number of attractiveness, keeping a secret, dullness, love of logic, getting things done before it is too late, melancholy or sensuality
Watery letters:ح ـ خ – د – ر – ع – غ – ل, the owners of a wide number of them, we find them loving politics and peace with sobriety, taking things relentlessly, and a tendency to fame and fickleness. See Table 1. Some Arabic letters do not allow kashida after the letter, where

Table 1: Fiery, Earthy, Airy, and Watery letters

Fiery letters	أ	ذ	ش	ط	ف	م	ـه
Air letters	ث	ج	ز	س	ظ	ق	ك
Earth letters	ب	ت	ص	ض	ن	و	ي
Water letters	ح	خ	د	ر	ع	غ	ل

the shadowed lines in Table 1. are referred to as (Isolated letters), and there are four types: Isolated-Firey letters have 10 items (أ,إ,آ,ا,ى,ئ,ء,ذ,ة,ه), Isolated-Airy letters contain one item (ز), Isolated-Earthy letters contain two items (ؤ,و), and Isolated-Watery letters contain two items (ر,د).
The proposed approach presents 4 scenarios of Isolated letters in order to hide the secret bits

inside the Arabic text cover. Each scenario has 4 cases. Here, the process of concealing is used (Unicode letters) that are not seen by the reader.

A. Isolated-Firey scenario. The first case; is depending upon adding a zero width no joiner (ZWNJ) letter, next to the Isolated- Firey char, to conceal the two secret elements (00), the second case is relying on replacing with (Hair) letter in the case where the secret elements are (01). In 3rd case is based on the replacement by the right to the left letter(R-T-L), and the 4th case is based on replacing with zero width no-break space (zws), see Table 2.

Table 2: Process of embedding for Isolated-Firey Letter.

Secret elements (bits)	Work
00	ZWNJ(\u200c)
01	Hair(\u200A)
10	R-T-L(\u200F)
11	ZWS(\uFEFF)

B. Isolated –Airy scenario. This scenario is based on Table 3 and the method can be described as follows:

- The first case is depending on replacing hidden bits with (Hair) letters) 00).
- The second case is depending on substituting secret bits with (Arabic letter mark) letters (01).
- The third case is depending on replacing secret bits with (R-T-L) letters (10).
- The fourth case is depending on replacing secret bits with the letter (ZWNJ) (11).

Table 3: Process of embedding for Isolated-Airy Letter.

Secret elements (bits)	Work
00	Hair
01	Arabic letter mark(\u061c)
10	R-T-L
11	ZWNJ

C. Isolated- Earthy scenario. This scenario is based on Table 4 and the method can be described as follows:

- In the first case, secret bits are changed with the letter (zws) (00).
- The second case is depending on replacing secret bits with the letter (ZWNJ) (01).
- The third case is depending on replacing hidden bits with (Hair) letters (10).
- The fourth case is depending on replacing secret bits with (R-T-L) letters (11).

Table 4: Process of embedding for Isolated-Earthy letter.

Secret elements (bits)	Work
00	ZWS
01	ZWNJ
10	Hair
11	R-T-L

D. Isolated –watery scenario. This scenario is based on Table 5 and the method can be described as follows:

- In case one. it depends on substituting with (R-T-L) letter when secret bits (00).

• In case two it depends on substituting with the (zws) letter when secret bits (01).
• In case three it depends on substituting with (Arabic letter mark) letter when secret bits (10).
• In case four it depends on substituting with (Hair) letter when secret bits (11).

Table 5: Process of embedding for Isolated-Earthy letter.

Secret elements (bits)	Work
00	R-T-L
01	ZWS
10	Arabic letter mark
11	Hair

In addition, the non-shaded rows in Table 1, represented the letters that can accept kashida after these letters. In this case method, there are "4" scenarios for embedding secret bits within Arabic text coverage (text message) with "4" cases per scenario.
1) Fiery_letters Scenario. This scenario is based on Table 6 and the method can be described as follows:
• In case one, it depends on adding two letters (kashida+zws) after the fiery_letter, in case the secret elements are (00).
• In case two, it depends on what is implemented by adding (kashida+ word joiner(wj)), in case the secret elements are (01),
• In case three, it depends on substituting two letters (zero width joiner (ZWJ) +wj), in case the secret elements are (10).
• In case four, it depends on substituting with (ZWJ), in case secret elements are (11).

Table 6: Process of embedding for fiery_letters.

Secret elements (bits)	Work
00	Kashida(\u0640) + zws(\uFEFF)
01	Kashida + wj(\u2060)
10	ZWJ(8205) + wj
11	ZWJ

2) Airy_letters scenario. This scenario is based on Table 7 and the method can be described as follows:
• In case one, it depends upon the addition of (kashida) after the Airy_letter, for the case (00) of secret bits.
• In case two it depends on adding two letters (kashida+ zws), in case the secret bits are (01)
• In case three, it depends on substituting two letters (zwj+wj) when the secret bits are (10).
• In case four, it depends on substituting with (zwj+zws), in case the secret bits are (11).
3) Earthy_letter scenario. This scenario is based on Table 8 and the method can be described as follows:
• In case one, it depends on adding two letters (zwj+zws) after the Earthy_letter, in case the secret bits are (00)
• In case two, it depends upon adding (kashida), in case the secret elements are (01)
• In case three, it depends on substituting with (ZWJ), in case the secret elements are (10)
• In case four, it depends on substituting two letters (kashida+wj), in case the secret elements are (11).

Table 7: Process of embedding for Airy_letters.

Secret elements (bits)	Work
00	Kashida
01	Kashida + zws
10	ZWJ + wj
11	Zwj+zws

Table 8: Process of embedding for Earthy_letters

Secret elements (bits)	Work
00	Zwj+zws
01	Kashida
10	ZWJ
11	Kashida + wj

4) Watery_letters scenario. This scenario is based on Table 9 and the method can be described as follows:

• In case one, it depends on adding two letters (zwj+wj) after the watery_letter, in case the secret elements are (00).

• In case two, it depends on adding (ZWJ), in case the secret elements are (01)

• In case three, it depends on replacing with two letters (kashida+wj), in case the secret elements are (10),

• In case four, it depends on substituting with (kashida), in case the secret elements are (11).

Table 9: The embedding process for watery_letters

Secret elements (bits)	Work
00	Zwj+wj
01	ZWJ
10	Kashida+wj
11	Kashida

3.3 Embedding Algorithm.

Input: The cover of Arabic Text (cv), secret message(s).
Output: The Arabic-Stego (A_S).
1. n=The counter of cover.
2. Turn the secret(s) into bits (sb), k=bit counter.
3. Verify if:
   (1) Cv(n)= (The Arabic Diacritics Category) then, n++.
   (2) Cv(n)= (The Delimiters Category) then, n++.
   (3) Cv(n)= (NL-new line) OR (CR-courage return) then, n++.
4. If cv(n)= (Isolated_Firey Category) then:
   (1) If (sb(k)=0) and (sb(k+1)=0) thereafter set (ZWNJ after character).
   (2) If (sb(k)=0) and (sb(k+1)=1) thereafter set (Hair).
   (3) If (sb(k)=1) and (sb(k+1)=0) thereafter set (R-T-L).
   (4) If (sb(k)=1) and (sb(k+1)=1) thereafter set (zws).
5- If cv(n)= (Isolated_Airy Category) then:

(1) If (sb(k)=0) and (sb(k+1)=0) thereafter set (Hair).
(2) If (sb(k)=0) and (sb(k+1)=1) thereafter set (Arabic letter mark).
(3) If (sb(k)=1) and (sb(k+1)=0) thereafter set (R-T-L).
(4) If (sb(k)=1) and (sb(k+1)=1) thereafter set (ZWNJ).

6- If cv(n)= (Isolated_Earthy Category) then:
(1) If (sb(k)=0) and (sb(k+1)=0) thereafter set (zws).
(2) If (sb(k)=0) and (sb(k+1)=1) thereafter set (ZWNJ).
(3) If (sb(k)=1) and (sb(k+1)=0) thereafter set (Hair).
(4) If (sb(k)=1) and (sb(k+1)=1) thereafter set (R-T-L).

7- If cv(n)= (Isolated_Watery Category) then:
(1) If (sb(k)=0) and (sb(k+1)=0) thereafter set (R-T-L).
(2) If (sb(k)=0) and (sb(k+1)=1) thereafter set (zws).
(3) If (sb(k)=1) and (sb(k+1)=0) thereafter set (Arabic letter mark).
(4) If (sb(k)=1) and (sb(k+1)=1) thereafter set (Hair).

8- If cv(n)= (Firey_letters Category) then:
(1) If (sb(k)=0) and (sb(k+1)=0) thereafter set (Kashida+zws).
(2) If (sb(k)=0) and (sb(k+1)=1) thereafter set (Kashida+wj).
(3) If (sb(k)=1) and (sb(k+1)=0) thereafter set (zwj+wj).
(4) If (sb(k)=1) and (sb(k+1)=1) thereafter set (zwj).

9- If cv(n)= (Airy_letters Category) then:
(1) If (sb(k)=0) and (sb(k+1)=0) thereafter set (Kashida).
(2) If (sb(k)=0) and (sb(k+1)=1) thereafter set (Kashida+zws).
(3) If (sb(k)=1) and (sb(k+1)=0) thereafter set (zwj+wj).
(4) If (sb(k)=1) and (sb(k+1)=1) thereafter set (zwj+zws).

10- If cv(n)= (Earth_letters Category) then:
(1) If (sb(k)=0) and (sb(k+1)=0) thereafter set (zwj+zws).
(2) If (sb(k)=0) and (sb(k+1)=1) thereafter set (Kashida).
(3) If (sb(k)=1) and (sb(k+1)=0) thereafter set (zwj).
(4) If (sb(k)=1) and (sb(k+1)=1) thereafter set (Kashida+wj).

11- If cv(n)= (Watery_letters Category) then:
(1) If (sb(k)=0) and (sb(k+1)=0) thereafter set (zwj+wj).
(2) If (sb(k)=0) and (sb(k+1)=1) thereafter set (wj).
(3) If (sb(k)=1) and (sb(k+1)=0) thereafter set (Kashida+wj).
(4) If (sb(k)=1) and (sb(k+1)=1) thereafter set (Kashida).

12- End.

3.4 The Algorithm of Decoding.

Input: The Arabic-Stego (A-S).
Output: The secret message.
1. Convert (A-S) to a character array (St).
2. n= The counter of cover.
3. Verify every character for (A-S) with Isolated-Firey $(\mathrm{ISO}_F)Category$:
a. If (ST (n) = $(\mathrm{ISO}_F)char.)and(ST(n+1) = ZWNJ)thereaftergrab(00)$.
b. If (ST (n)= $(\mathrm{ISO}_F)char.)and(ST(n+1) = Hair)thereaftergrab(01)$.
c. If (ST (n) = $(\mathrm{ISO}_F)char.)and(ST(n+1) = R-T-L)thereaftergrab(10)$.
d. If (ST (n)= $(\mathrm{ISO}_F)char.)and(ST(n+1) = zws)thereaftergrab(11)$.

4.$Verifyevery character for(A-S)with Isolated-Airy(Iso-A)Category$:
a. If (ST (n) = $(\mathrm{ISO}_A)char.)and(ST(n+1) = Hair)thereaftergrab(00)$.
b. If (ST (n)= $(\mathrm{ISO}_A)char.)and(ST(n+1) = Arabicletter.mark)thereaftergrab(01)$.
c. If (ST (n) = $(\mathrm{ISO}_A)char.)and(ST(n+1) = R-T-L)thereaftergrab(10)$.
d. If (ST (n)= $(\mathrm{ISO}_A)char.)and(ST(n+1) = ZWNJ)thereaftergrab(11)$.

5.$Verifyevery character for(A-S)with Isolated-Earthy(ISO_E)Category$:
a. If (ST (n) = $(\mathrm{ISO}_E)char.)and(ST(n+1) = zws)thereaftergrab(00)$.

b. If (ST (n)= (ISO_E)$char.)and(ST(n+1) = ZWNJ)thereaftergrab(01)$.
c. If (ST (n) = (ISO_E)$char.)and(ST(n+1) = Hair)thereaftergrab(10)$.
d. If (ST (n)= (ISO_E)$char.)and(ST(n+1) = R-T-L)thereaftergrab(11)$.

6.$Verifyevery character for(A-S)with Isolated-Watery(Iso-W)Category$:
a. If (ST (n) = (ISO_W)$char.)and(ST(n+1) = R-T-L)thereaftergrab(00)$.
b. If (ST (n)= (ISO_W)$char.)and(ST(n+1) = zws)thereaftergrab(01)$.
c. If (ST (n) = (ISO_W)$char.)and(ST(n+1) = Arabicletter.mark)thereaftergrab(10)$.
d. If (ST (n)= (ISO_W)$char.)and(ST(n+1) = Hair)thereaftergrab(11)$.

7.$Verifyevery character for(A-S)with Fireyletters(F)Category$:
a. If (ST (n) = (F) char.) and (ST(n+1) = (Kashida+zws)) thereafter grab (00).
b. If (ST (n)= (F)char.) and (ST(n+1) = (Kashida+wj)) thereafter grab (01).
c. If (ST (n) = (F) char.) and (ST(n+1) =(zwj+wj)) thereafter grab (10).
d. If (ST (n)= (F) char.) and (ST(n+1) = zwj) thereafter grab (11).

8. Verify every character for (A-S) with Air letters (A) Category:
a. If (ST (n) = (A) char.) and (ST(n+1) = (Kashida)) thereafter grab (00).
b. If (ST (n)= (A)char.) and (ST(n+1) = (Kashida+zws)) thereafter grab (01).
c. If (ST (n) = (A) char.) and (ST(n+1) =(zwj+wj)) thereafter grab (10).
d. If (ST (n)= (A) char.) and (ST(n+1) = (zwj+zws)) thereafter grab (11).

9. Verify every character for (A-S) with Earthy letters (E) Category:
a. If (ST (n) = (E) char.) and (ST(n+1) = (zwj+zws)) thereafter grab (00).
b. If (ST (n)= (E)char.) and (ST(n+1) = (Kashida)) thereafter grab (01).
c. If (ST (n) = (E) char.) and (ST(n+1) =(zwj)) thereafter grab (10).
d. If (ST (n)= (E) char.) and (ST(n+1) = (Kashida+wj)) thereafter grab (11).

10. Verify every character for (A-S) with Watery letters (W) Category:
a. If (ST (n) = (W) char.) and (ST(n+1) = (zwj+wj)) thereafter grab (00).
b. If (ST (n)= (W)char.) and (ST(n+1) = (zwj)) thereafter grab (01).
c. If (ST (n) = (W) char.) and (ST(n+1) =(Kashida+wj)) thereafter grab (10).
d. If (ST (n)= (W) char.) and (ST(n+1) = Kashida) thereafter grab (11).

11. End.

3.5 Evaluation. In this research, the primary purpose is about getting better for the capacity percent and concealment capacity.

We need to know the following terms and equations, which have been used in this method [19]:

➢ Real use of char is how many initial characters in the media coverage can hide the secret. [20]

➢ Percentage Capacity (PC): it is intended to provide a percentage of media coverage.

P.C = (real use of cover/length of cover) * 100

➢ Hiding Capacity (HC): For percent of actual coverage usage (bytes).

➢ H.C = secret (bits)/Real use (bytes)

➢ Ratio (secret/cover [21]. It helps to understand the ratio between the total number of secret bits concealed in a certain number of characters in the media coverage that are enough to cover up such secrets.

Ratio (secret/cover) = real use of cover/secret bits.

4. Result and Discussion There are three languages, (Persian, Arabic, and English) with several sizes are used together with two covers of Arabic text. Table (10) details the specification of secret and cover messages. The test has done using a laptop of specification) CPU: 2.8GH Core i7, 8.0 GB SD RAM, and Windows 10 as the operating system. The program has been designed and implemented using C# language.

Secret message (Ms), the cover message (Cr), the English language message (Eng.), the Arabic language message (Ara) and the Persian language message (Pers.)

Three secret messages (Ms1, Ms2, Ms3) have been tested with cover Cr1 and three other messages (Ms4, Ms5, Ms6) have been tested with cover Cr2, to check on the performance of

Table 10: Details of secret messages and covers.

Covers and Secret message	The Language	The number of letters(byte)
Ms1	Eng.	28
Ms2	Eng.	346
Ms3	Pers.	1678
Ms4	Ara.	2391
Ms5	Eng.	3212
Ms6	Ara.	3828
Cr1	Ara.	25143
Cr2		257895

the proposed model for the terminology of the real use of characters, hiding capacity and cover percentage. Then these results; are compared with our previous work in [17], as shown in Figure 2 and Table 11.

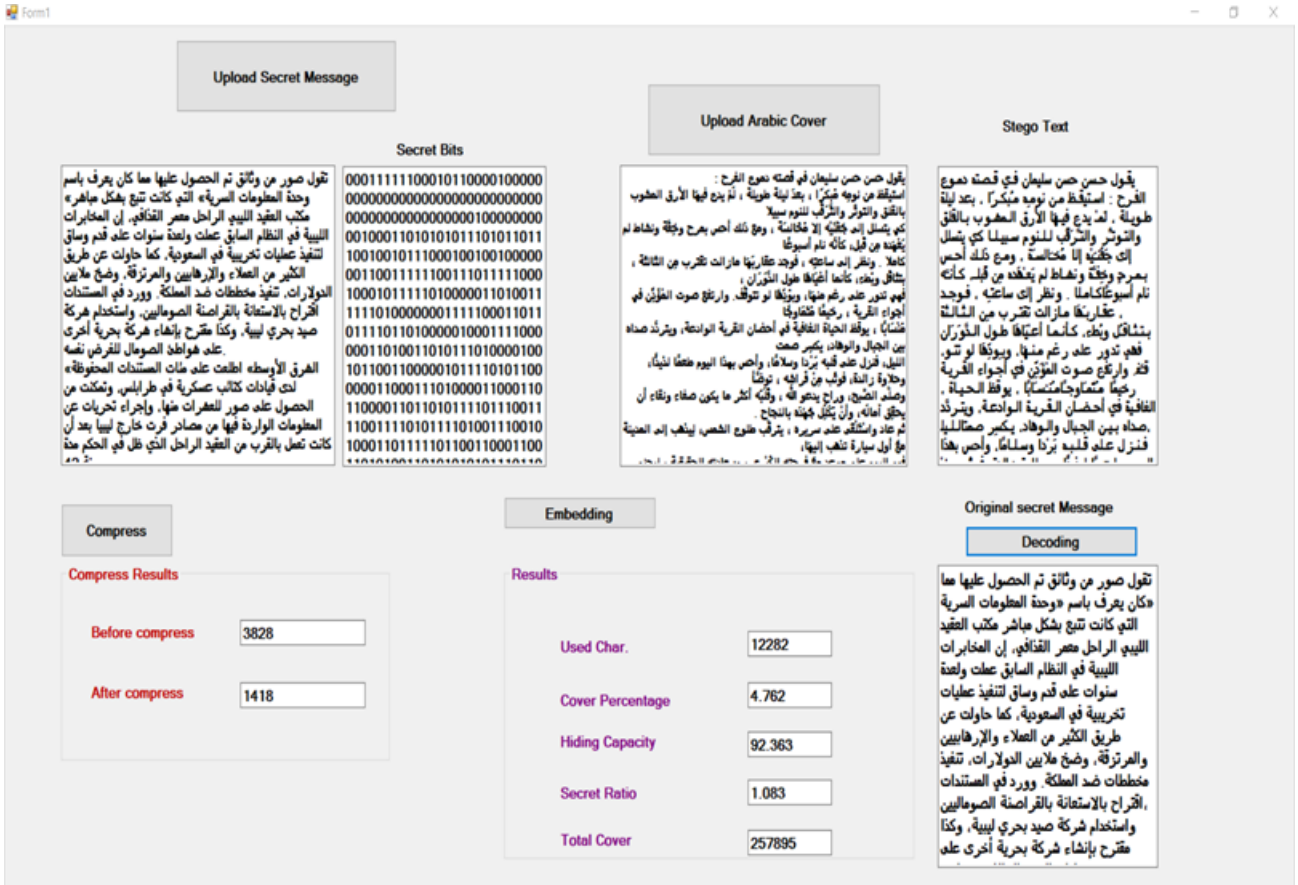

Figure 2: Instant GUI used to apply message 6 with cover 2.

As shown in Table (11), the real used characters in the proposed method are less than the method [17], the best lower values for (secret ratio and cover percentage), and the best values of (hiding capacity) are for the proposed method than the method [17].

Table 11: Results of percent coverage, hiding capacity, and real use of the character for 5 messages and 2 covers, secret message (before and after compression).

The secret message	The cover	The real use of char. (byte)		Percent coverage		Hiding capacity (byte)		Secret ratio (byte)		Secret message (byte)	
		Proposed method	Method [17]	Proposed method	Method [17]	Proposed method	Method [17]	Proposed method	Method [17]	Before comp.	After comp.
M1	C1	266	268	1.058	1.066	84.2	83.5	1.188	1.196	28	-
M2	C1	2104	2435	8.368	9.685	87.072	75.236	1.148	1.329	346	229
M3	C1	5220	5872	20.761	23.354	87.050	77.384	1.149	1.292	1678	568
M4	C2	8503	9549	3.297	3.703	92.861	82.689	1.077	1.209	2391	987
M5	C2	14111	15621	5.472	6.057	92.183	83.273	1.085	1.201	3212	1626
M6	C2	12282	13770	4.762	5.339	92.363	82.382	1.083	1.214	3828	1418

5. Conclusion. A new method; has been proposed for concealing secret messages within the Arabic letters (fiery, airy, earthy, and watery), by using kashida and a few Unicode letters, which will not catch readers' attention. Many aspects of this work, the most important are:

1- High capacity for concealing relatively large size secret messages in Arabic letters.
2- Hiding two secret elements (bits) rather than just one.
3- Hiding in any language.

References

References

[1] J. Shukla, "Method and apparatus for end-to-end secure data communication." *U.S. Patent Application* 09/910,667, filed April 11, 2002.

[2] N. Agrawal, and S. Marios. "Biometric data hiding: A 3-factor authentication approach to verify identity with a single image using steganography, encryption, and matching." In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 85-92. IEEE, 2009.

[3] A. Al-Mohammad, "Steganography-based secret and reliable communications: Improving steganographic capacity and imperceptibility." *Ph.D. diss., Brunel University, School of Information Systems, Computing and Mathematics Theses*, 2010.

[4] Arvind K., Km. P., "Steganography- A Data Hiding Technique", *International Journal of Computer Applications (0975 – 8887)*, Volume 9– No.7, November 2010.

[5] M.Y. Elmahi, and M.H. Sayed, "Text steganography using compression and random number generators." vol, 6, pp.259-263, 2017.

[6] H. Alshahrani, S. Mesfer, and W. George. "Hybrid Arabic text steganography." *International Journal of Computer and Information Technology 6, no. 6*, 329-338, 2017.

[7] S. V. Broecke, "A Scheme of Heaven: The History of Astrology and the Search for Our Destiny in Data." *Science 367, no. 6475*, 255-255, 2020.

[8] R. Thabit, I.U Nur, MY Sharifah, A. Aziah, A. R. Nuur, and D. Roshidi. "A comparative analysis of Arabic text steganography." *Applied Sciences*, 11, no. 15, 6851, 2021.

[9] H.K. Tayyeh, S.M. Mohammed, and AS Ahmed AL-Jumaili. "Novel steganography scheme using Arabic text features in Holy Quran." *Int. J. Electr. Comput. Eng 9, no. 3*, 1910, 2019.

[10] A. Taha, S.H. Aya, and M.S. Mazen, "A high-capacity algorithm for information hiding in Arabic text." *Journal of King Saud University-Computer and Information Sciences 32, no. 6*, 658-665, 2020.

[11] A. Odeh, and E. Khaled, "Steganography in Arabic text using zero width and kashida letters." *AIRCC's International Journal of Computer Science and Information Technology 4, no. 3*, 1-11, 2012.

[12] Y.M. Alginahi, N.K. Muhammad, and T. Omar. "An enhanced Kashida-based watermarking approach for Arabic text documents." In *2013 International Conference on Electronics, Computer, and Computation (ICECCO)*, pp. 301-304. IEEE, 2013.

[13] M.A. Ala'a, and Q.OA. Jehad, "A meliorated Kashida-based approach for Arabic text steganography." *AIRCC's International Journal of Computer Science and Information Technology 9, no. 2*, 99-112, 2017.

[14] A. S. Anes, "Text steganography using extension Kashida based on the moon and sun letters concept," *International journal of advanced computer science and applications*, vol. 8, no.8, pp.286-290, 2017.

[15] A. Taha, S.H. Aya, and S.M. Mazen, "A high capacity algorithm for information hiding in Arabic text." *Journal of King Saud University-Computer and Information Sciences 32, no. 6*, 658-665, 2020.

[16] A. Ditta, A. Muhammad, N. Shahid, G.R. Khurra, A.K. Muhammad, and I. Zafar, "A secure and size efficient algorithm to enhance data hiding capacity and security of cover text by using Unicode." *Journal of King Saud University-Computer and Information Sciences*, 2020.

[17] F.R. Shareef, "Text steganography based on Noorani and Darkness," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 12, no.3, 2021.

[18] H., Mushera, "Discover your personality from the letters of your name," (2011, 4 24). Retrieved from https://www.youm7.com

[19] A. S. Anes, "Text steganography using extension Kashida based on the moon and sun letters concept," *International journal of advanced computer science and applications*, vol. 8, no.8, pp.286-290, 2017.

[20] A. Gutub, "High Capacity Steganography Tool for Arabic Text Using 'Kashida," vol.2, no 3, pp. 107-118, 2010.

[21] F.R. Shareef, "A novel crypto technique based cipher-text shifting," *Egyptian Informatics Journal, vol. 21, no. 2*, pp.83-90, 2020.