A MODEL DESIGN TO ENHANCE THE SECURITY OF SECRET MESSAGING USING FONT TYPE TEXT STEGANOGRAPHY

[1]Sumathy Kingslin, [2]N. Kavitha
[1]Associate Professor, [2]M.Phil Research Scholar,
PG and Research Department of Computer Science, Quaid-E-Millath College for women, Chennai – 600002, Tamil Nadu, India;

Abstract: ***Information that is passed via network can be easily hacked. Security becomes mandatory to the information passed through communication lines. Steganography makes the information to be passed with no suspicion. The proposed method for securing MS-word document using look-alike fonts is used to hide secret message in a cover text. Hexadecimal code combined with pattern table of look-alike fonts. By using look-alike fonts give similar between cover and stego text. Two methods have been taken to compare with proposed method. In proposed MS-word document using look-alike fonts the embedding capacity is high when compared to other two techniques because all alphabet letters are used to hide and its similarity measure is high. Criteria have been used to measure the goodness of the algorithm. Similarity measures and capacity ratio are also used to find wellness of the proposed method.***

Keywords: Network Security, Steganography, Text Steganography, Format Based Method, Look-Alike Fonts, Capacity Ratio, Four Criteria, Similarity Measure

1. INTRODUCTION

Nowadays data sharing through network is very common due to its convenience. Hacking and modifying the transferred data are commonly practiced in the communication lines. Security is made mandatory to data that are transferred through anonymous network. Steganography is the art and science of data hiding. Steganography makes the secret message to be hidden in cover media in such a way that no one suspects it during transmission. The main aim of steganography is to send secured data through a hidden cover. The cover medium may be an image, audio, video or text data. Steganography is derived from Greek word 'Steganous' meaning "covered" and 'graphy' meaning "writing". So it is known as "covered writing".

1.1 Types of Steganography

Steganography has four basic types based on the medium used for hiding the secret text. They are text, image, audio and video. The data can be hidden in any cover medium i.e., text, image, audio or video as shown in figure 1.

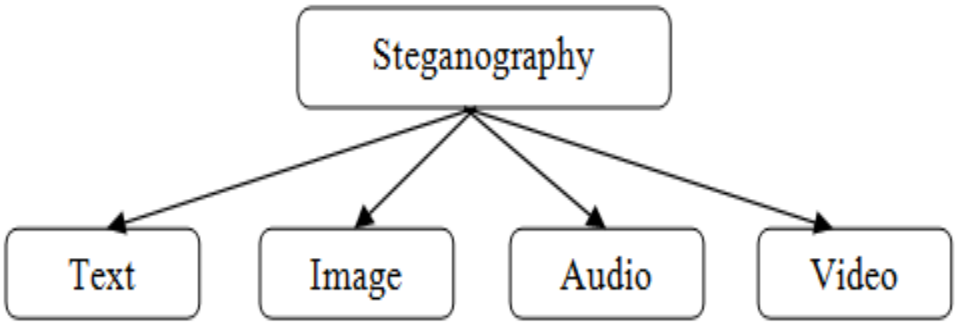

Figure 1: The Basic Types of Steganography

1.2 Text Steganography

When data is embedded or hidden inside a text as cover is called text steganography [1]. Text steganography is preferred over other media, because of lesser space occupied by the text, communicate more information and need less cost for printing as well as some other advantages [2]. Each steganography communication system consists of an embedding algorithm and an extraction algorithm. The secret message embedded in cover text using embedding algorithm [3]. Hiding information may require a steganographic key which is additional secret information, such as a password, required for embedding the information [4]. The embedding algorithm then produces a stego text that can be stored and/or transferred through communication channels. The extracting algorithm receives the stego text and the (optional) stego-key, and extracts the secret message as shown in figure 2[5].

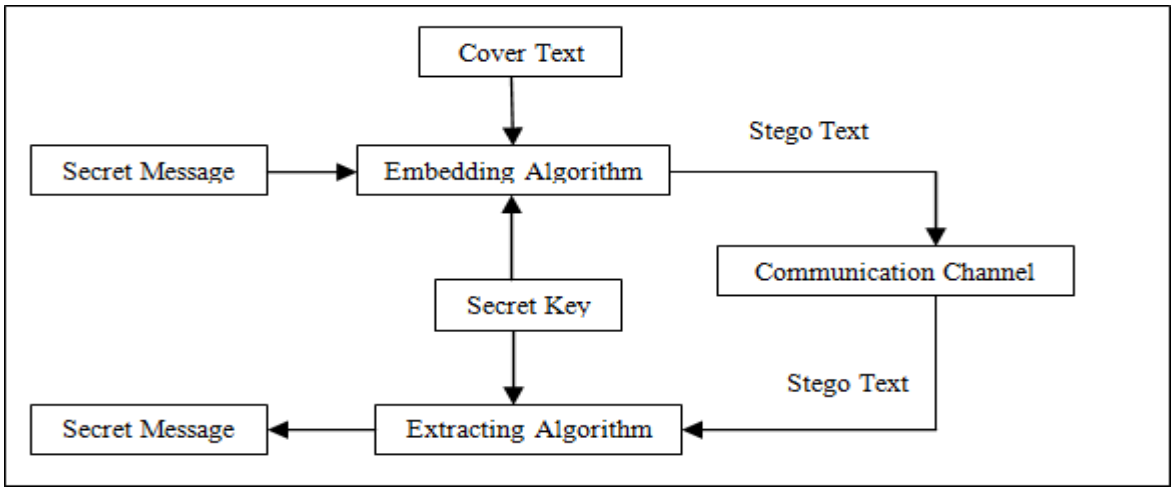

Figure 2: The Main Steps of Data Hiding and Extracting [5]

There are different ways by which secret data can be embedded in text steganography:

- Format Based Method
- Random and Statistical Method
- Linguistic Method.

1.2.1 Format-Based Method

It modifies the existing cover text in order to hide the secret text. It involves the insertion of spaces, resizing the text and changing the style of text to hide the secret message [6].

1.2.2 Random and Statistical Method

Random method hides the characters that appear in random sequence. Statistical methods determine the statistics such as means, variance and chi square test which can measure the amount of redundant information to be hidden within the text [6].

1.2.3 Linguistic Method

Linguistic method is a combination of syntax and semantics methods. Linguistic steganography considers the linguistic properties of generated and modified text, and uses linguistic structure as the space in which messages are hidden [6].

1.3 Criteria for Measuring Goodness of Data Hiding Algorithm

The efficiency of a technique used for a particular purpose is measured against different criteria. No algorithm matches all but a balance should be made before choosing one.

- Embedding Capacity: Embedding capacity (also known as payload) is the amount of data that can be hidden in a cover, compared to the size of the cover. This feature can be measured numerically in units of bit-per-bit (bpb) [7].
- Invisibility: Any data hidden in a cover causes it to be modified. Invisibility (also termed perceptual transparency or algorithm quality) is a measure of the amount of distortion (alteration) to the cover [7].
- Undetectability: An attacker may be able to detect the presence of hidden data in a given file by computing certain statistical properties of the file and comparing them to what is expected in that type of file [7].
- Robustness: This is a measure of the ability of the algorithm to retain the data embedded in the cover even after the cover has been subjected to various changes as a result of lossy compression and decompression or of certain types of processing such as conversion to analog and back to digital [7].

2. LITERATURE SURVEY

Hiding a secret text inside a cover media which is also a text is a tricky one where the cover text file has less redundant bits used for hiding.

2.1 Text Rotation in MS Excel Document

Convert the secret message to be hidden into binary bits using ASCII to Binary conversion method. Select the excel document to be used as a cover text. First find the non-empty cell and then find the length of non-empty cell. If the calculated length is less than or equal to limit (limit means number of letters in a cell) specified and if the secret bit is 1 then find that cell contains text or numeric. If it is text then rotate that cell to 1^{o} rotation. Else if it is numeric then rotate that cell to -1^{o} rotation. If the secret bit is 0 leave that cell with no rotation. Finally the formatted excel document is the stego text [8].

2.2 Mixed-Case Font in MS Word Document

First, the secret message is converted into bits as an array S. The Text file is chosen as a cover text. Each letter is separately taken in an array T. If ith element of S is bit 1 then the ith element of T is changed to capital letter else if ith element of S is bit 0 then ith element of T is changed to small letter. This method is iterated until the last index element of S is completed [9].

2.3 Font-Type in MS-Word Document

Before starting this method create a resemble font array which contains a table of cover document font and their resembling fonts for assumption 15 type of cover document fonts and their resembling font. Create a code table that contains coding of each symbol in secret message represented by three types of fonts, thus, 27 characters(English alphabets with space) can be hidden in 3 letters of cover using 3 different fonts, for example: similar font array of Century font is: Century = {Century751BT, CenturyOldStyle, CenturyExpdBT}. Secret message is embedded in Capital letters of the cover text document. First step is to find the font of the cover document to get the its resemble fonts array. Secondly, scan the capital letters in cover text i.e., needed three capital letters to hide one symbol. Finally, choose the corresponding font type of character in secret message from code table [10].

2.4 Inter-Sentence Spacing

This method hides binary bits in a text by changing one or two spaces after each end of character, for example a dot, a semicolon and so on. If the bit is 0 then single space is include in text and if the bit is 1 the double space is included. This method has some drawbacks. It is not efficient because it requires more text to hide few bits. Structure of the text is to be decided before hiding the bits (Some text, such as free-verse poetry, lacks consistent or well-defined termination characters.) The number of spaces is set after periods to one or more characters by most of the word processors. Finally, there is a lack of compatibility in using white spaces [11].

2.5 End-of-Line Spacing

White spaces are inserted at the end of the line to hide the bits. To hide one bit two spaces are inserted at the end of the line. To hide more bits first text is justified. To reveal the white space at the end of lines rules have been added. This method can be done with any text, and it will go unnoticed by readers, are the additional advantages of this method. A limitation to this method is that the hidden data cannot be retrieved from hard copy [11].

3. TEXT STEGANOGRAPHIC METHOD FOR MS-WORD DOCUMENT USING LOOK-ALIKE FONTS

Most commonly used format for information sharing is a text document. Format based text steganography method rely on TXT, MS Word, PDF, and PPT files. This method hides secret message in MS Word document using look-alike 0066onts of cover text. In the proposed method, Tahoma is taken as the base font for the cover text. The look-alike fonts for Tahoma are Verdana, Arial, and MS sans Serif. With the look-alike fonts the secret message are hidden in cover text.

The proposed method is implemented in three stages:

- Create pattern table (the stego key)
- Embedding process
- Extracting process

3.1 Create Pattern Table (Stego-key)

The pattern for each Hexadecimal number in secret message is represented by three types of fonts. The secret message (may be an alphabets (upper and lower case), numbers or special symbols, etc.,) can be hidden in 3 letters of cover text using 3 different fonts. Tahoma is taken as a font for the cover text. The look-alike fonts for Tahoma are: Tahoma = {Verdana, Arial, MS sans Serif}. If the pattern of a Hexadecimal number for 0 is 1, 1, 1, then fonts for first three alphabets will be first look-alike, first look-alike, and first look-alike from the pattern table. If the pattern of the next Hexadecimal number for 1 is 1, 1, 2, and then fonts for next three alphabets will be first look-alike, first look-alike, and second look-alike and so on. Table 1 shows the Pattern Table used for hiding process.

Table 1: Pattern table for Hex code of alphanumeric and symbols.

Index	Characters	F1	F2	F3
1	**0**	1	1	1
2	**1**	1	1	2
3	**2**	1	1	3
4	**3**	1	2	1
5	**4**	1	2	2
6	**5**	1	2	3
7	**6**	1	3	1
8	**7**	1	3	2
9	**8**	1	3	3
10	**9**	2	1	1
11	**A**	2	1	2
12	**B**	2	1	3
13	**C**	2	2	1
14	**D**	2	2	2
15	**E**	2	2	3
16	**F**	2	3	1

3.2 Embedding Process

Secret message is hidden in each English alphabet of the cover text document. Alphabets are alone used for hiding. Three alphabets are needed to hide one hexadecimal number. First step is to convert secret message into hex code representation. Then, choose the corresponding

pattern of each character in hex code representation from pattern table as shown in table1. Change the first three alphabets according to the pattern in the cover text document. After hiding first hexadecimal number, take the second hexadecimal number and choose its corresponding pattern from pattern table. Then change the next three alphabets in the cover text document according to the pattern selected. Likewise all the hexadecimal numbers are hidden in the cover document with the corresponding patterns. It is explained in the following Figure 3.

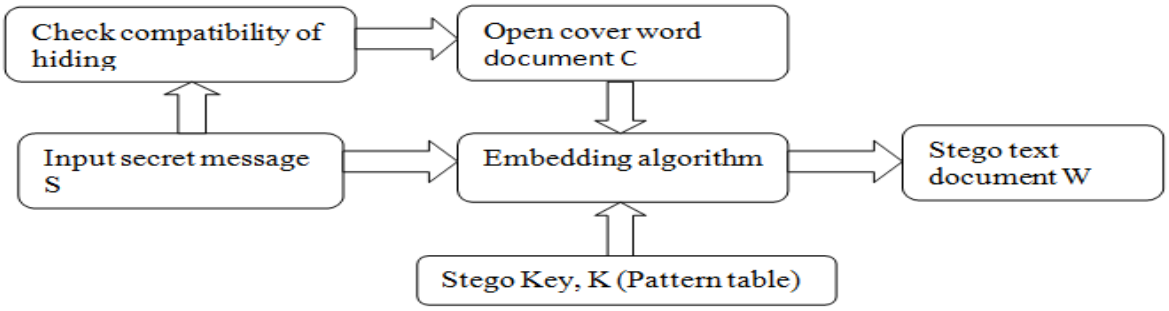

Figure 3. Methodology of Embedding Process.

3.2.1 Embedding Algorithm

The steps to hide secret message in the cover word document to receive stego text document as an output is given below.

Input: *Secret message(S), Stego key (K), Cover word document(C)* ***Output:*** *Stego text document (W)*
i. *Convert every character of secret message to Hex code representation based on ASCII representation.* *ii.* *For each digit in hex code representation of secret message* *ii.a.* *Retrieve its pattern from pattern table.* *ii.b.* *Change font type of three alphabet letters using look-alike fonts according to its pattern.* *iii.* *Repeat the step ii till all the Hex codes are hidden.* iv. *Return stego text, W.*

3.3 Extracting Process

The stego text document is sent to the recipient through communication media. For each three alphabets the code of one hiding hex code is determined using pattern table. The hex code representation is converted into text which symbolizes the secret message. The extracting process is explained in Figure 4.

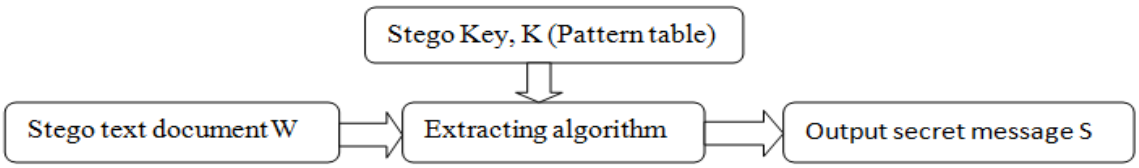

Figure 4. Methodology of Extracting Process.

3.3.1 Extracting Algorithm

The steps to extract secret message from the stego text document to receive secret message as an output is given below.

Input: *Stego text document (W), Pattern table (K)* ***Output:*** *Secret message(S)*
i. *Open stego document* *ii.* *For each three alphabets of the stego document* *ii.a.* *Determine the pattern using pattern table* *iii.* *Repeat the step (ii) until all the characters of stego document are checked.* *iv.* *Convert the hex code representation to ASCII code (Hex code).* *v.* *Secret message is extracted.*

4. RESULTS AND DISCUSSION

The sender chooses the cover text document by calculating the number of alphabet letters in cover text document with input secret message and hides secret message using the similar looking fonts. The embedding process is shown in the figure 5.

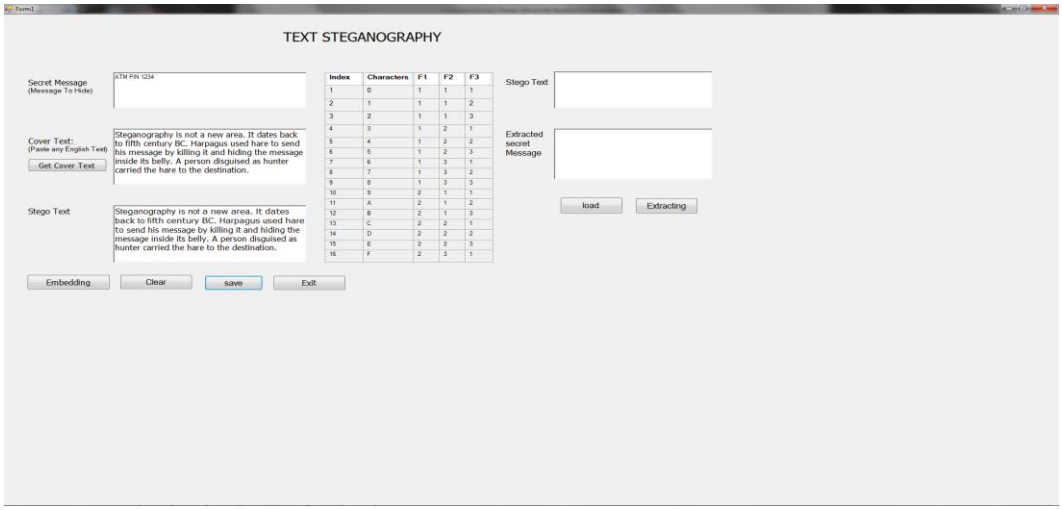

Figure 5: Embedding Process

In the receiver end the stego text document is loaded from the folder to retrieve the secret message. The Extracting process using pattern table retrieves the secret message. The extracting process is shown in the figure 6.

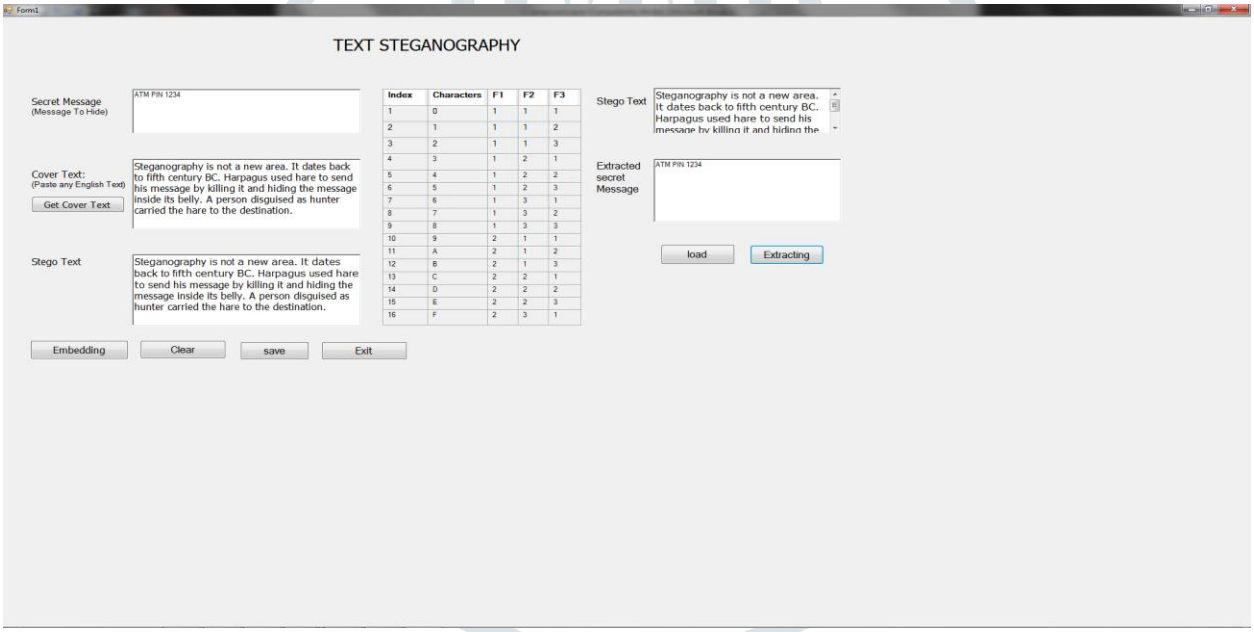

Figure 6: Extracting Process

In the above sample, the secret message is: ATM PIN 1234. The number of letter to hide is 12 (with space). Hex code for Symbol (A) is 41 in which pattern for 4 are: (1, 2, 2) i.e., (first look-alike, second look-alike, second look-alike) depending on the Pattern Table as shown in **Table 1**. The default font for the cover document is Tahoma, then 4 is coded by first three alphabet letters of cover document, by replacing it with look-alike. The steps are continued until all the letters from secret message are hidden. **Figure 7 and 8** represents cover document and stego document respectively.

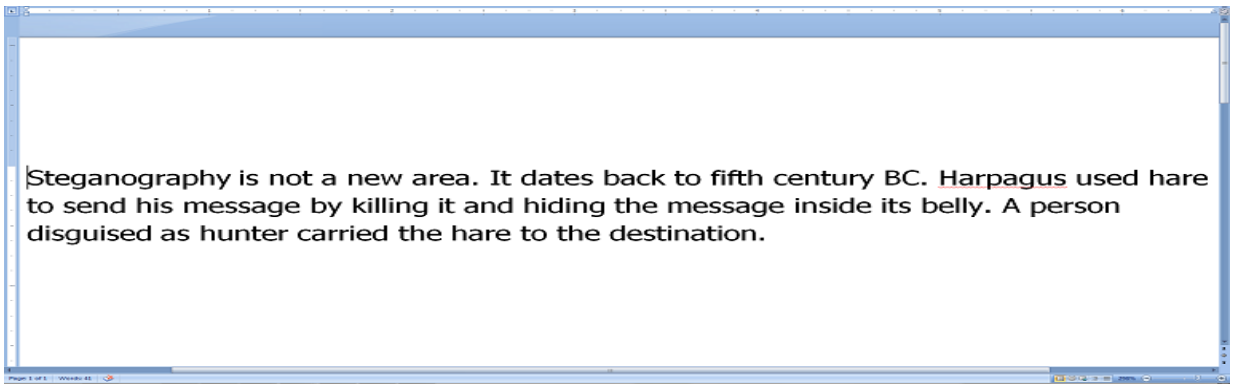

Figure 7. Cover Document

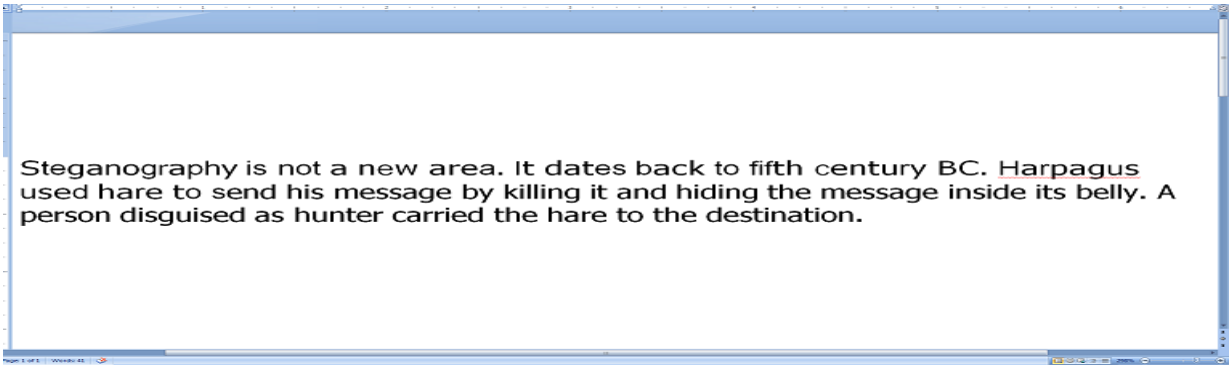

Figure 8. Stego Document

Adherence of Goodness Criteria for Data Hiding

Any meaningful English text document is taken as a cover text document. To improve the security the secret message is first converted into hex code representation. The converted hex code is hidden in a cover text document. The percentage capacity of the above sample is 20.33. The proposed MS-word document using look-alike fonts are not changing the words of the cover text document. Cover and stego text are exactly same in words; it gives the jaro score as 1. All the English alphabet letters in cover text document are taken to hide a secret message; its embedding capacity is very high. Distortion given to the stego text does not give any suspicion. Its invisibility is high. Look-alike fonts are used to hide the secret message in a cover text document. Look-alike fonts do not make any difference and it's hard to detect through Human vision when comparing between cover text and stego text. Its Undetectability is very high. Text remains unaltered when it is compressed or copied between the system programs. Its robustness is very high.

The proposed method is evaluated based on various parameters with different secret messages for a sample set of various sizes as listed below:

1. Ego (3 byte)
2. Minute (6 byte)
3. Happiness (9 byte)
4. Hello World (11 byte)
5. Smile is an inexpensive way to improve your looks. (50 byte)
6. Its not the load that breaks you down, its the way you carry it. (63 byte)
7. Don't find hundred reasons why you can't do a thing, but just find one reason why you can and do it. (100 byte)
8. Steganography is the art and science of data hiding. Steganography helps to hide information to be passed in secured way.(120 byte)
9. Steganography is not a new area. It dates back to fifth century BC. Harpagus used hare to send his message by killing it and hiding the message inside its belly.(160 byte)
10. Tide recedes and leaves behind bright sea shells on sand Sun sets but its warmth lingers on land Music stops and its echoes on in sweet refrains For every joy that passes, something beautiful remains (202 byte)

- **Capacity ratio**

  Capacity is defined as the ability of a cover text to hide secret message. The capacity ratio is computed by dividing the amount of hidden bytes by the size of the cover text in bytes.

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

  Assuming one character occupies one byte in memory, we have calculated the percentage capacity which is capacity ratio multiplied by 100 [12].

4.1 Text Similarity Measures

The similarity between the cover and stego text is calculated based on the various text parameters listed below.

- **Jaro–Winkler Distance**

  The Jaro-Winkler score (or distance) takes into account the number of matching characters and the transposition of characters in two strings. If the Jaro score is 0 then the two strings are dissimilar and 1 mean both are exactly same. Jaro score nearest to 1 indicates cover text and Stego text is closely similar. The number of matching (but different sequence order) characters divided by 2 defines the number of transpositions [13].

  In matching the characters, characters can't be more than floor (max (length (s1), length (s2)) / 2) - 1 spaces apart from each other. To get the Jaro score,

$$d_j = \begin{cases} 0 & \text{if } m = 0 \\ \frac{1}{3}\left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m}\right) & \text{otherwise} \end{cases}$$

  Where, m is the number of matching characters,
  s1 is the first string,
  s2 is the second string,
  t is the number of transpositions.

To calculate Jaro – Winkler Distance,

Jaro_score + (L * p * (1 - Jaro_score)),

Where, L is the length of the common prefix at the start of the string up to a maximum of 4, P is the constant scaling factor (usually 0.1 and not more than 0.25) [14].

- **Hamming distance and Levenshtein distance**

If the hamming distance and Levenshtein distance is 0 both the strings are similar. When the value of hamming distance and Levenshtein distance increases the dissimilarity between cover and stego text also increases.

Table 2 shows the observed percentage capacity, Jaro score, Levenshtein distance, Hamming distance of the proposed approach over the above ten experimental samples.

Table 2: Text similarity measures of the proposed approach over the ten experimental samples.

Samples	I	II	III	IV	V	VI	VII	VIII	IX	X
Capacity ratio	15.000	13.330	13.230	13.410	11.765	12.377	11.876	12.358	12.618	13.307
Jaro score	1	1	1	1	1	1	1	1	1	1
Levenshtein distance	0	0	0	0	0	0	0	0	0	0
Hamming distance	0	0	0	0	0	0	0	0	0	0

4.2 Image similarity measures

Histogram

The histogram represents the graphical representation of the tonal distribution in the digital images in digital image processing. It represents the number of pixels for each tonal value. The x-axis represents the tonal variations and y-axis represents the number of pixels in particular tonal value [15].

Pixel Difference Measurement

In this Measurement technique two images are taken one of them is image of the cover text and other is image of stego text whose image quality is to be assessed. A sum of an undistorted cover signal and an error signal helps in evaluation of quality of image signal. Commonly used Pixel Difference-based measures are: Mean Square Error (MSE) and Peak Signal-to-Noise Ratio.

- Mean Square Error (MSE), MSE is computed by averaging the squared intensity of the original (input) image and the resultant (output) image pixels.

$$MSE = \frac{1}{NM} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} e(m,n)^2$$

Where e(m, n) is the error difference between the original and the distorted images [16].

- Peak Signal-to-Noise Ratio

It is the ratio between the cover signal and the distortion signal in an image, given in decibels. In general, a higher PSNR value should correlate to a higher quality image, but tests have shown that this isn't always the case. Let us assume that $X = \{x_i | i = 1, 2 \ldots N\}$ and $Y = \{y_i | i = 1, 2 \ldots N\}$ are two infinite length, discrete signals (this discrete signal is considered as a visual signal), where N is the number of pixels in digital image and x_i and y_i are the values of i^{th} pixel of the digital image X and digital image Y respectively. Mathematically, the PSNR for the full reference Image quality metrics is given by:

$$PSNR(X, Y) = 10 \log_{10} \left(\frac{MPP^2}{MSE(X, Y)} \right)$$

Where, MPP is Maximum Possible Pixel in an image, i.e. if the image of 8 bit then the MPP = 28 – 1 = 255 pixels. MSE(X, Y) is the Mean Square error of the image X and Image Y [17].

Table 3 shows the observed PSNR and MSE method of the proposed approach over the above ten experimental samples.

Table 3: Image similarity measures of the proposed approach over the ten experimental samples.

Samples	I	II	III	IV	V	VI	VII	VIII	IX	X
PSNR	33.08	32.45	33.04	33.26	31.02	31.08	31.31	31.41	31.23	31.16
MSE	30.15	35.51	31.52	29.88	41.75	47.91	45.72	44.74	46.45	47.39

Table 4 shows the average percentage capacity, Jaro score, Levenshtein distance, Hamming distance, PSNR and MSE of the proposed approach

Table 4: Average text and image similarity measure of the proposed approach.

Method	Average % Capacity	Average Jaro Score	Average Hamming Distance	Average Levenshtein distance	Average PSNR	Average MSE
Proposed MS-word document using look-alike fonts	12.927	1.000	0	0	31.904	40.102

Histogram Comparison

To highlight the importance characteristics of the proposed method, a histogram comparison between the resultant stego image and the cover image is presented below in figure 9. The histogram test shows that the stego image is not affected by the hidden image. The histogram of the cover image is approximately the same as the histogram of stego image as shown in the figure 9.

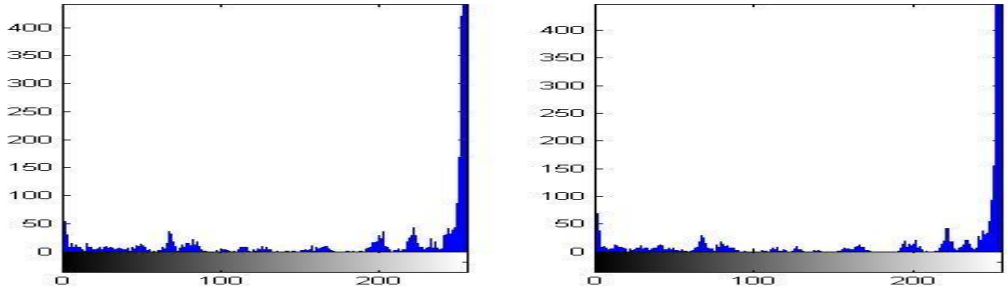

Figure 9. Histogram Analysis of Image of Cover Text and Stego Text

5. ANALYSIS OF THE PROPOSED METHOD

To analyze the performance of the proposed method, two existing text steganographic techniques mixed case font method and font type method have been taken. Table 5 shows the average of percentage capacity, jaro score, Hamming distance and Levenshtein distance, PSNR, MSE for the proposed using look-alike fonts, mixed case font and font type method.

Table 5: Comparison of the Three Methods with Text Similarity Measures and Average Capacity Ratio.

Methods	Average % Capacity	Text Similarity Measure			Image Similarity Measure	
		Average Jaro Score	Average Hamming Distance	Average Levenshtein Distance	Average PSNR	Average MSE
Mixed Case Font [9]	9.7	0.65	29.25	29.25	32.035	38.365
Font Type in MS-Word Document [10]	1.05	1	0	0	33.268	29.248
Proposed MS-Word Document using Look-Alike Fonts	13.74	1	0	0	32.957	31.765

Figure 10 shows the average capacity, average text similarity measures and average image similarity measures for the proposed using look-alike fonts, mixed case font and font type method.

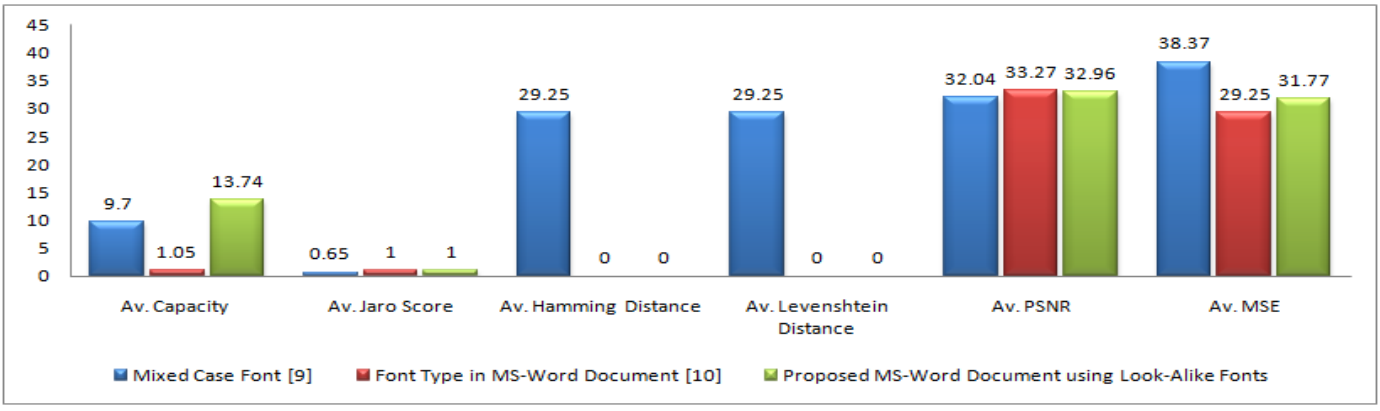

Figure 10: Average text similarity measure and average capacity ratio.

6. CONCLUSION

To analyze the performance of the proposed method, two existing text steganographic techniques mixed case font method and font type method have been taken. Average capacity ratio has been calculated for the proposed look-alike fonts, mixed case font and font type method. Proposed look-alike fonts show highest capacity ratio with the value 13.74 than mixed case font and font type method. In the proposed look-alike fonts and font type method secret message is hidden without altering the cover text, cover text and stego text are same. So, the average jaro score is 1 and average Hamming distance, average Levenshtein distance is 0. In the mixed case font method the average jaro score is 0.65 and average Hamming distance, average Levenshtein distance is 29.25 which show slightly dissimilar between the cover text and stego text. Proposed look-alike fonts, mixed case font and font type method have almost same and closer PSNR value with 32.957, 32.035 and 33.268 respectively. This shows same type of image quality. The proposed look-alike fonts and font type method have closer MSE value with 31.765 and 29.248 respectively. This shows even though the embedding capacity is larger than font type method which has low embedding capacity, proposed method's MSE has 2.5 differences with font type method. The histogram test shows that the stego image is not affected by the hidden image for the proposed method. The proposed look-alike fonts is the best method than mixed case font and font type method which takes benefit from both other method namely higher embedding capacity than mixed case font and takes best similarity measures from font type method.

7. REFERENCES

[1] Vidya G, Preetha R H, Shilpa G S, Kalpana V, "Image Steganography using Ken Ken Puzzle for Secure Data Hiding", Indian Journal of Science and Technology, Vol 7(9), 2014 Sep, pp. 1403–1413.

[2] Shirali-Shahreza M H Shirali-Shahreza M, "Steganography In Persian And Arabic Unicode Texts Using Pseudo-Space And Pseudo Connection Characters", Journal of Theoretical and Applied Information Technology © 2005 - 2008 JATIT. 2008; 4(8), pp. 682-87.

[3] Ramalingam M, Isa N A M, "A Steganography Approach over Video Images to Improve Security", Indian Journal of Science and Technology, 2015 Jan, 8(1), pp. 79–86.

[4] Kumar A, Pooja Km, "Steganography- A Data Hiding Technique", International Journal of Computer Applications (0975 – 8887), 2010 Nov., 9(7):pp.19-23.

[5] Bhattacharyya S, Banerjee I, Sanyal G, "A Novel Approach of Secure Text Based Steganography Model using Word Mapping Method(WMM)", International Journal of Computer and Information Engineering 2010; 4(2):PP.96-103.

[6] Mahajan S, Singh A, "A Review of Methods and Approach for Secure Stegnography", International Journal of Advanced Research in Computer Science and Software Engineering, ISSN: 2277 128X . 2012 Oct., 2(10):pp.67-70.

[7] Kumar K A, Pabboju S, Desai N M S, "Advance Text Steganography Algorithms: An Overview", International Journal of Research and Applications, ISSN (online): 2349-0020, Jan-Mar © 2014 Transactions; 1(1):pp.31-35.

[8] Yang B, Sun X, Xiang L, Ruan Z, Wu R, "Steganography in Ms Excel Document using Text-rotation Technique", Information Technology Journal ISSN 1812-5638 / DOI: 10.3923/itj.2011.889.893. 2011; 10(4): 889-93.

[9] Ali A A, Al – Hussien Seddik Saad, "New Text Steganography Technique By Using Mixed-Case Font", The Online Journal on Computer Science and Information Technology (OJCSIT), . 2013; 3(2):PP.138-41.

[10] Bhaya W, Rahma A M, AL-Nasrawi D, "Text Steganography Based On Font Type In Ms-Word Documents", Journal of Computer Science, ISSN: 1549-3636, © 2013 Science Publications, doi:10.3844/jcssp.2013.898.904 Published Online 2013;9 (7): 898-904.

[11] Singh H, Singh P K, Saroha K, " A Survey on Text Based Steganography", Proceedings of the 3rd National Conference; INDIACom-2009.

[12] Agarwal M, "Text Steganographic Approaches: A Comparison", International Journal of Network Security & Its Applications (IJNSA), 2013 Jan., 5(1):pp.91-106.

[13] Jaro-Winkler distance. 2015 Jan. Available from: http:// en.wikipedia.org/wiki/Jaro%E2%80%93Winkler_distance

[14] Approximate String Matching. 2015 Jan. Available from: http://biostat.mc.vanderbilt.edu/wiki/Main/ApproximateString Matching

[15] Rajani, Muhammad Tauheed Khan, "Data Hiding in Digital Image Processing Using Steganography: A Review", ISSN: 2321-9939, *IJEDR*, 2(3): 2994-96.

[16] Yusra A. Y. Al-Najjar, Dr. Der Chen Soong,"Comparison of Image Quality Assessment: PSNR, HVS, SSIM, UIQI", *International Journal of Scientific & Engineering Research*, Volume 3, Issue 8, August-2012, ISSN 2229-5518.

[17] Megha Goyal, Yashpal Lather, Vivek Lather, "Analytical Relation & Comparison Of PSNR And SSIM On Babbon Image And Human Eye Perception Using Matlab", *International Journal of Advanced Research in Engineering and Applied Sciences*, ISSN: 2278-6252.