Egyptian Informatics Journal (2014) **15**, 79–87

Cairo University

Egyptian Informatics Journal

www.elsevier.com/locate/eij
www.sciencedirect.com

FULL-LENGTH ARTICLE

An improved algorithm for information hiding based on features of Arabic text: A Unicode approach

A.A. Mohamed *

Salman Bin Abdulaziz University, Kingdom of Saudi Arabia

Received 1 August 2013; revised 29 March 2014; accepted 9 April 2014
Available online 19 May 2014

KEYWORDS

Information hiding;
Information security;
Arabic language;
Steganography;
Unicode

Abstract Steganography means how to hide secret information in a cover media, so that other individuals fail to realize their existence. Due to the lack of data redundancy in the text file in comparison with other carrier files, text steganography is a difficult problem to solve. In this paper, we proposed a new promised steganographic algorithm for Arabic text based on features of Arabic text. The focus is on more secure algorithm and high capacity of the carrier. Our extensive experiments using the proposed algorithm resulted in a high capacity of the carrier media. The embedding capacity rate ratio of the proposed algorithm is high. In addition, our algorithm can resist traditional attacking methods since it makes the changes in carrier text as minimum as possible.

1. Introduction

With the rapid growth and wide application of Internet and smart devices in recent years, the network delivery of large bulk of novels, journals and documents in digital way has become more popular. Most of this kind of information can be categorized or directly converted to text formatting so that they are easily copied. Thus, how to protect the transmitted information effectively on the internet has become more important. In addition, the text steganography has become an effective way to solve such a problem very well.

Steganography is a Greek word coming from cover text. "Stegano" means hidden and "Graptos" means writing [1]. In steganography, the secure data will be embedded into another object; so the eavesdropper cannot catch it which literally means "covered writing".

Obviously the purpose of steganography is to avoid drawing suspicion to the transmission of hidden information. A message is a hidden information in the form of plain text, cipher text, images or anything that can be encoded into a bit stream. This message is embedded in a cover-carrier to create a stego-carrier. A possible formula of the process may be represented as follows:

$$\text{Stego medium} = \text{Cover medium} + \text{Embedded message} + \text{Stego key}$$

* Tel.: +966 201023303373.
E-mail address: dr_ashrafa@yahoo.com
Peer review under responsibility of Faculty of Computers and Information, Cairo University.

Production and hosting by Elsevier

1110-8665 © 2014 Production and hosting by Elsevier B.V. on behalf of Faculty of Computers and Information, Cairo University.
http://dx.doi.org/10.1016/j.eij.2014.04.002

Different types of cover media including image, Sound, video as in [2–4] and text can be used in steganography. Choosing carrier file is very sensitive as it plays a key role to protect the embedded message.

But using text is preferred over other media, because the texts occupy lesser space, communicate more information.

Text steganography represents the most difficult type as there is generally lack of data redundancy in the text file in comparison with other carrier files. The existence of such redundancy can help increase the capacity of hidden data size.

Furthermore, text steganography depends on the language as each language has its own unique characteristics, which are completely different from other languages. For example, the letter shape in English language does not depend on its position in the word. At the same time, Arabic letters have different forms which depend on letter positioning.

After data embedding, the text with secrets which is called stego-text, is sent from sender side to receiver side over the Internet. The security concept is central around the idea that no one can easily discover the secrets embedded into the stego-text by using statistical computation or other methods of detection.

There are three important parameters (criteria) in designing steganography systems [5]:

I. Perceptual transparency.
II. Robustness.
III. Hiding capacity.

I. The security or Perceptual transparency refers to the ability of an eavesdropper to figure, or suspect the hidden information easily. We can achieve high security by minimizing the embedding Impact (distortion). Intuitively, we can try to achieve this by minimizing the distortion between the cover text and stego text and by restricting the distortions to the portions of the cover text that are difficult to model.
II. The robustness refers to the ability of protecting the unseen data from corruption especially when transmitted through the internet.
III. The capacity or the embedded rate refers to the ability of a cover media to store secret data, which can be measured by the amount of secret data (bits) that can be hidden in a kilo byte of a cover media. It is used to measure the hiding efficiency and is defined by Eq. (1) as follows:

$$\text{Embedded rate(Capacity Ratio)} = \text{Size of hiddeng secrete in bits/ the size of carrier in kilobytes.} \quad (1)$$

One reason of why text steganography is difficult is that text contains little redundancy compared to other media. Another is that humans are sensitive to abnormal-looking text. Text steganographic schemes must be specifically designed to exploit the specific characteristics of the target language, because the grammatical and orthographic characteristics of every language are different.

The following parts of the paper are structured as follows. Section 2 reviews the text steganography methods for Arabic text and the related works. The proposed method is detailed in Section 3. The experiments and discussions are presented in Section 4. Some conclusions are made in Section 5. The future work is given in Section 6.

2. Related works

Most of the text steganography methods are applied to English texts. However, there are a few text steganography methods applied to other languages [1,6,7]. A few works have been done on hiding information in Arabic texts [8–10]. The following is a list of different methods of the works for Arabic text carried out and reported thus far.

2.1. Steganography by shifting points

In Arabic languages, dot is very important and 14 of 28 Arabic letters have one or more dots. In this method, data are hidden in Arabic texts by using these letters as explained in [8,11]. Using the approach of the vertical displacement of the points, we can hide information in the texts. However, we need to use a special font created mainly for this purpose.

2.2. Steganography using Arabic diacritics (Harakat)

Arabic language uses different marks or diacritics, Arabic extension character, (Harakat) which are optional to use. The main reason to use these symbols is to distinguish between words that have same letters (i.e., so that every word pronounced differently). The diacritic such as "Fatha" is used to hide 1. However, the rest of the diacritics are used to hide a 0 because it was found out that "Fatha" represents almost half of the diacritics in any Arabic text as explained in [9,12]. The main disadvantage of this method is that it attracts the attention of the reader.

2.3. Kashida based steganography

In this method, we add extension (Kashida in Arabic) to a word to hold secret bit 'one' and we leave the word without extension (Kashida) to hold secret bit 'zero' as explained in [10,11,13]. Note that letter extension does not have any effect on the writing content. The main disadvantage of this method is that it attracts the attention of the reader, increases the file size and changes the apparent of the text.

2.4. Unicode based steganography

Arabic letters have many forms according to Unicode standard. It is divided into two groups of codes for Arabic letters; the representative code and the code of the possible shapes of the letter. In this method, we use the different possible Unicode values of the same letter to hide bits as explained in [14,15,5]. This method is suitable for internet and modern device as smart phone. But it is weak against traditional attacks. Some techniques of Unicode based steganography provide high capacity and low security [9,11] and vice versa.

The disadvantages of Unicode based steganography can be summarized as follows:

Figure 1 The behavior of shape-determination routine when it modifies stego-text produced by the above technique.

1. Some techniques of Unicode based steganography provide high capacity but it need to change the content of the carrier text drastically such as [15], while the central concept in steganography is that the method have to be statistical undetectable (i.e. we need to make the changes in carrier text as minimum as possible).
2. Some techniques of Unicode based steganography such as [15] change the Unicode of each letter in the target word to hide data so these words appear as words with foreign letters to the shape-determination routine. This routine may allow for automatic selection of the appropriate shape of letter according to context as directed by the software user [16] such that when the user tries to change any letter in this word of the stego text the word is corrupted which draw attention to that text (see Fig. 1).

So we need an algorithm which does not corrupt the shape of the words. However, the proposed algorithm in this research is Unicode based but it is designed to avoid falling into such drawbacks.

2.5. Linguistic based steganography

The linguistic steganography can be classified as follows:

2.5.1. Lexical based steganography

In lexical steganography, secret message is hidden to natural language text by using synonym [17].

In synonym-based approach, the cover text may look legitimate from a linguistics point of view given the adequate accuracy of the chosen synonyms. But reusing the same piece of text to hide a message can raise suspicion and the embedding capacity of information is low [17,18].

2.5.2. Translation based steganography

This scheme hides a message in the errors (noise), which are naturally encountered in a machine translation (MT). The secret message is embedded by performing a substitution procedure on the translated text using translation variations of multiple MT systems [18].

2.5.3. Noise-based approach

Here, typos and ungrammatical abbreviations in a text, e.g., emails, blogs, forums, etc., are employed for hiding data. However, this approach is sensitive to the amount of noise (errors) that occurs in a human writing [18].

3. The suggested algorithm

Before explaining the method, first of all, we will explain the main characteristics of Arabic languages. Secondly, we will expose the Unicode standard of Arabic category. Finally, we will show our steganographic algorithm in details.

3.1. The characteristics of Arabic language

Arabic language is used by nearly two billion people. The Arabic alphabet has 28 letters (see Fig. 2).

There is some disagreement over the number of Arabic letters resulting from different classifications of what is a diacritic and from ignoring some of the letters. Most commonly, the Arabic alphabet is said to have 28 letters (basic 28, sometime substituting "ا" With "أ") or 29 letters (basic 28 plus the Hamza–on- the-line letter constructed from the Hamza letter form) [19].

In Arabic, the letters are connected to each other in the printed texts, while in English the letters are written separately. In English, the letters are written in a left-to-right format but in Arabic the letters are written in a right-to-left format. In Arabic languages, dot is very important and 14 of 28 Arabic letters have one or more dots, while in English only two small letters "i" and "j" have dot [19] (see Fig. 3).

In Arabic, there are 22 letters out of 28 letters which have 4 variants like " ف " (Feh) and " غ " (Ghain) (see Fig. 3).

That is to say Arabic letter can have four different shapes.

1. Standing alone letter.
2. As the first letter in a word.
3. Inside the word, between two other letters.
4. As the last letter in a word.

The shape of each letter is determined by its position in a word.

For example the letter Ghain " غ " is written as " غـ " at the beginning of a word, as " ـغـ " in the middle, as " ـغ " at the end, and as " غ " in the isolated form.

As for the remaining six letters of the Arabic letters like (Thal) " ذ " (Dal) " د ", (Reh) " ر ", (Zain) " ز ", (Waw) " و " and (Alef) " ا " have only 2 variants as follows:

1. As the first letter in a word.
2. As the last letter in a word.

These letters cannot be joined to the succeeding letter, even when they are inside a word. This means that the writer has to lift his pencil, and even if his hand is inside the same word. The next letter will have to be written as if it was the first in a word. The letter comes after these six letters is written as the last one and it must be an isolated letter. If it comes as the first letter it is written as isolated letter.

The following sentence gives an example for the above mentioned explanations (see Fig. 4):

Each red letter refers to an isolated letter in this word of sentence. We used these characteristics of the six letters as hidden keys in our steganographic algorithm which enable us to avoid the drawbacks of the Unicode based algorithm as we will discuss in the proposed algorithm.

3.2. Unicode for Arabic language

In the Unicode standard, the Arabic block has been developed to cover the characters of the languages which use Arabic writing system. This standard has detailed explanations about the implementation methods including letters-connection

أ ب ت ث ج ح خ د ذ ر ز س ش ص ض ط ظ ع غ ف ق ك ل م ن ه و ي

Â b t θ j H x d ð r z s š S D T Ď ς γ f q k l m n h w y

Figure 2 Arabic letters and their transliterate.

Figure 3 22 of Arabic letters have 4 shapes.

method, the exhibition of the right-to-left and bi-directional texts [19,16,20,21].

In this standard, each Arabic letter has a representative code and it has another code for its shapes in the word as we said before. For example, the code of letter Khah ("خ") in the Unicode standard is 062E and this code represents this letter and the codes of different forms are FEA5 for the isolated form ("خ"), FEA6 for the final form ("ـخ"), FEA7 for the initial form ("خـ") and FEA8 for the medial form ("ـخـ") as shown in Tables 1 and 2.

In the Unicode standard, the code of the representative letters is used only to save data in the digital media. Table 1 shows Unicode for each Arabic representative letter. The program (i.e. the shape-determination routine [16,21]) which display the Arabic text selects the correct shape for each letter regards to its position in the word from the Table 2 to display the correct shape of the letter in this word.

3.3. Our method

As we said in Section 3.1, In Arabic there is a group of six letters written isolated in a text on the condition that each letter is the first letter in a word. Any letter succeed them has to be written as an isolated if it is the last letter in this word. We use the isolated letters of this type as hidden keys in Arabic texts written in Unicode format. So the hiding program looks for these letters in the word displayed in the carrier text. So we can hide data in the carrier text by using this type of letters without any noticeable change in the target word. Simplifying the complexity of algorithm, one has to consider the isolated letters at the beginning of the words and at the end of the words not all the isolated letters in the words.

For example we can hide a packet of 9 zeros or we can hide a packet of 8 ones in the following sentence using the proposed algorithm as follow (see Fig. 5).

3.3.1. How to hide a secret code in a plain text

Suppose the cover media is the following piece of poetry (i.e. plain text) and we need to hide the secret code '11010' in the following Arabic text:

صـلاح أمرك للأخلاق مرجعه فقـوم النفس بـالأخلاق تسـتقم

According to our algorithm we do the following to hide this secret code:

Arabic Sentence	إنما الأمم الأخلاق ما بقيت فإن هم ذهبت أخلاقهم ذهبوا															
Separated Words	إنما		الأمم	الأخلاق		ما	بقيت	فإن	هم	ذهبت	أخلاقهم	ذهبوا				
Isolated letters in each word	إ		ا	ق	ا	Non	Non	ن	Non	ذ	أ	ذ	ا			

Figure 4 Isolated letters in Arabic Sentence.

Table 1 A part of the first group of Arabic letters (representative letter) in Unicode table.

	0	1	2	3	4	5	6	7	8	9	0A	0B	0C	0D	0E	0F
600	ـ	ـب	ـت	ـث								ـف	ـق	ـك	ـل	ـم
620		ء	آ	أ	ؤ	إ	ئ	ا	ب	ة	ت	ث	ج	ح	خ	د
640	-	ف	ق	ك	ل	م	ن	ه	و	ى	ي					

Table 2 A part of the second group of Arabic letters (different letter shapes) in Unicode table.

	0	1	2	3	4	5	6	7	8	9	0A	0B	0C	0D	0E	0F
FE80	ء	آ	آ	أ	أ	ؤ	ؤ	إ	إ	ئ	ئ	ئ	ئ	ا	ا	ب
FEA0	ـجـ	ـحـ	ـجـ	ـحـ	ـحـ	ـخـ	ـخـ	ـخـ	ـخـ	ـدـ	ـدـ	ذ	ذ	ر	ر	ز
FEC0	ـضـ	ـطـ	ـطـ	ـطـ	ـطـ	ـظـ	ـظـ	ـظـ	ـظـ	ـعـ	ـعـ	ـعـ	ـعـ	غ	غ	غ
FEE0	ـلـ	ـمـ	ـمـ	ـمـ	ـمـ	ـنـ	ـنـ	ـنـ	ـنـ	ـهـ	ـهـ	ـهـ	ـهـ	و	و	ى

Arabic Text	إنما الأمم الأخلاق ما بقيت فإن هم ذهبت أخلاقهم ذهبوا											
Separated Words	ذهبوا		أخلاقهم	ذهبت	هم	فإن	بقيت	ما	الأخلاق		الأمم	إنما
The number of Isolated letters in each word	ا	ذ	أ	ذ	-	ن	-	-	ق	ا	ا	إ
The number of changed letters is zero	□	□	□	□		□			□	□	□	□
The Maximum number of hidden zero is 9	0	0	0	0		0			0	0	0	0
The number of changed letters is 2	■	□	□	□		□			□	□	□	■
The Maximum number of hidden ones is 8	1		1	1		1			1	1	1	1

Figure 5 The maximum number of zeros and the maximum number of ones we can hide in the above Arabic Sentence using the proposed algorithm. Where ■ means that we changed the Unicode of this letter, □ means we did not change the Unicode of this letter and '-' means word with no isolated letter.

Step 1: Convert the binary code of the secret message into run length encoding format as follow:

$$\text{Binary string } (11\ 0\ 1\ 0) \xrightarrow{\text{Run length encoding}} (21\ 10\ 11\ 10)$$

where "21" means we have a packet of ones consists of two ones and "10" means we have a packet of zeros consists of one zero and so on.

Since our Hidden technique considers the secret code as packets of zeros and ones. So we hide the secret code as a group of ones or group of zeroes or vice versa as follow:

Step 2: we read the first pair of hidden secret (i.e. 21) which is packet of two ones and according to our algorithm.

We get a word of isolated letter from cover text to change its code as the beginning of hidden process for this packet.

In this case the word 'صـلاح' has one isolated letter which can be changed to mark the beginning of the hidden process to this packet by changing the letter 'ح'. Now we have hidden the first "1" in this packet.

Step 3: we get another word 'أمرك' which has two isolated letters and we leave the first letter 'أ' unchanged as an indication of hiding '1' and we change the last letter 'ك' as the end of the hiding process of this packet.

Step 4: we read the second pair of hidden secret (i.e. 10) which is zero packet of one zero and according to our algorithm we get a word (i.e. 'للأخلاق') with isolated letter 'ق' from the cover text and to leave it as the beginning of hidden '0' and the end of this process.

Step 5: we read the third pair of hidden secret (i.e. 11) which is packet of one '1' and according to our algorithm.

We get a word of isolated letter from cover text and we change its Unicode as the beginning of hidden process for this packet. In this case the word 'فقوم' has one isolated letter which can be changed to mark the beginning of hidden process to this packet by changing the letter 'م'. Now we have hidden "1" for this packet.

Step 6: we get another word 'النفس' which has one isolated letter we change this letter 'ا' as indication of the end of the hiding process of this packet.

Step 7: we read the second pair of the hidden secret (i.e. 10) which is zero packet of one zero and according to our algorithm we get a word (i.e. 'بالأخلاق') with isolated letter 'ق' from cover text and we leave it as the beginning of hidden '0' and the ending of this process. In Fig. 6 we summarized the above steps.

From the previous example we can summarize the hidden process using the proposed algorithm as follow:

i. To hide a packet of zeros which contains N zeros we have to leave N isolated letters unchanged from less or equal number of words with isolated letters from the cover text and If the last word taken from them contains one extra isolated letter, we change its Unicode to its corresponding Unicode from Table 2 as an indication of the end of the encoding process.
ii. To hide a packet of ones which contains N ones, we have to get a word from the carrier with isolated letter or two isolated letters (i.e., one at the front of the word and the other at the end of the word). We change the Unicode of these letters from the representative letter Unicode to their corresponding Unicode from Table 2 (i.e. the code which determines the correct letter shape in the word). We consider such change as the beginning of the encoding process of the one's packet. Then we have to leave $N-1$ isolated letters unchanged, disregarding any number of words taken from the carrier text. If the last word taken from them contains one extra isolated letter, we change its Unicode to its corresponding Unicode from Table 2 as an indication of the end of the encoding process, else we end this process by changing the isolated letters from the next word from the carrier text.

3.3.2. How to extract the secret code from the stego text

Suppose we want to extract the secret message from the stego text produced from the previous example. According to the proposed algorithm we do the following:

Step 1: We read a word with isolated letters from the stego text. If all the isolated letters in this word are changed, we have a packet of ones: otherwise, we have a packet of zeros.

Cover Text	صـلاح أمـرك للأخـلاق مرجعه فقـوم النفس بـالأخلاق تسـتقم								
Separated Words	صـلاح		أمـرك		للأخـلاق	مرجعه	فقـوم	النفس	بـالأخلاق	تسـتقم
Isolated letters in each word	ح	أ	ك		ق	–	م	ا	ق	–
The changing location	■	□	■		□		■	■	□	
The hidden Secrete	1	1		0			1		0	
Stego Text	صـلاح أمـرك للأخـلاق مرجعه فقـوم النفس بـالأخلاق تسـتقم									

Figure 6 How to hide '11010' in Arabic Sentence where ■ means that we changed the Unicode of that letter, □ means we did not change the code of this letter and – means not used word.

Stego Text	صـلاح أمـرك للأخـلاق مرجعه فقـوم النفس بـالأخلاق تسـتقم								
Separated Words	صـلاح		أمـرك		للأخـلاق	مرجعه	فقـوم	النفس	بـالأخلاق	تسـتقم
Isolated letters in each word	ح	أ	ك		ق	–	م	ا	ق	–
The changed locations	■	□	■		□		■	■	□	
The hidden Secrete	1	1		0			1		0	
The secret message	1 1 0 1 0									

Figure 7 How to extract a secret message from stego text where ■ means that this letter is changed, □ means this letter is not changed and – means not used word.

In our example we get the first word in the stego text which has one changed isolated letter (i.e. the word **'صـلاح'** has one changed isolated letter '**ح**'). So, we have a packet of ones and we initiate the extracting string by "1".

Step 2: we get another word **'أمرك'** which has two isolated letters and the first letter 'أ' is unchanged so we add another '1' to the extracting string to become '11' and the second letter(i.e. last letter '**ك**') is changed which is as indication to the end of that packet.

Step 3: we get another word with isolated letter (i.e. The word **'للأخلاق'** has one unchanged isolated letter '**ق**') So, we have a packet of zeros and we add '0' to the extracting string to become "110".

Step 4: we get another word of one isolated letter (i.e. the word **'ففوم'** has one changed isolated letter '**م**'). So, we have a packet of ones and we add '1' to the extracting string to become "1101".

Step 5: we get another word **'النفس'** which has one isolated letters **'ا'** which is changed as indication to the end of this packet.

Step 6: we get another word of one isolated letter (i.e. the word **'بالأخلاق'** has one unchanged isolated letter '**ق**') So, we have a packet of zeros and we add '0' to the extracting string to become "11010".

Simply speaking, the technique of extraction is opposite to encoding technique as we have shown in Fig. 7.

From the above example we notice that the proposed algorithm encodes the hidden process and it may change the code of an isolated letter to hide '1' in some place and it may leave the same isolated letter to hide '1' in another place which made the proposed algorithm hard to break statistically.

3.3.3. Encoding algorithm

In this subsection we present the pseudo-code of the proposed Arabic steganographic algorithm.

The encoding algorithm consists of two sub algorithms; the first for one's packet and the second for zero's packet.

If we want to encode a packet of zeros we follow the sub1; otherwise we follow sub2.

***Encoding Algorithm*: Hiding secret bits**
***Input*: Cover Text, secret bit stream in run length format**
***Output*: Stego Text**

```
/*************************************************************/
/* The pseudo-code of subl for encoding a packet of N zeroes */
/*************************************************************/
Function Subl()
1. Get a word from carrier text which contains isolated letters.
2. N = N − number of isolated letters in this word
3. If N = 0 then Return
Else
If N < 0 then
{Change the last isolated letter Unicode in the word to their
Corresponding Unicode;
Return;}
4. Loop 1
End Subl
/*************************************************************/
/*The pseudo-code of sub2 for encoding a packet of N ones*/
/*************************************************************/
Function Sub2()
1. Get a word from carrier text which contains isolated letters.
2. Change the isolated letters Unicode in the word to their corresponding Unicode.
3. Get the next word from carrier text which contains isolated letters.
4. N = N − number of isolated letters in word
5. If N < 0 then
{Change the isolated letters Unicode in the word to their corresponding Unicode;
Return;}
6.If N = 0 then
{If the word contains one isolated letter
then
Change the isolated letter Unicode in the word to their corresponding Unicode;
Return;}
Else
If the word contains two isolated letter
then
{Change the last isolated letter Unicode in the word to their corresponding
Unicode;
Return;}
7. loop 3
End Sub2
Loop Step II
End Encoding Algorithm
```

Step I: Initiate the pointer of the reading from Carrier text to the first word in the text.
Step II: while not end of the secret bit stream do the following.
Step III: Read a pair of numbers (N, B) from secret bit stream where B is '1' or '0' and N is the number of B in this packet.
Step III: If B = zero then it is zeros packet else it is ones packet.
Step IV: If ZeroPacket then Sub1 Else Sub2

3.3.4. Extraction algorithm

The extracting algorithm consists of two subalgorithms; the first for extracting from one's packet and the second for extracting from zero's packet. If we have a packet of zeros we follow the sub1; otherwise we follow sub2.

Extraction Algorithm: **Discover secret bits**
Input: **Stego Text**
Output: **secret bit stream**

```
/*************************************************************/
/* The pseudo code for subl to extract a packet of N zeroes */
/*************************************************************/
Function Subl()
1. N = Number of isolated letters in this word;
2. If the word contains two isolated letter and the last one was changed then Return N;
3. Get a word from Stegano text which contains isolated letters;
4. If the word contains one changed isolated letter adjust the pointer of reading the
 word to the last word then Return N;
5. If the word contains two isolated letter and the last one was changed
then Return N + 1;
6. N = N + Number of isolated letters in this word;
7. Loop 3;
End Subl
/*************************************************************/
/* The pseudo code for sub2 to extract packet of N ones */
/*************************************************************/
Function Sub2()
1. N = 1;
2. Get a word from Stegano text which contains isolated letters;
3. If the word contains one changed isolated letter Return N;
4. If the word contains two isolated letter and the last one was changed
then Return N + 1;
5. N = N + Number of isolated letters in this word;
6. Loop2;
end Sub2
Loop Step II
End Extraction Algorithm
```

Step I: Initiate the pointer of the reading from stego text to the first word in the text.
Step II: read a word with isolated letters from the stego text.
Step III: if all the isolated letters in this word are changed, we have a packet of ones: otherwise, we have a packet of zeros.
Step 3: If we have a packet of zeros we follow the sub1; otherwise we follow sub2.

4. Experimental result

In this section we will explain all the experimental results. In the adopted method, the information is hidden in Arabic texts using the Unicode standard. We test our method on some Arabic text files downloaded from some Arabic web sites. They include a variety of news like sports, health, science, politics, technology and literature. The files resources are selected for computing the capacity ratio of the method for hiding data. We calculate the capacity ratio as follows:

$$\text{Capacity Ratio} = \text{Size of hidden secrete in bits} / \text{the size of carrier in kilobytes.}$$

Table 3 shows the results. Our method capacity is approximately equal to 52 bit/kB. As it is seen in Table 3, the adopted method capacity is high. We will discuss some advantages of this method in the next section.

Table 3 The experimental Results of the proposed algorithm.

File name	Text size in kilobyte	Our method text capacity (Bit)	Our method capacity ratio (b/kB)
art01	10.50	556	52.95
art02	4.42	229	51.78
art03	2.10	119	56.65
econ01	13.85	711	51.34
econ02	7.08	378	53.35
econ03	16.00	806	50.375
health01	5.33	296	55.58
health02	12.28	650	52.94
health03	8.73	439	50.26
news01	8.31	467	56.19
news02	3.59	189	52.59
news03	14.12	761	53.91
sport01	3.92	189	48.20
sport02	25.15	1282	50.97
sport03	3.36	157	46.71
tech01	14.57	745	51.14
tech02	7.42	384	51.78
tech03	16.11	822	51.03
var01	15.36	761	49.54
var02	21.95	1037	47.25
var03	16.59	883	53.22

5. Conclusion

This paper presents a new text steganography method for Arabic texts or any languages written in Arabic letters like Urdu (the official language of Pakistan), Persian (the official language of Iran) and Pashto (the official language of Afghanistan) [22] based on the features of Arabic text which offers high capacity, more security, and good robustness (i.e. we introduce a compromised solution to our problem).

Experimental results show the following advantages of the suggested algorithm:

(1) The proposed algorithm capacity ratio is approximately equal to 52 bit/kB which is high and the embedding size of our method can be changed flexibly according to the carrier text size.
(2) The proposed algorithm hides information with minimum change in the cover text hence the stego-texts generated by the proposed algorithm will draw less attention (i.e. It satisfies perceptual transparency).
(3) The hidden process is robustness because the stego text will not change during copy and paste between computer programs, and the data hidden in texts remain intact during these operations.
(4) The proposed algorithm can resist traditional attack methods since it hide the secrete message as packets of zeros and ones in run length encoding format which enable us to encode the secrete message by using minimum changing in cover text and this technique is a new one, therefore the possibility of breaking this method is very low [23].
(5) The proposed algorithm is Unicode based which is suitable for internet and smart devices but it is designed to avoid falling into such drawbacks of Unicode based algorithms such as [15].
(6) The adopted method does not change the size of the text and it is not dependent on any special format and it does not require a specific type of font as in [11,6,7].

6. Future work

As a future work we can increase the capacity ratio more than 100 bit/kB by considering all the isolated letters in every word in text according to our initial estimation according to our experiments. Beside the electronics text steganography, this method can also be applied to hard copy. For this, we can use a special type of font for isolated letters only to mark the beginning of encoding zeros and ones as we discussed in Section 3.3 beside the normal fonts which is used in the text. After hiding the information we print the documents. To extract data from the hard copy document, first we scan the document and then unhide the information by computer, using the proposed algorithm.

References

[1] Kessler G, Hosmer C. An overview of steganography. Advan Comp 2011:83, ISSN: 0065-2458.
[2] Mohan Malini, Anurenjan PR. A new algorithm for data hiding in images using contourlet transform. Rec Advan Intell Comput Syst (RAICS) IEEE 2011:411–5.
[3] Gopalan K. Audio steganography using bit modification. In: Proceedings of the IEEE international conference on acoustics, speech, and signal processing, (ICASSP '03), vol. 2; 2003. p. 421–4.
[4] Doërr G, Dugelay JL. Security pitfalls of frame by-frame approaches to video watermarking. IEEE Trans Signal Process, Suppl Secure Media 2004;52(10):2955–64.
[5] Shahreza MS, Shahreza MH. An improved version of Persian/Arabic text steganography using "La" word. In: Proceedings of IEEE 6th national conference on telecommunication technologies; 2008. p. 372–6.
[6] Shirali-Shahreza M. A new method for steganography in HTML files. In: Proceedings of the international joint conference on computer, information, and systems sciences, and engineering (CISSE), December 2005. p. 247–51.
[7] Khairullah Md. A novel text steganography system using font color of the invisible characters in Microsoft word documents. In: Second international conference on computer and electrical engineering, vol. 1; 2009. p. 482–4.
[8] Odeh Ammar, Alzubi Aladdin, Hani Qassim Bani, Elleithy Khaled. Steganography by multipoint Arabic letters. In: Systems, applications and technology conference (LISAT); 2012. p. 1–7.
[9] Bensaad Mohamed Lahcen, Yagoubi Mohamed Bachir. High capacity diacritics-based method for information hiding in Arabic text. In: International conference on innovations in information technology; 2011. p. 433-436.
[10] Al-Azawi AF, Fadidhil MA. Arabic text steganography using Kashida extensions with Huffman code. J Appl Sci 2010;10 ():436–9.
[11] Al-Nazer Ahmed, Gutub Adnan. Exploit Kashida adding to Arabic e-text for high capacity steganography. In: Proceedings of the third international conference on network and system security NSS '09. IEEE; 2009. p. 447–51.
[12] Aabed MA, Awaideh SM, Elshafei AM, Gutub AA. Arabic diacritics based steganography. In: Proceedings of the international conference on signal processing and communications (ICSPC 2007); 2007. p. 756–9.
[13] Al-Haidari Fahd, Gutub Adnan, Al-Kahsah Khalid, Hamodi Jameel. Improving security and capacity for Arabic text steganography using 'Kashida' extensions. In: The 7th ACS/IEEE international conference on computer systems and applications. Rabat; 2009. p. 396–9.
[14] Por Lip Yee, Wong KokSheik, Chee Kok Onn. UniSpaCh: a text-based data hiding method using Unicode space characters. J Syst Softw 2012;85:1075–82.
[15] Shirali-Shahreza M, Shirali-Shahreza S. Persian/Arabic Unicode text steganography. In: The fourth international conference on information assurance and security. IEEE; 2008. p. 62–6.
[16] IBM web site. Automatic shaping program <http://www-01.ibm.com/software/globalization/guidelines/i1.html> [last visited: 01.03.14].
[17] Alabish Ahmad, Goweder Abdulbaset, Enakoa Anes. A universal lexical steganography technique. Int J Comp Commun Eng 2013;2 ():153–7.
[18] Listega Desoky A. List-based steganography methodology. Int J Inform Sec 2009;247–61.
[19] Habash Nizar. Introduction to Arabic natural language processing. A Publication in the Morgan & Claypool Publishers series; 2010 [ISBN: 9781598297959].
[20] Gillam Richard. Unicode demystified: a practical programmer's guide to the encoding standard. Addison Wesley; 2002 [ISBN: 0-201-70052-2].
[21] The Unicode Consortium. The Unicode Standard <http://www.unicode.org/charts/PDF/U0600.pdf> [last visited: 01.03.14].
[22] Talip Munire, Jamal Anwar, Wen-Qiang Guo. A proposed steganography method to Uyghur script. In: International conference on cyber-enabled distributed computing and knowledge discover; 2012. p. 125-128.
[23] Lingjun Li, Liusheng Huang, Xinxin Zhao, Wei Yang, Zhili Chen. A statistical attack on a kind of word-shift text-steganography. In: International conference on intelligent information hiding and multimedia signal processing, 2008, IIHMSP'08. IEEE; 2008. p. 1503–7.