Journal of Theoretical and Applied Information Technology
15th November 2011. Vol. 33 No.1

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

SHARP-EDGES METHOD IN ARABIC TEXT STEGANOGRAPHY

[1]NUUR ALIFAH ROSLAN, [2]RAMLAN MAHMOD, NUR IZURA UDZIR[3]

[1] Department of Multimedia, FSKTM, UPM, Malaysia-43400

[2]Prof, Department of Multimedia, FSKTM, UPM, Malaysia-43400

[3]Dr, Department of Multimedia, FSKTM, UPM, Malaysia-43400

E-mail: nuuralifahroslan@gmail.com, ramlan@fsktm.upm.edu.my, izura@fsktm.upm.edu.my

ABSTRACT

One of the issues that arise in the text steganography is the capacity of hiding secret bit. Focusing in Arabic text steganography we propose a sharp-edges method to encounter the issue. This new method will hide the secret bits in the sharp-edges for each character in the Arabic text document. The main processes involved are identifying sharp-edges in the cover-text, secret message preparation to be hidden as a binary string and lastly, the bit hiding process. The experiments show that the capacity percentage used to hide the secret bit was increased up to 37.8%, resolving the capacity issue. The stego-text for this method has high invisibility, and therefore it is possible to be published publicly. We introduce keys to determine the possition of the secret bit randomly. This method utilizes the advantages of the Arabic text for steganography and can also be implemented to the same text scripting of languages such as Jawi, Persian or Urdu.

Keywords: *Steganography, Text Steganograpy, Arabic Text Steganography, Arabic Character, Sharp-Edges.*

1. INTRODUCTION

Steganography is a means of communication intended to make the very presence of the message undetectable. The word steganography literary means "covered writing", define from the root word of the ancient Greek *Steganos Graphos.* This will enable two parties to have a secret communication and it was classified under the information hiding field [1].

Digital form of media as a cover-object being use in steganography are pictures, video clips, music and sounds [2]. Text steganography also have been used since 2000 bc as a cover media. Nowdays, the texts have been moderate into the digital form whereas the steganography was also implemented in the digital text form.

Text steganography is the most difficult kind of steganography [1], due largely to the relative lack of redundant information in a text file as compared to picture or sound [3].

Moreover, the grammatical and orthographic characteristics of every language are different, text stegonographic schemes must be specifically

designed to exploit the specific characteristics of the target language. Recently there have been several successful attempts to design text steganographic schemes for English, Japanese, Korean, Chinese, Thailand, Persian, and Arabic [4].

Through observation, image steganography have more advantages than text steganography because of many redundant bits in an image. Therefore, large capacity of secret information can be hidden in the image. Brassil [5] makes an expression that text steganography is the most difficult kind of steganography and this statement was supported by Bender [2] that it is due largely to the relative lack of redundant information in a text file as compared with a picture or a sound file.

In Arabic Script, several researches have been done, e.g. the dot steganography [13], La steganography [14], pointed letters with extension

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

(Kashida in Arabic) [16], diacritics Arabic text steganography [15] and pseudo-space and pseudo-connection [17]. Most of these researches focus in increasing the capacity for hidden bits.
However, most of them have their disadvantages, e.g. diacritics methods [15] is low invisibility and can raise suspicious to the readers and this methods also needs a fully diacritical texts, but most of Arabic texts have no diacritics. Shirali-Shahreza's [17] pseudo-space and pseudo connection methods rely on the zero width non joiners (ZWNJ) and zero width joiner (ZWJ). Once the Unicode of ZWNJ and ZWJ are detected and removed it will affect the secret information.

We exploit other advantages of Arabic characters, i.e. the sharp-edges as this method provides a large capacity to hide the secret information with high level of invisibility. Furthermore, there is a security layer being introduced to encounter the previous problems.

2. RELATED WORKS

The main issue in text steganography is the redundancy of data. However, the exploits of text orthographic characteristics of every language is different. Recently, in text steganography field it was specifically designed to exploit the specific characteristic of the target language.

There have been several successful attempts to design text steganography based on the characteristic of their features, for example in these languages; English [5]-[8], Japanese [9], Korean [10], Chinese [11]-[12], Arabic [13].

Our method focuses on the Arabic script which, from our observation, has many benefits that can be utilized from this text scripting.

Unlike Latin alphabet, Arabic text scripting does not differentiate between the upper and lower case or between written or printed letters [3]. There are many beneficial features of Arabic character set that have been exploited for text steganography. The Arabic text steganography is in their beginning phase, where the proposed method related with dots and connectivity.

Arabic and English languages have dots in the character set, but the Arabic has dots in 15 letters out of 28 character set, while the Latin character has dots in only two letters (i and j) [13]. Table 1 classifies the dots (pointed) and un-dots (un-pointed) character set of Arabic letters.

The Shirali-Shahreza's [13] methods hide the secret information in the points (dots) location within the pointed (dots) letters. The length of the secret information is identified (example 20 bits) and compress. The cover medium text is scanned line by line, character by character. Whenever a pointed (dots) letter is detected, its points (dots) location may be affected by the hidden information bit. If the hidden bit is one, the point (dot) is slightly shifted up; otherwise the concerned cover-text character point location remains unchanged.

Figure 1 shows the point (dot) shifting for the Arabic letter "Fa". This method has the advantage in capacity and secrecy. The other remaining characters are also changed randomly to create more secrecy for this method. Nevertheless, the main weakness for this method is the lack of robustness. The hidden information is lost in any retyping or scanning. Moreover, the output text is fixed for the use of only one font [13].

The Kashidah method proposed by Adnan [15] exploits the existence of the redundant Arabic extension character (Kashidah) and the pointed (dots) letters. This method is more practical: the pointed letters with a Kashidah will hold the secret bit "1" and the unpointed (without dots) letters with a Kashidah to hold "0". The character extension has a standard character hexadecimal code of 0640 in Unicode system and this method does not have any effect to the writing content. Table 2 shows an example of this method.

The La Steganography method [14] uses a special form of La word (a combination of the Lam and Alef caharacters) for hiding the data, i.e. by inserting an Arabic extension character between the Lam and Alef. For hiding bit 0, they use the normal form of La, whereas bit 1 is hidden using the special word La with a unique code in the Standard Unicode (i.e. FEFB in the Unicode hex notation).

This method is not limited to electronic documents (e-documents) and can be used on printed documents. However, the La word use is limited in the Arabic sentences and this affect the capacity to hide and the full use of La word will raise up stego-text the suspicions.

The pseudo-space and pseudo-connection character method [17] hides one bit in each letter: the method

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

will look whether the letter of a word is connected to the next letter. If it is, they insert a ZWJ letter between the two letters to hide bit 1 and do not add anything for hiding bit 0.

Because the letters are connected together, adding ZWJ for connecting the letters together does not have any effect on the appearance of the text. The drawback of this method is that removing the Unicode of ZWJ will also affect the secret information.

The next method presents the Arabic diacritic [16]. There are eight diacritics in Arabic text. In this method, the cover-text is assumed to be a fully diacritics text. The bit '1' is kept in the diacritics and for the bit '0' is kept in the non-diacritic and the other diacritics are unused.

This method has high capacity but low invisibility as it attracts the attention of the reader. Furthermore, this method also needs fully diacritical text, but most of Arabic text has no diacritic.

Through these review, we found the uniqueness of the Arabic characters is that they have many sharp-edges as illustrated in Figure 2 and can be categorizing as in the Table 3. From here we propose a high level of capacity, with high invisibility for bit hiding. We introduce the keys in this method for positioning the secret bit.

The different number of sharp-edges show the possibility to hide the secret bit 1 and 0. The character with the number of sharp-edges 1 is possible to hide the secret bit in two possible conditions, which is either hiding secret bit 0 or 1at the sharp-edges position. Meanwhile, if the number of sharp-edges is 2, the possibility for the secret bit position is in four possible conditions; 11, 10, 00 or 01.

4. METHODOLOGY

The Sharp-Edges Stego's main architecture consists of two main modules, the hiding module and retrieving module. Figure 3 schematically illustrates the main flow of this method.

The hiding module is used by the sender of the hidden secret message, while the retrieving module is used by the receiver to extract the secret message from the stego-text. The input is determined by the sender (i.e. key 1) and the file involved is the secret message and the cover-text file. The hiding module requires a stego-key to randomly position the secret bit. This process generates a stego-text which can be publicly distributed without raising suspicions.
The retrieving module requires the same stego-key as the sender. Once the key is recognized by the method, the method will retrieve the secret message based on the hard-coded reference table. The extracted secret message is the final output from this method. The hiding module is the most important module as it is related with the hiding place of the secret bit. It has two main methods, Odd and Even methods, the use of which will be determined by the input from the sender. Flow chart in Figure 5 shows how the method flows.

Assuming we have a secret bit (110010) and we have a cover text which in Arabic text document with UTF-8 encoding. Each character will be recognized as an isolated character. The example is as in Table 4. The sender then creates a key (key 1, i.e.: 18990) which the sum of will be calculated, and if the result is an even number it will proceed to the even method of hiding, otherwise the hiding process will proceed with the odd method of hiding.

The even method of hiding will calculate the number of sharp-edges for Arabic characters with dots, while the odd method will calculate the number of sharp-edges for characters without dots.

For our example above, the sum value of the input is:

1+8+9+9+0 = 27

Therefore, the input will proceed to the odd hiding method where the focus place for hiding the secret bit is at the character without dots (i.e. ر, ا, ر, ا, ل, و, ر, ا, ر, ل). The total of the sharp-edges is then calculated as 19. The total number of the sharp-edges represents the number of places to hide the secret bit.

Next, a reference number which is also used as a key 2 to retrieve back the secret bit is generated based on the reference table as shown in Table 5. Through the reference table, the position of the secret bit is determined. Once the hiding process is complete, we will receive a stego-text and key 2 which is an important input in the retrieving module and Figure 6 shows the retrieving module flow chart. This is how our method runs and the following section presents our experimental results.

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

5. EXPERIMENTAL RESULT

We compare our method against the previous Shirali-Shareza's [17] methods. These resources are selected for computing the capacity of the methods for hiding the data. The Internet address of these newspapers and the capacity of each text for hiding data are shown in Table 6.

Two measurements were used in conducting this experiment, the capacity and the invisibility measurement. The capacity ratio is calculated based on the equation proposed by Shirali-Shahreza [13]. As our method has *t*, sharp-edges for hiding each secret bit in *c* cover-text file (kilobyte), therefore the hiding capacity of our method as bit/kilobyte is:

$$\textit{Hiding Capacity} = \frac{t \ \ \textit{(indicate each bits)}}{c \ \ \textit{(kilobyte)}}$$

Through the experiments which is involved the even and odd methods, our method shows a higher capacity in hiding secret bits compared to Shirali-Shahreza's, as shown in Figure 4(a) and Figure 4(b).

For the even method, the sharp-edges method has 27% more capacity than the Pseudo Steganography, 80% higher capacity than Dot Steganography and up to 99% more capacity as compared to La Steganography.

Higher capacity is also provided by the odd method, with the capacity of 45% higher than Pseudo Steganography, 85% more than Dot Steganography and a huge difference of up to 100% increase in capacity than La Steganogrraphy.

The numbers of sharp-edges present the number of places to hide the secret bit. The large hiding capacity provided by our method is due to Arabic characters having one to five sharp-edges.

We also propose keys implementation: Key 1 is the input from the user which will determine either the odd or even method; Key 2 is generated from the reference table which specified the position of the secret bit in order to retrieve back the secret information.

The second measurement, invisibility refers to unsuspicious look of the stego-text [19]. Different sizes for two files with identical contents may raise suspicions, and this can be considered as low invisibility. Invisibility is measured by calculating the differences between the cover-text file size and the stego-text file size. Ratio 0 reflects high invisibility; and the higher the ratio, the lower invisibility.

Through this invisibility experiment, both sub-methods gave the same results as shown in Figure 7. The stego-text is in the highest invisibility level which is 1.02 kilo bytes.

6. CONCLUSION

There are three important parameters in designing steganography methods: perceptual transparency, robustness and hiding capacity. These requirements are known as the "magic triangle" and are contradictory [18].

Sharp-Edges method resolves the capacity issue which is always a concern in text steganography. Through comparison with the other methods, our method still provides the highest capacity of hiding even though not all the characters in the cover-text are fully used.

This result is based on the even and odd method hiding module. Through this module only the selected character will be chosen to hide the secret bit based on the given input. However, the capacity can increase more by using all the cover-text characters.

Through our findings, the strength of this method is in the reference table, since the secret bit position is determined by this table. Therefore, a high security layer is needed to ensure that the table is secure from eavesdroppers. The implementation of the keys provides one of the contributions to this research where a random position for hiding in high capacity is introduced. Furthermore, the stego-text has high invisibility which enables it to be published to the public.

Finally, through our research we briefly examined some directions in future that can be pursued based on what we have done so far. The extensions for this work can include the combination with cryptography layer, randomizing the location of the secret bits and the implementation of other character scripts.

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

REFERENCES

[1] N. F. Johnson, Z. Duric and S. Jajodia, *Information hiding: Steganography and watermarking-Attacks and countermeasures*, Kluwer Academic Publishers, Boston, 2000.

[2] J.T. Brassil, S. Low, N.F. Maxemchuk, and L.O'Gorman, "Electronic Marking and Identification Techniques to Discourage Document Copying", *IEEE Journal on Selected Areas in Communications*, vol. 13, Issue. 8, October 1995, pp. 1495-1504.

[3] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding", *IBM Systems Journal*, vol. 35, Issues 3&4, 1996, pp. 313-336.

[4] Natthawut Samphaiboon and Matthew N. Dailey, "Steganography in Thai Text, *IEEE Journal on Selected Areas in Electrical Engineering/Electronic, Computer, Telecommunications and Information Technology,* vol. 1, 2008, pp. 133-136

[5] J. T. Brassil, S. Low, and N. F. Maxemchuk, "Copyright protection for the electronic distribution of text documents," *Proceedings of the IEEE,*vol. 87, no. 7, pp. 1181–1196, July 1999.

[6] D. Huang and H. Yan, "Interword distance changes represented by sine waves for watermarking text images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 12, pp. 1237–1245, December 2001.

[7] Y.-W. Kim, K.-A. Moon, and I.-S. Oh, "A text watermarking algorithm based on word classification and inter-word space statistics," *Proceedings of the Seventh International Conference on Document Analysis and* Recognition, pp. 775–779, October 2003.

[8] M. Topkara, C. M. Taskiran, and E. J. Delp, "Natural language watermarking," in *Proceedings of SPIE-IS & T Electronic Imaging*, 2005, San Jose, USA, January 2005.

[9] T. Amano and D. Misaki, "A feature calibration method for watermarking of document images," in Proceedings of the Fifth International Conference on Document Analysis and Recognition ICDAR'99, September 1999, pp. 91–94.

[10] Y.-W. Kim and I.-S. Oh, "Watermarking Text document images using edge direction histograms,"Pattern Recognition Letters, vol. 25, no. 11, pp. 1243–1251, August 2004.

[11] X. Sun, G. Luo, and H. Huang, "Component-based digital watermarking of Chinese texts,"in Proceedings of the 3rd International Conference on information security, Shanghai, China, November 2004, pp. 76–81.

[12] W. Zhang, Z. Zeng, G. Pu, and H. Zhu, "Chinese text watermarking based on occlusive components," The 2nd Information and communication Texhnology ICTTA'06, vol. 1, pp. 1850–1854, April 2006.

[13] M. Shirali- Shahreza,"A New Approach to Persian/Arabic Text Steganography", *Proceedings of the 5th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2006)*,Honolulu, HI, USA, July 10-12 2006, pp. 310-315.

[14] M. Shirali-Shahreza, "A New Persian/Arabic Text Steganography Using "La" Word", *Proceedings of the International Joint Conference on Computer, Information, and Systems Sciences, and Engineering (CISSE2007)*, Bridgeport, CT, USA, 2007.

[15] A. Gutub and M. Fattani, "A Novel Arabic Text Steganography Method Using Letter Points and Extensions", *Proceedings of the WASET International Conference on Computer,Information and Systems Science and Engineering (ICCISSE)*, Vienna, Austria, Vol.21, 2007, pp. 28-31.

[16] Gutub, A.Am Ghouti, L.M,Elarian, Y.S., Awaideh, S.M., Alvi, A.K., "Utilizing diacritic marks for Arabic text steganography", *(2010)*, Kuwait Journal of Science and Engineering, 37 (1B), pp.89-109.

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

[17] M.H. Shirali-Shahreza and M. Shirali-Shahreza, "Steganography in Persian and Arabic Unicode Texts Using Pseudo-Space and Pseudo-Connection Characters," *Journal of Theoretical and Applied Information Technology (JATIT)*, Vol. 4, No. 8, August 2008, pp. 682-687.

[18] N. Cvejic, *Algorithms for Audio Watermarking and Steganography*, Oulu University Press, Finland, 2004.

[19] Cole, E.2003. Hiding the Goods with Steganography, Hiding in Plain Sight, Canada; Bob Ipsen, pp. 60-61, ISBN: 0-471-444449-9.

Journal of Theoretical and Applied Information Technology
15th November 2011. Vol. 33 No.1

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

Table 1: Arabic letters

un-pointed letters	pointed letters
ا ح د ر س ص ط ع ك ل م هـ و	ب ت ث ج خ ذ ز ش ض ظ غ ف ق ن ي

Table 2: Steganography example adding extension after letters

Secret bits	110010
Cover-text	من حسن اسلام المرء تركه مالا يعنيه
Steganographic text	من حسن اسلام المرء تركه مالا يعنيه 1 1 0 0 1 0

Table 3: Arabic letters with different number of sharp-edges

Number of Sharp-edge (s)	Characters
1	و ة ف ه م
2	ا ب ت ث د ذ ر ز ي ل ط ظ ن
3	غ ع ء ح ج خ
4	س ش
5	ك

Table 4: Isolated character for cover

Secret-bits	1 1 0 0 1 0
Cover-text	لا ضرار ولا ضرر
Isolate for each character	ل ض ر ا ر و ل ا ض ر ا ر

Table 5: References Table

Arabic Character	Name	Sharp -edges	Rep.Bit	Rep.num
و	wāw	1	0	09000
			1	09001
ا	alif	2	10	00010
			11	00011
			00	00012
			10	00013
ل	lām	2	10	60000
			11	60001
			00	60002
			01	60003
ض	□ād	2	10	00600
			11	00601
			00	00602
			01	00603
ر	Rā	2	10	00100
			11	00101
			00	00102
			01	00103

Table 6: Related website and the capacity of the required

Online Iranian Newspaper	Website Address	Text Size (Kilo Byte)
Farhange Ashti	http://www.ashtidaily.com	13.3
Hamshahri	http://www.hamshahri.com	6.82
Iran	http://www.iraninstitute.com	6.64
JameJam	http://www.jamejamdaily.net	3.84
Javan	http://www.javandaily.com	8.03
Khorasan	http://www.khorasannews.com	4.40
Keyhan	http://www.kayhannews.ir	2.92
Quds	http://www.qudsdaily.net	9.98

Figure 1: Point (dots) shift-up for Arabic letter “Fa”

Journal of Theoretical and Applied Information Technology
15th November 2011. Vol. 33 No.1

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

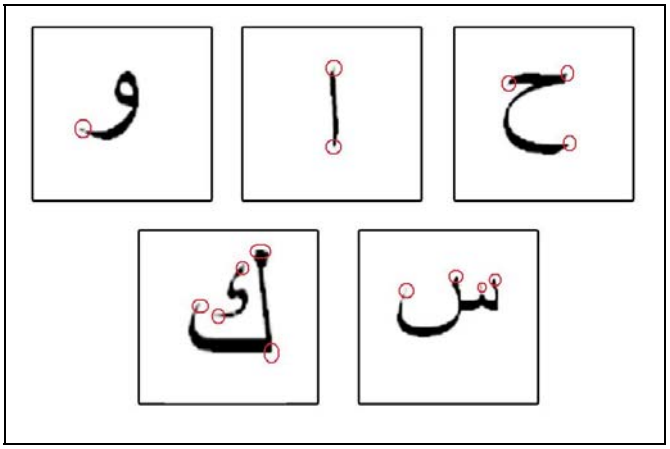

Figure 2: Arabic Character's Sharp-Edges

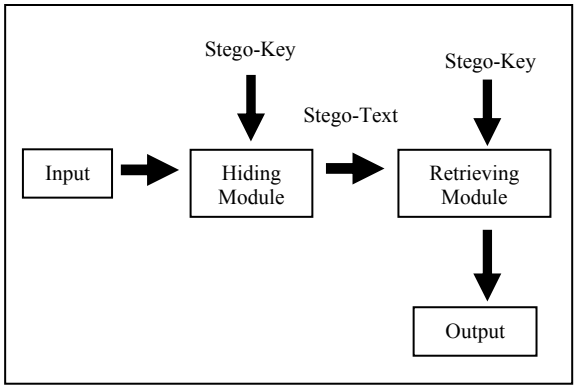

Figure 3: Architecture of Sharp-Edges Stego method

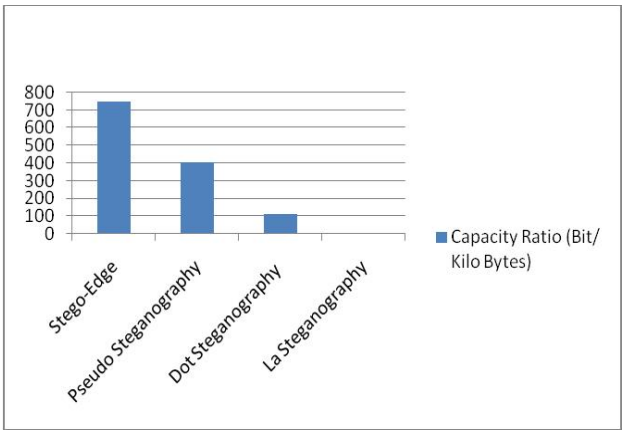

Figure 4 (a): Capacity comparison for Even Method

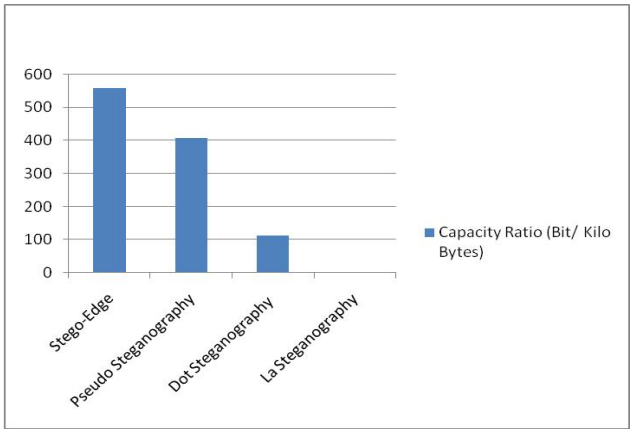

Figure 4 (b): Capacity comparison for Odd Method

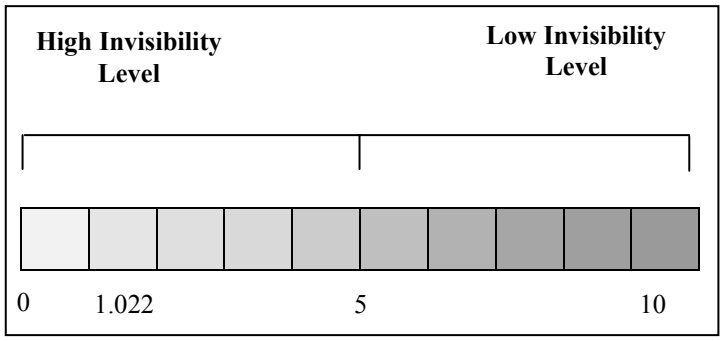

Figure 7: Invisibility Level Ratio

Journal of Theoretical and Applied Information Technology
15th November 2011. Vol. 33 No.1

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

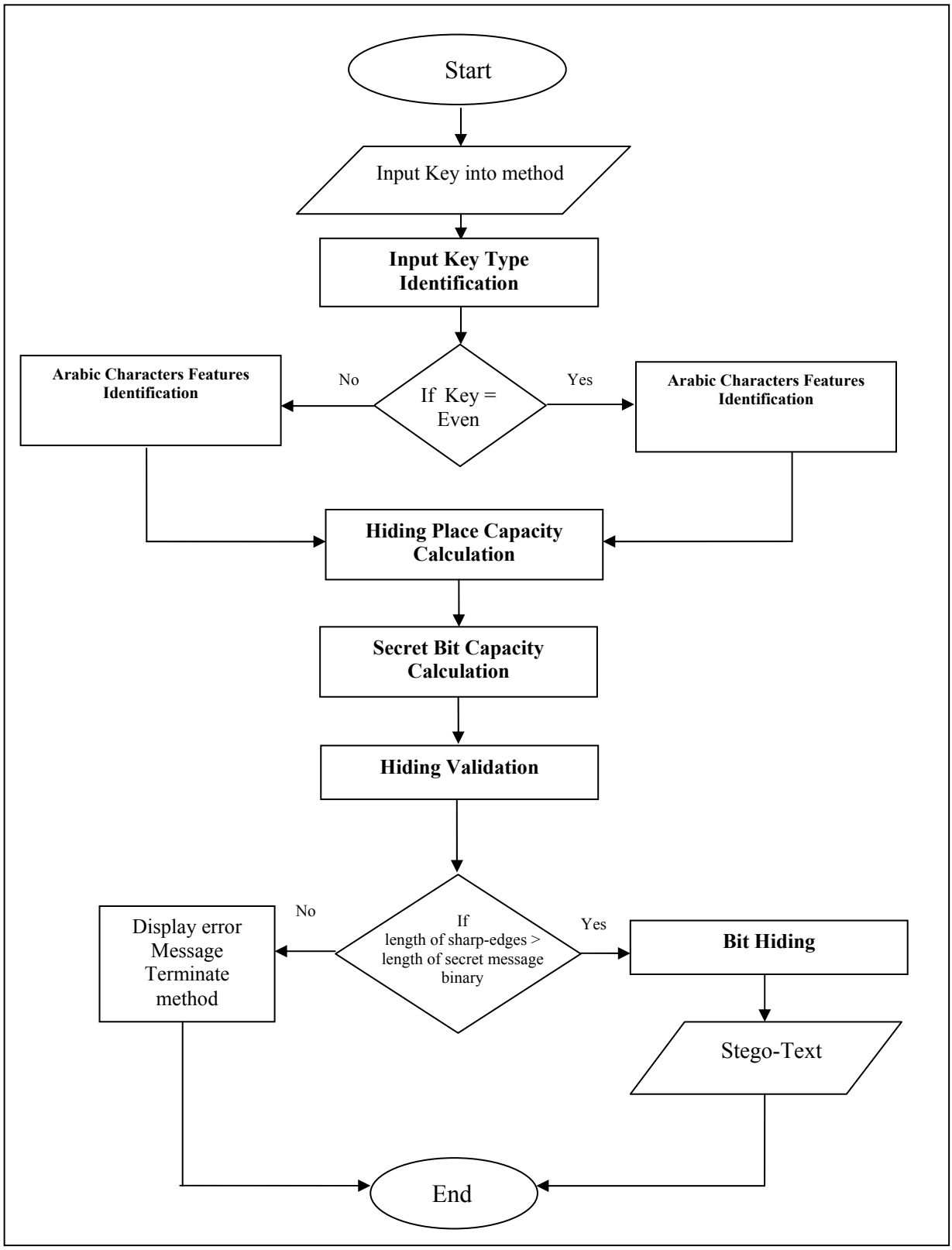

Figure 5: Hiding Module Flow Chart

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

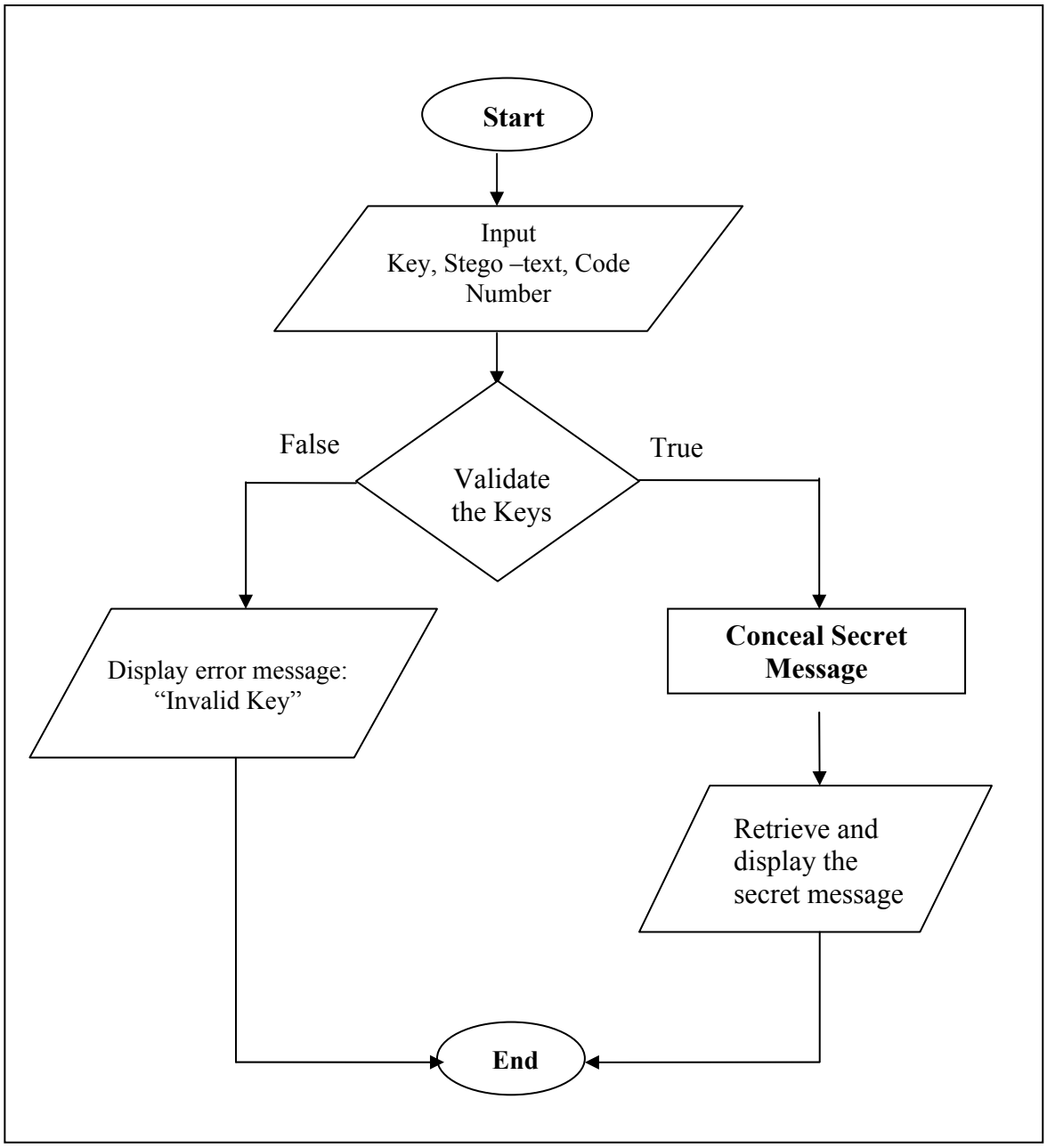

Figure 6: Retrieving Module Flow Chart