ISSN: 0067-2904

Text Steganography Method Based On Modified Run Length Encoding

Suhad Malallah Kadhem*
Computer Science Department, Technology University, Baghdad, Iraq

Abstract

Data hiding (Steganography) is a method used for data security purpose and to protect the data during its transmission. Steganography is used to hide the communication between two parties by embedding a secret message inside another cover (audio, text, image or video). In this paper a new text Steganography method is proposed that based on a parser and the ASCII of non-printed characters to hide the secret information in the English cover text after coding the secret message and compression it using modified Run Length Encoding method (RLE). The proposed method achieved a high capacity ratio for Steganography (five times more than the cover text length) when compared with other methods, and provides a 1.0 transparency by depending on some of the similarity measures of Steganography.

Keywords: Steganography, RLE, Security, Text steganography.

طريقة اخفاء نص بالاعتماد على ترميز طول التكرار المحدثة

سهاد مال الله كاظم*
قسم علوم الحاسوب، الجامعة التكنولوجية، بغداد، العراق

الخلاصة:

إخفاء البيانات (Steganography) هي طريقة تستخدم لاغراض أمنية البيانات ولحمايتها أثناء الإرسال. Steganography يستخدم لإخفاء وجود الإتصال بين جهتين وذلك بتضمين الرسالة السرية داخل الغطاء(صوت، نص، صورة او فيديو). في هذه البحث تم اقتراح طريقة جديدة للاخفاء داخل نص بلاعتماد على معرب و الترميز(ASCII) للاحرف الغير مطبوعة لإخفاء المعلومات السرية في نص الغطاء الانكليزي بعد ترميز الرسالة السرية وضغطها باستخدام طريقة ترميز طول التكرار(RLE) المحدثة. ان الطريقة المقترحة حققت نسبة عالية من السعة في الاخفاء (وصلت لخمسة مرات ضعف طول النص الغطاء) مقارنة بالطرق الاخرى، وكذلك وفرت 1.0 للشفافية بالاعتماد على بعض مقاييس التشابه للاخفاء.

1. Introduction:

Steganography is an art of sending secret messages over a public channel so that their presence is not revealed by a third party [1]. When information hiding methods are utilized, even if an un authenticated intrudes the sent round media, he couldn't infer that there is communication since it is implemented in a hidden way. Steganography overcomes the limitations of cryptography by hiding the message in an unsuspicious media called cover [2]. In the term of steganography, the embedded data refer to the information to be concealed in the cover data. The stego data are the data containing both the cover and the embedded information. The term cover is utilized to describe the original text,image, audio, or video [3]. There are three sides in information hiding models which are related with each other: capacity, security, and robustness. Capacity refers to the size of the data that is able to be embedded in the medium, whereas security is essential when a secret communication is still secret and imperceptible by eavesdroppers and finally, the robustness could be defined by the resistance of the stego-medium against the tampering [4].

*Email:suhad_malalla@yahoo.com

Natural language processing (NLP) is an artificial intelligence branch It includes natural language understanding and generation. NLP has some levels of analysis for understanding natural language processing, those levels are: Phonology, morphology, syntax, semantics and pragmatics. Syntax level (parser) that use the rules (grammar) of natural language for combining words into legal phrases and sentences [5].

Run Length Encoding (RLE) is a compression technique which is used for a given file that contains many redundant data. The input file or message is called run which is encoded into two bytes. The first byte contains the number of times for a given character appears in the run. The second byte represents the value of the character [6].

There are several different characters, such as a space and tab, which are not normally displayed on the screen, these characters are called non printed characters, for instance, the special character to determine the end of a line or the end of a paragraph, and so on[7]. Since most of these characters do not appear when written so these characters are utilized to hide the secret information in the proposed system.

In this paper a new text steganography method is proposed. Since the secret information needs to be encoded and compressed to be hidden in an efficient way in the cover message (English message), ASCII and binary representations are utilized to make encoding and a new improvement to run length encoding method is done to compress the secret information after encoding. Also in the proposed method, Parser is utilized to increase the security level since it is used to generate a dynamic secret key in order to make a rotation to the data (tables) that the proposed modified Run Length Encoding (MRLE) based on.

The rest of this paper is organized as follows: Section 2 presents a brief explanation about the text steganography, section 3 presents different approaches that are related to the text steganography, section 4 describes the details of the proposed method including the idea and the required algorithms, section 5 describe the implementation of the proposed method, section 6 presents the experimental results and finally the main conclusions are summarized in section 7.

2. Text Steganography

Steganography can be categorized into different kinds: image steganography, text steganography, audio steganography and video steganography depending on which cover media are utilized to embed data[8]. Text steganography is approved to be the difficult because of the restrictions of the redundant information that shown in image, audio or a video file. The body of text depositions is corresponding to what was noticeable, while in other kinds like pictures; the structure of deposition is distinct from what was noticeable. Therefore, in such depositions, information could be hidden by producing alterabilities in the body of the document without working a sensible alterability in the output [9]. Unrecognized alterabilities could be made to an image or an audio file, but, in text files, even extra character or punctuation could be marked by a random reader [10]. Storing text file don’t need large memory and it’s quick as well as effortless when compared to the other types of steganography methods. Text steganography could be broadly categorized into three types: Format based, Random and Statistical generation, and Linguistic methods [11].

3. Related Work

The following are some of the related works that are relevant to the approach of this paper (text steganography):

In 2010, Adnan A. , et al. [12], has proposed a new method to embed a secret information into an Arabic text cover media utilizing an Arabic extension character(Kashida). The presented method is an attempt to maximize the utilizing of (Kashida) to hide more information in Arabic text cover text. In this method, some algorithms have been designed and implemented in a system called MSCUKAT (Maximizing Steganography Capacity Utilizing Kashida in Arabic Text).

In 2011, Nuur A. et al. [13] has proposed a new method that hide the secret bits in the sharp-edges for each character in the Arabic text document (cover media).

In 2014, Estabraq A.K. [14] has proposed a method that combine the Elliptic Curve (EC) arithmetic and metaheuristic algorithms with steganography techniques in order to increase the capabilities for steganography in diacritical Arabic text. The Estabraq method consists of three stages. Firstly compression stage to compress the secret messages into small codes based on B^+ tree indexing method, secondly cryptography stage to add the security by producing new algorithm to generate mask key based on EC algebra operations and metaheuristic algorithms, and thirdly is the steganography stage by embedding the encrypt secret message in diacritics Arabic text based steganography algorithm, this steganography algorithm is based on DNA coding for Arabic diacritics and Arabic grammar rule.

In 2015, Abdualraheem A.A.[15] utilize the NLP (for Arabic text) techniques as a tool in order to increment the efficiency of the steganography. Each sentence in the cover text will be parsed in order to get its hiding method, so more than one hiding method is used in one text, and therefore the system complexity is increased. This method depends on the grammar of the Arabic language to choose the method of hiding, i.e. each sentence must be parsed in order to obtain its grammar and according to this grammar the method of hiding will be chosen, so the security will be increased because multiple hiding methods will be utilized for one Arabic cover text. The B^+ Tree is utilized to index the grammar in the lexicon.

In 2016, Suhad M. , et al. [16] This paper propose a method that encrypt secret message and then embedding it in a cover text. This done by scrambled secret message first through AES encryption algorithm then hiding it into Arabic cover text. This paper also study the efficient embedding technique that can be used for hiding the encrypted data in cover text to hide it from attackers and sent message to the receiver in a safe mode.

4. The Proposed Text Steganography Method

In this paper, two ideas have been suggested, the first one is the modification of run length encoding (MRLE) by utilizing non printed characters that based on the grammar rule (that introduced by the parser), and the second one is the text steganography method that based on this suggested MRLE. The proposed method consists of two sides: The sender side and the receiver side, and in the following sections we will explain them in details with the required algorithms.

4.1 At The Sender Side

The proposed method at the sender side takes two texts as inputs: The secret text (that will be hidden) and the cover text (that will be embedded with the secret text) as shown in Figure-1.

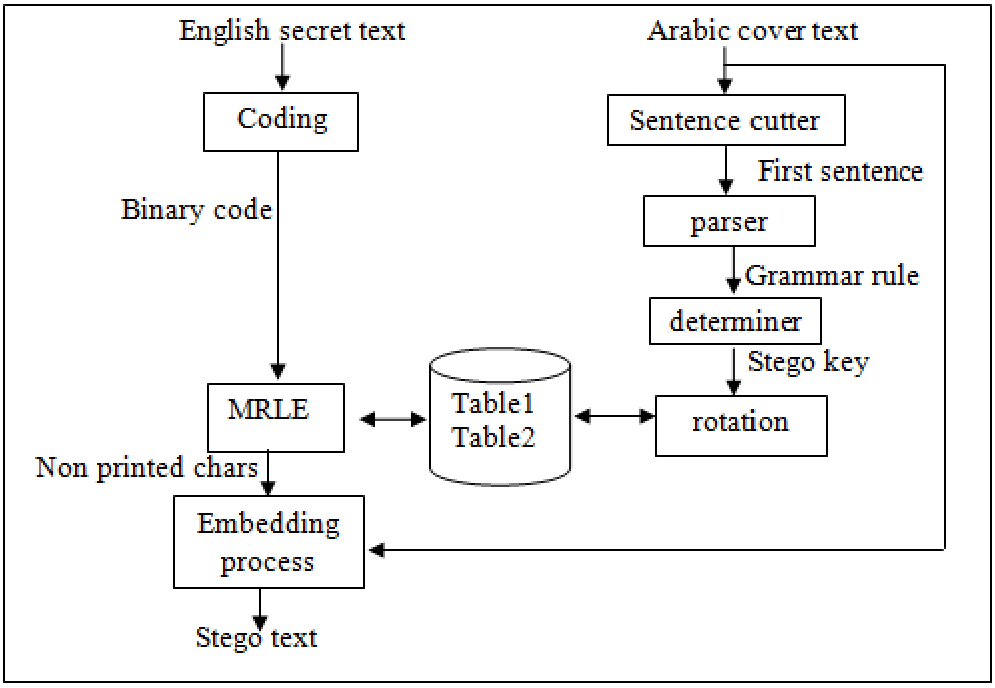

Figure 1- The proposed method at the sender side

At the beginning, the secret English text will be encoded into binary code according to the ASCII of cover characters, the output from this step (binary code) will be the input to the MRLE algorithm as shown in algorithm1.

The MRLE will do the processes of classical RLE at first, the output from this processes will be the counters (run length) of zeros or ones according to the input binary code, after that we will depend on two tables: Table-1 that represents the mapping between the zeros counter and ASCII code of non printed characters(see Table-1), and Table-2 that represents the mapping between the ones counter and ASCII code of non printed characters(see Table-2). Since there are two types of bits (zero and one) so the bit's type will determine which table of the two specified tables will be utilized in swapping the counters, for example, if the output from RLE is 30(which means three zeros) this mean that Table-1 will be used to make swapping and the counter (3) will be swapped with its corresponding ASCII of non printed character taken from this table, and the output will be 16, on the other side, when the receiver sees the ASCII 16 then he(or she) will understand that there are three zeros by depending on Table-1 since 16 is appears only on table1 that represents zeros counter and it corresponds to the counter 3.

In our proposed method we need to increase the security, so we need these Tables (Table-1 and Table-2) to be dynamic (in order to increase the probabilities), therefore we make a rotation for these two tables according to a stego key that also must be not static. In order to make the stego key dynamic (to increase the security level), we will use the grammar rule (that produced by an English parser) of the first sentence in the cover text (to be an indication between the sender and the receiver to determine the stego key). After that we make a rotation to Table-1 and Table-2 according to this stego key. These rotated tables will be utilized by our proposed MRLE method that take the binary code as input and produce a list of its corresponding ASCII code of non printed characters as shown in algorithm2.

```
Algorithm (1) :- The proposed  algorithm at the sender side
Input: English cover text (E), Secret text (T).
Output: Stego text (S).
Process:
Begin
Step1: Convert T to its equivalent binary code list (L) using its ASCII code.
Step2: Make rotation for table1 and table2.
        2.1 Cut the first sentence (FS) from E.
        2.4 Parse (FS) to get its grammar rule (G).
        2.5 determine the code (C) that is correspond to (G).
        2.6 Apply rotation to the table1 and table2 according to G.
Step 3: Call algorithm2 (MRLE) that takes (L) and produces list of ASCII code for non printed
        characters (NL) with based on the rotated table1 and the rotated table2.
Step4: Call algorithm3 that embed (NL) into the (E) to produce the stego text (S).
Step5: Return (S).
End.
```

Table 1- Zeros Counters Swapping

Counter	ASCII of Non Printed Character
0	12
1	14
2	15
3	16
4	17
5	18
6	19
7	20
8	21
9	25

Table 2- Ones Counter Swapping

Counter	ASCII of Non Printed Character
0	2
1	3
2	4
3	5
4	6
5	7
6	22
7	23
8	24
9	1

Algorithm (2): MRLE Algorithm

Input: Binary code list (L), table0 and table1 after rotation process.

Output: list of ASCII code of non printed characters (NL).

```
Process:
Begin
Step1:initialize variables
    1.1 N = length of L.
    1.2 i=1
Step2: while ( i<=N ) do
    Begin
    2.1 Count=1.
    2.2 j=i+1
    2.3 while (L[j]=L[i]) do
        begin
          2.3.1 Count=Count+1
          2.3.2 j=j+1
        end
    2.4. If L[i]=0 then
      2.4.1 for each digit(D) in Count do
      begin
            2.4.1.1 Get the corresponding value (C) to (D) from table(1).
            2.4.1.2 add C to NL
     End for
    Else
    2.4.2  for each digit(D) in Count do
    begin
            2.4.2.1 Get the corresponding value (C) to (D) from table(2).
            2.4.2.2 add C to NL
    End for
2.5 i=j
End /*while end */
 Step3: Return (NL).
End.
```

The last process of the proposed method will be the embedding process, that will take the output from previous MRLE (non printed characters) and English cover text as input and produce the stego text. Because of these non printed characters when embedded will be written as spaces so this process will utilize the spaces between English words of cover text for embedding, and if the spaces between words of cover text are not enough for embedding all non printed characters then we will utilize the spaces that found after the cover text as shown in algorithm3.

Algorithm (3) : *The embedding algorithm*

Input: list of ASCII code of non printed characters (NL), English cover text (E).

Output: Stego text (S).

```
Process:
Begin
Step1:  initialize variables
N=length of NL.
i=1.
S="".
Step2: while (E !="") do
Begin
2.1 Read character (C) from E
2.2. If C != space then
    2.2.1 Add C to S.
  Else
   2.2.2 If  i<=N then
        begin
        2.2.2.1 Add NL[i] to  S.
       2.2.2.2 i=i+1.
      End if
     Else
       2.2.2.3 Add Space to S.
End while
Step3: While i<= N do
Begin
        3.1 Add NL[i] to  S.
       3.2 i=i+1.
End while
Step4: Return (S).
End.
```

4.2 At The Receiver Side

The proposed method at the receiver side will take the stego text as input and produce the secret text as output. At the beginning the non printed characters are extracted from the stego text, and this process is the opposite of embedding process.

In our proposed method, the stego text is the same as cover text (because we embed the secret information using non printed characters) so in order to get the stego key we will do the same steps that we described it in the sending side (see algorithm4), we will take the first sentence of stego text and parse it to get its grammar rule, this grammar rule will be used to determine the stego key. This stgo key will be used to apply rotation to the Table-1 and Table-2. After that the modified run length decoding method (MRLD) is applied to return a list of binary code (that correspond to these non printed characters) based on these two rotated tables as shown in algorithm5, Then this binary code is decoded, such that each 8 bits of the binary code are returned to its corresponding ASCII code and then converted to its corresponding characters to compose the original secret text.

Algorithm (4) :- The extracting algorithm

Input: Stego text (S).

Output: Secret text (T).

```
Process:
Begin
Step1: Extract all the non printed characters from (S) to get NL.
Step2: Make rotation for table1 and table2.
      2.1 Cut the first sentence (FS) from S.
      2.4 Parse (FS) to get its grammar rule (G).
```

```
        2.5 determine the code (C) that is correspond to (G).
        2.6 Apply rotation to the table1 and table2 according to G.
Step 3: Call algorithm5 (MRLD) that takes (NL) and produces binary code (L) with based on the
        rotated table1 and the rotated table2.
Step4: Convert the L to its corresponding characters using its ASCII to produce S.
Step5: Return (S).
End.
```

Algorithm (5) :- MLRD algorithm

Input:: List of ASCII for non printed characters (NL), rotated table1 and rotated table2.

Output: binary code list (L).

```
Process:
Begin
Step1: initialize variables
        1.1 L=[]
        1.2 i=1
        1.3 N=length of NL
Step 2: while i<=N do
begin
          2.1 j=i
          2.2 Count=0
          2.3 Order=1
          2.4 While NL[j] is found in rotated table(1) do
              Begin
                     2.4.1 Get the corresponding value (C) of NL[j] from rotated table1.
                     2.4.2 Count=Count+C*Order.    /*Order is used if the counter is composed of
                                                      more than one digit*/
                     2.4.3 j=j+1.
                     2.4.4 Order=Order*10.
              End while
          2.5 if Count >=1 then
              begin
                     2.5.1 Add zeros to L according to Count value.
                     2.5.2 Count=0.
                     2.5.3 Order=1.
              End if
          2.6 While NL[j] is found in rotated table(2) do
              begin
                     2.6.1 Get the corresponding value (C)of NL[j] from rotated table2.
                     2.6.2 Count=Count+C*Order.
                     2.6.3 j=j+1
                     2.6.4 Order=Order*10
              End while
          2.7 if Count >1 then
                     2.7.1 Add ones to L according to Count value.
                     2.7.2 i=j
End While
  Step3:-Return (L).
End.
```

5. Implementation of the proposed method

Consider the following secret text is**:** Branch of Software
Suppose the cover text is: Computer Science
The ASCII code of secret text= [66 114 97 110 99 104 32 111 102 32 115 111 102 116 119 97 114 101]
The binary code of ASCII =[1000010 1110010 1100001 1101110 1100011 1101000 0100000 1101111 1100110 0100000 1110011 1101111 1100110 1110100 1110111 1100001 1110010 1100101].
The grammar rule for the first sentence in cover = "Noun + Noun" (that is produced by the parser)
The code that correspond to this grammar = 1
The stego key=1.
Table-1 and Table-2 are rotated one time (according to the stego key).
Applying classical RLE=
[11**4**0**1**1**1**0**3**1**2**0**1**1**1**0**2**1**4**0**3**1**1**0**3**1**1**0**2**1**3**0**4**1**1**0**1**1**4**0**1**1**5**0**2**1**1**0**6**1**2**0**2**1**2**0**1**1**5**0**3**1**2**0**4**1**1**0**6**1**2**0**2**1**1**0**3**1**1**0**1**
1**2**0**3**1**1**0**5**1**4**0**4**1**2**0**1**1**1**0**2**1**2**0**1**1**1**0**1**1**]**
Applying the MRLE algorithm =
[3,17,3,14,5,15,3,14,4,17,5,14,5,14,4,16,6,14,3,17,3,18,4,14,22,15,4,15,3,18,5,15,6,14,22,15,4,14,5,14,3,155,14,7,17,6,15,3,14,4,15,3,14,3]
Embedding process= Computer3Science17 3 14 5 15 3 14 4 17 5 14 5 14 4 16 6 14 3 17 3 18 4 14 22 15 4 15 3 18 5 15 6 14 22 15 4 14 5 14 3 55 14 7 17 6 15 3 14 4 15 3 14 3
(since these non printed characters are not shown when written, then stego text will be equal to cover text)
Stego text=Computer Science

6. Experimental result

There are three main aspects that should be taken into account when testing the results of text steganography: Security, capacity and robustness, the following sections will illustrates the results of each aspect for the proposed approach.

6.1 Measure of Capacity

Damerau-Levenshtein distance calculates the minimum number of operations necessary to convert a string to another, where a transaction is defined as the insertion, deletion, or substitution of a single character, or as a transposition of two characters[11].In the proposed method the Damerau-Levenshtein distance is (0) since there is no need for any operation to apply on the cover text to convert to the stego text.

The capacity ratio is computed by the following equation[8]:

Capacity ratio = (amount of hidden text in bytes) / (size of the cover text in bytes) (1)

Table-3 illustrates the results of capacity ratio for the proposed text steganography approach with different cover text file sizes that was tested according to equation (2). Table-4 illustrates capacity ratio of other previous approaches . Table-3 clarifies the robust of this proposed method by providing high capacity ratio when compared with the capacity of other methods of text steganography that shown in Table-4. The proposed approach achieved a high capacity ratio for Steganography that reached to five times more than the cover text length.

Table 3- Capacity ratio of the proposed approach

Secret text size(in byte)	Real size used of cover text (in byte)	Capacity ratio of hiding
15	12	125%
37	12	308.3%
50	12	416.66%
55	12	458.3%
58	12	483.3%
434	88	493%
1095	502	218%
2506	502	499%

Table 4- Capacity ratio of other approaches

Approach	Hiding Capacity Ratio
Shirali-Shaherza [17]	74.32%
Gutub and Fattani [18]	33.68%
Estabraq A. K.[14]	106.4 %
Abdualraheem A.A.[5]	150 %

6.2 measure of Perceptual Transparency (Security)

There are two measures that are used in text steganography field to check the security: Jaro Winkler distance and Damerau Levenshtein distance. The higher Jaro Winkler distance between two strings means they are more similar. The result is normalized to have a measurement between 0 and 1, zero representing the absence of similarity [10]. The Jaro distance is computed using the following equation:

$$\mathbf{d_j=1/3\ (\ m/|S1| + m/|S2| + m\text{-}t\ /m)} \qquad \mathbf{(2)}$$

Where:
|Si|: String length
m: Number of matched characters
t: Number of transpositions

In the proposed method the Jaro-Winkler distance is equal to (1.0) which means that the stego text and the cover text are identical which not give the chance to the attacker to has doubts that this stego text contains secret information. Table-5 illustrate a the results of Jaro similarity ratio for the proposed steganography method that tested according to equation (2) and the ratio of other previous methods.

Table 5- Result of Jaro similarity ratio

Approach	Jaro Similarity Score
Estabraq A. K.[14]	0.90
Abdualraheem A.A.[5]	0.937
The Proposed Method	1.0

7. Conclusions

In this paper the following points can be concluded:

1) Making the stego key dynamic (not static) by utilizing the grammar rule of the first sentence as an indication between the sender and receiver will provide a good security level.

2) Making the tables of data (that MRLE depend on) dynamic by rotating them according to a dynamic stego key will provide a good security level.

3) Using Non Printed characters that don’t appear on screen provide us a good hiding tool that can be used for steganography purpose, since the similarity between cover text and stego text is 1.0.

4) Since we utilize the spaces between (and after) the words of cover text, we achieved a high capacity ratio for Steganography that reached to five times more than the cover text length.

5) The proposed Modified Run Length Encoding method (MRLE) that depends on the Non Printed Characters will provides a good compression ratio that can be used for steganography purposes, since we embed a block of data each time rather than single bits.

6) The Proposed steganography method that depends on modified run length encoding method provides a complete similarity with high capacity(more than five times the cover text length).

7) RLE is not useful with un identical data (since the data will be expanded rather than compressed), while MRLE solve the problem, so even un identical data is not compressed but it is not expanded also (since we don’t store the bit type).

References:

1. Pramatha N, Tanmay B. **2010**. On Embedding of Text in Audio – A case of Steganography, In: International Conference on Recent Trends in Information, Telecommunication and Computing; 12-13 March; Kochi, Kerala, India: IEEE. pp:203 - 206.

2. Kefa R. **2004**. Steganography-The Art of Hiding Data, *Information Technology Journal,* 3(3), pp:245-269.

3. Manish M, Navdeep K. **2012**. Adaptive Steganography: A survey of Recent Statistical Aware Steganography Techniques, *International Journal of Computer Network and Information Security (IJCNIS)*. 4(10), pp: 76-92.

4. Samira Mersal, Safiah Alhazmi, Razan Alamoudi and Noura Almuzaini**. 2014**. Arabic Text Steganography in Smartphone*, International Journal of Computer and Information Technology*, 03(02).

5. Abdulraheem A.A.**2014**. Information Hiding In Arabic Text Using Natural Language Processing Techniques, M.Sc. Thesis, Department of Computer Sciences of University of Technology, Baghdad, Iraq.

6. M.P.Bhuyan, V.Deka , S.Bordoloi and Burrows Wheeler. **2013.** Data Compression and Secure Transmission, Gauhati University.

7. Allen Wyatt. **2015**. Displaying Non Printing Characters, Available at: http://wordribbon.tips.net/ t008879-DispalayingNonPrintingcharacters.html

8. Benett K. **2004**. Linguistic Steganography-Survey, Analysis and Robustness Concerns for Hiding Information in Text, Purdue University, CERIAS Tech. Report 13.

9. Shahreza M. S., and Shahreza M. H. S.**2007**.Text Steganography in SMS, International Conference on Convergence Information Technology.

10. Bender W., Gruhl D., Morimoto N., and A. Lu.**1996**. Techniques for Data Hiding, *IBM Systems Journal,* 35.

11. Por L. Y., Delina B. **2008**. Information Hiding: A New Approach in Text Steganography, 7th WSEAS int. Conf. on Applied Computer & Applied Computational Science (ACACOS '08), Hangzhou, China, April 6-8.

12. Adnan Abdul-Aziz Gutub , and Ahmed Ali Al-Nazer. **2010**. High Capacity Steganography Tool for Arabic Text Using 'Kashida, *The ISC Int'l Journal of Information Security*.

13. Nuur Alifah Roslan, Ramlan Mahmod and Nur Izura Udzir. **2011**. Sharp-Edges Method In Arabic Text Steganography, *Journal of Theoretical and Applied Information Technology*.

14. Estabraq A. K. **2014**. Improving of Information Hiding Using Artificial Intelligent Techniques, M.Sc. Thesis, Department of Computer Sciences of University of Technology, Baghdad ,Iraq.

15. Abdulraheem A.A.**2014**. Information Hiding In Arabic Text Using Natural Language Processing Techniques, M.Sc. Thesis, Department of Computer Sciences of University of Technology, Baghdad, Iraq.

16. Suhad Malalla and Farah R. Shareef. **2016**. Improving Hiding Security of Arabic Text Steganography by Hybrid AES Cryptography and Text Steganography *International Journal of Engineering Research and Applications (IJERA)*, 6(6), ISSN: 2248-9622.

17. M. Hassan Shirali-Shahreza and Mohammad Shirali- Shahreza. **2006**. A New Approach to Persian/Arabic Text Steganography, 5th IEEE/ACIS International Conference on Computer and Information Science (ICISCOMSAR06).

18. Adnan Gutub and Manal Fattani. **2007**. A Novel Arabic Text Steganography Method Using Letter Points and Extension, WASET International Conference on Computer, Information and Systems Science and Engineering (ICCISSE), Vienna, Austria.