Text Steganography Using character Spacing after Normalization

Syed Tahir Ali Shah,
Computer science department,
Iqra University, Islamabad, Pakistan
syedtahirkzm@gmail.com

Dr Aihab Khan,
Computer science department,
Iqra University, Islamabad, Pakistan
aihab@iqraisb.edu.pk

Dr Afaq Hussain
Computer science department,
Iqra University, Islamabad, Pakistan
drafaqh@gmail.com

Abstract— Steganography is technique of hiding a secret message in a cover medium. Text steganography uses a text file as a cover medium. There are many ways for hiding the secret message in the cover text without making any noticeable change in the cover. In character level embedding techniques the secret message is hidden in certain characters that are selected based on certain properties. These methods have low data hiding capacity because the occurrence of selected characters in the cover document is not uniform. In this research, we have addressed this problem by using the frequency modulation techniques and improved the data hiding capacity of the cover document. The font attributes, character spacing has been used to embed the secret data. With our proposed technique, one character can be hidden in three alphabets on the average and there are eight possible ways to hide each secret character. The capacity is very close to uniform and the secret message remained inconspicuous to an adversary.

Keywords — Text Steganography, Character space, cover document, stego document, Normalization

1 INTRODUCTION

The word steganography is made up of two Greek words “stegan” mean concealed or protected and ‘graphein' means writing [5][3]. Steganography is the art of hiding information in communication in such a way that no one doubts about the presence of the hidden information inside the cover media [1]. Moreover, Steganography has been regarded as the most significant tool in the modern era as information is very important nowadays [2]. Another discipline for secure communication is cryptography in which contents of the message are changed. Such changes capture the intruder’s attraction towards it and hence face many malicious attacks. Steganography means the practice of concealing a file, message, image, or video within another file, message, image, or video [4]. A cover that is used to hide information is called Stego-carrier and the secret information is called stego-text. Robustness, hiding capacity, and perceptual transparency are the most important criteria in designed method [5]. Robustness is the ability to protect of unseen data from corruption, when it is transmitted through the internet. Hiding capacity refers to the ability of storage of secret data in a fixed size of cover media. Perceptual transparency refers to the eavesdropper’s ability to figure out the hidden information. Data hiding using character level embedding techniques have low data hiding capacity as the occurrence of selected charters in cover document is not uniform [5]. In steganography hiding capacity is very important and to enhance this researchers have proposed using high frequency characters [21]. Frequency normalization has been utilized by some researchers to hide one character in maximum of three characters of cover document and each character has eight possible ways to hide data. Cover document is independent of capitalization and data is embedded using font attributes, character space and character position. This utilization of word attributes and character selection for steganography is proposed with comparatively better security of information. Our research aims to reduce the weaknesses of previous work. It suits well for steganography using word attributes. The advantages of Text steganography using word attributes and character for secret bits hiding are increase in capacity and larger application range. When we use font attributes every character of each word can be selected for data hiding. By applying this technique, we enhanced the capacity for data hiding and improved the three important criteria in designing steganography. Character level space has more hiding capacity as compared to text steganography using font format. In our study we have used the font attributes: character space. We have introduced new techniques for Text stenography. In our proposed technique we have used word attributes to increase data hiding capacity, robustness and perceptual transparency. We selected each character of each word and applied the selected attributes of space and position to hide secret character. This technique does not depend upon the case of character. If there is any change in the attributes of stego document the hidden data will be destroyed. Text steganography using different word font attributes and improved frequency modulation has not been used in the past, so our work was a good initiative for frequency modulation for text steganography with better security of user's information and revealed better outcomes. The previous approaches exploited attributes of word to achieve impermeability and had low hiding capacity. Our research aims were to overcome the limitation of previous steganography approaches and improve the data hiding capacity. We worked on text steganography using character spacing after Normalization This paper is organized as follow. Section 2 discussed the related work published and section 3 presents out materials and methods. Section 4 explains the proposed Algorithms and conclusions are shown in Section 5,Future work are drawn in section 6

2 RELATED WORKS

By and large textual data covered a lot of electronic content comprising articles, resolution, news, instruction material, and business information. Furthermore if these containers are used for the stenographic purpose then it is called text steganography. This type of steganography is harder than the other types of steganography. The text files format is alike what we look at a while at different types of file for example, in a picture, the structure of the text is not quite the same as what we observe. So it is difficult to use the file structure of the textual files to embed the secret message in the text. Instead of file structures, usually, contents are used for embedding the secret in the text. There is a low redundancy among the text so the capacity of this method is also low. Redundancy in the text depends upon the various features like font, semantic, syntax, production, etc. [6]. Text steganography based on font type in Microsoft Word documents. This method proposes to secret the message by using similar font types. Firstly the table was created by using the font used in the cover file and all the fonts were alike e.g. Microsoft san serif, Tahoma and Arial were similar font to the Verdana. Code table was prepared, which was used to conceal the secret message. If the Verdana font was used in the cover file than code table, three capital characters in the text were used to hide one character of secret message. But the method had a low capacity as there was the limited number of capital characters in the text and only one character could be embedded per three capital characters. Only 26 alphabets and space character could be embedded by this way. Changing the font type of text destroys the secret message. The main strengths were: Data was to be hidden using similar font types, one character was to be embedded per three capital characters, code table was prepared which was used to conceal the secret message, Perceptual transparency was very high and Robustness was also very high. The main weaknesses were changing the font type of text lead to the destruction of the secret message and able to embedded only 26 alphabets space character by using this technique. The limited number of capital characters in the text and this technique had low data hiding capacity were also important limitations. [9] Kashida [7] that is an extension character which is used between the characters in Arabic text for formatting. Azawi et al. also suggested 'Kashida' based steganography technique to conceal the secret message. Kashida is an extension character which used between the characters in Arabic text for formatting. But it could not be used at the start or at the end of words. It could also not be inserted between isolated Arabic characters. Before embedding the secret message, it was compressed by using the Huffman encoding scheme. The secret message got compressed up to 44%. The message was converted to binary form. Kashida inserted on a paper place to hide bit '1' and not inserted for a bit '0'. This method has better capacity than the previous method without compression. The main strength is: Kashida was an extension character which was used between the characters in Arabic text for formatting. This method had better capacity than the previous method without compression. The secret message got compressed up to 44%. It is compressed by using a Huffman encoding scheme. The weaknesses of this method are as follows. It could not be inserted between isolated Arabic characters. It could not be used in a start or at the end of words. Changing the format of text destroyed the secret message. Some researchers have used diacritics for hiding secret message. In Arabic text, eight Diacritics (Harakat) are used for vowel sounds. These diacritics are inserted above or below the letter. While inserting diacritics position of the cursor cannot be changed. Gutub et al. [13] proposed two technique on the basis of diacritics i.e. Textual and image. In both of these approaches, diacritic was inserted multiple times at the same place but the cursor remained at the same point in the document. There were two scenarios used i.e. stream and RLE. In-stream scenario message was converted to the binary form and then converted to a decimal number. One single diacritic inserted as many times as number was produced from the secret message. In RLE approach the first diacritic was used as many times as the number of '1' was present consecutively in secret message bit stream. A similar procedure was followed for the number of zeroes and second diacritics. In image technique on each stroke diacritic got darken. The textual technique was better for the soft copy cover medium while image approach was better for the hard copy of the cover. This method has better capacity than the other diacritic method and could not be used for Urdu, Arabic and Persian contents easily. This method changes the file size of the cover medium. In both, the approaches diacritic is inserted multiple times at the same location. The main weakness is that the textual technique was better for soft copy cover medium and not good for hard copy. Mahato et al. [14] proposed a method to hide secret message by using the font size of the text in Word document. The font size of space character was slightly changed to indicate a "1" bit and remained unchanged for "0" bit. This method had a very low capacity as only one bit could be hidden for each space character. If font size of cover text was changed then the embedded message would be destroyed. To hide the message in the document, the number of bits to be covered up was calculated and it should be more than the number of bits of the secret message. Thus the number of spaces should be greater than the number of bits of secret message. To conceal bit '1' the space character size was altered slightly without affecting the observable appearance of the document. This Stego was transmitted over the network. On receiver side size of space between words was checked to retrieve the secret message. This method has shown low capacity as one bit could be embedded per each word. If the size of the text changed or deleted from the document the embedded message might be destroyed. This technique based on two different form of لا to conceal the secret message in Persian and Arabic like scripts. Shirali-Shahreza et al. [16] used two different form of ﻻ to conceal the secret message in Persian and Arabic scripts. In Arabic and Persian special form of ﻻ was used. There is also another form of the word which was ﻻ. These two alternate forms were used to hide the secret bits in the text. Length of the secret message was embedded in the start of the document. To hide secret bit ‘1’ special form ﻻ was used while for a bit ‘0’ ل was used. Letter ل was wider than the special form ﻻ so the appearance of the contents was also affected by the insertion of ل. The size and character count were also increased for this method. Capacity ratio of the proposed method was very low as the only one-bit secret message could be embedded at each occurrence of the word ﻻ. The main strength was that the Length of the secret message was embedded in the start of the document. To hide secret bit ‘1’ special form ﻻ was used while for a normal bit ‘0’ ل was used. The size and character count were also increased for this method. These two alternate forms were used to hide the secret bits in the text. Robustness was very high in this method and the main weakness was hiding Capacity ratio of this technique was very low. One-bit secret message could be embedded at each occurrence of the word ﻻ. This Unicode space characters based technique in this method was proposed. These space codes could be used to conceal the secret message in Microsoft Word Document. This method concealed the secret information into, between words, between sentences, end of a line and between section spacing [17] it combines chose characters with normal space characters to outline mystery information bits to each blend. It standardizes the width of the Unicode character which has bigger width by decreasing their font size so their width equal even with the width of hair or Six-Per-Em Unicode characters. This thusly, builds hiding capacity as the number Unicode spaces which can be inserted into the end of line furthermore, between section spacing are expanded. From the trial results, it is clear that this scheme achieve higher hiding capacity, better hiding proficiency just as higher visual imperceptibility in contrast with other existing techniques Two type fonts were used with SMS i.e. proportional and system fonts. The outlook of these two types of fonts was the same to human vision. And not recognized to human eye this method hides data in(1 and 0) forms, stego message will look like ordinary message Bhaya et al [18] tried the font types to fix concealed message. A bit “1” equal to the system font and o equal to proportional front In the SMS no of bits to embed was almost same to the number of characters. So this technique had better capacity in comparison with the old method used for SMS. M. Hassan Shirali-Shahreza proposed a technique for hiding information in Arabic and Persian Unicode. In Persian and Arabic, some letters are joined together when they are written in succession in a word. However, there are two characters, zero width non-joiner (ZWNJ) and zero width joiner (ZWJ) that may keep the Persian and Arabic letters from joining or compels them to combine. In this strategy by utilizing these two extraordinary characters, data is hidden in Persian and Arabic Unicode content records [29]. This method is not dependent on any special format and it does not make any apparent changes in the original text and has perceptual transparency. One bit of data is hidden in each letter so it has capacity. Rajeev and Aruna [17] have proposed a space character based reversible steganography technique that hides the secret information in a word processor. The secret data is hidden in blank areas by changing their style and font types. Blank spaces are imperceptible characters so the progressions made in their textual style don't raise any doubt. Along these lines, the variations in the cover content are undetectable and therefore counter the visual assaults. No additional space between characters is added to hide the secret information which helps in expanding concealing proficiency. This technique is more robust and has higher capacity of concealing secret data in the cover document. Grothoff proposed a translation based scheme for hiding secret message in the errors (noise) that are naturally encountered in machine translation (MT). The secret message is embedded by performing a substitution procedure on the translated text using translation variations of multiple MT systems [11]. Typos and non-grammatical abbreviations in a text, e.g., emails, blogs, forums, etc., are employed for hiding data and is called noise based approach. However, this approach is sensitive to the amount of noise (errors) that occurs in a human writing. Krishnan [21] used frequency normalization method and character string mapping for text stenography to enhance occurrence of English alphabet in cover document. Occurrence of English alphabet in cover document is not uniform therefore this method is very useful. Font attribute properties like character-space are used to hide secret message. Seven-character string has been utilized with the help of character string mapping, with seven possible ways to hide each character. By this character string-mapping data hiding capacity has been increased, with each character string arranged in such way that cumulative frequency of each string difference is relatively same. This method requires a maximum of 4 characters of cover document to hide one secrete character All embedding techniques that have been discussed above are character level or bit level embedding, and both have the disadvantage of low data hiding capacity because occurrence of character in cover document is not uniform. Frequency normalization and character string mapping may improve occurrence of characters to a uniform level thus enhancing the capacity. Robustness is another issue with these techniques as if the attacker changes the cover document to the same character space size than all hidden data will be destroyed. Most text steganography techniques have low data hiding capacity because data is hidden only in a specific character. The disadvantage of these techniques is that if the attacker changed the attributes of word document then some data will be destroyed. Formalization of a robust algorithm for data hiding with high capacity utilizing character normalization is the target of this research.

.

3. MATERIALS AND METHODS

We aim to improve frequency normalization for optimal selection of characters and then utilization of font attributes to achieve high data hiding capacity and robustness. The proposed technique is compared to other frequency normalization methods [21] and evaluation is done on robustness and data hiding capacity. The cover document has been created by random selection of text from newspaper website.

4. PROPOSED METHOD

In this section we explain our proposed methodology in detail with data embedding and extraction procedure. Our method is based on using frequency normalization for selection of characters and then utilizing font attributes like character space and position for character string mapping.

Steps of Proposed Method

In this section, we describe the steps in our algorithm. These were implemented in a c# method and consisted of the following components.

- Frequency Normalization
- Character and String Mapping
- Embedding process
- Extraction process

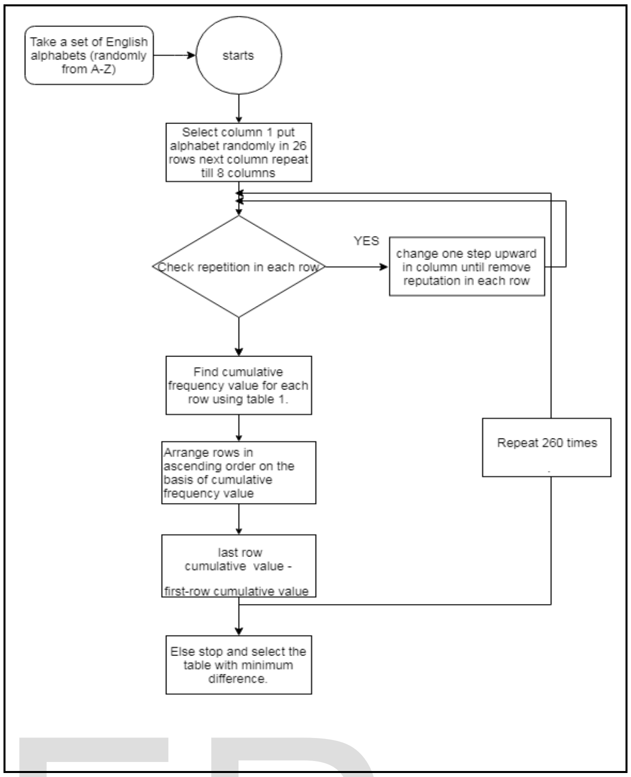
Frequency Normalization

In frequency normalization, we use sets of eight alphabets in such way that the cumulative frequency values are constant and there is no repetition of the alphabet in rows and columns

The complete algorithm is given below.

1. Take set of English alphabets (randomly selected from A-Z)
2. Select column 1 put alphabet randomly in 26 rows next column repeat till 8 columns
3. Check repetition in each row
4. If no repetition then go to next step else change one step upward in column until remove reputation in each row
5. Find cumulative frequency value for each row using table 1.
6. Arrange rows in ascending order on the basis of cumulative frequency value
7. Subtract the first-row cumulative value from last row cumulative value
8. Repeat 260 times select the minimum.
9. Else stop and select the table with minimum difference.

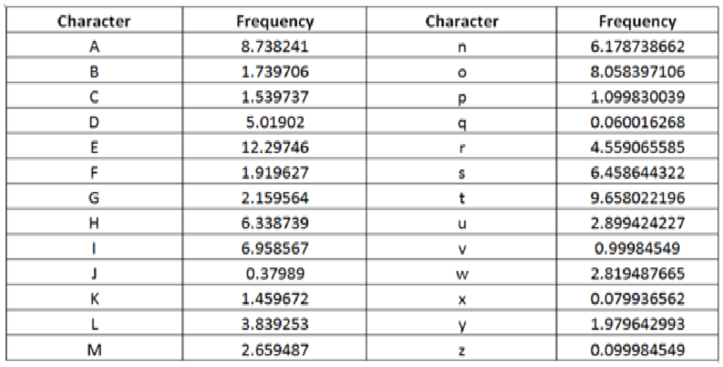
The output of Frequency normalization is given in table 2.


Take a set of English alphabets (randomly from A-Z)
starts
Select column 1 put alphabet randomly in 26 rows next column repeat till 8 columns
Check repetition in each row
YES
change one step upward in column until remove reputation in each row
Find cumulative frequency value for each row using table 1.
Arrange rows in ascending order on the basis of cumulative frequency value
Repeat 260 times
last row cumulative value - first-row cumulative value
Else stop and select the table with minimum difference.

Genration of Frequency normalization

FIGURE 1

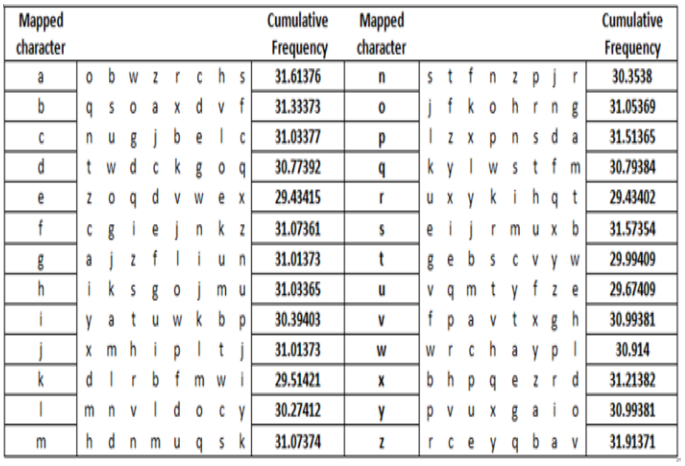
TABLE 1



Character	Frequency	Character	Frequency
A	8.738241	n	6.178738662
B	1.739706	o	8.058397106
C	1.539737	p	1.099830039
D	5.01902	q	0.060016268
E	12.29746	r	4.559065585
F	1.919627	s	6.458644322
G	2.159564	t	9.658022196
H	6.338739	u	2.899424227
I	6.958567	v	0.99984549
J	0.37989	w	2.819487665
K	1.459672	x	0.079936562
L	3.839253	y	1.979642993
M	2.659487	z	0.099984549

Standard occurrence Frequency of English alphabet [23]

TABLE 1



Mapped character		Cumulative Frequency	Mapped character		Cumulative Frequency
a	o b w z r c h s	31.61376	n	s t f n z p j r	30.3538
b	q s o a x d v f	31.33373	o	j f k o h r n g	31.05369
c	n u g j b e l c	31.03377	p	l z x p n s d a	31.51365
d	t w d c k g o q	30.77392	q	k y l w s t f m	30.79384
e	z o q d v w e x	29.43415	r	u x y k i h q t	29.43402
f	c g i e j n k z	31.07361	s	e i j r m u x b	31.57354
g	a j z f l i u n	31.01373	t	g e b s c v y w	29.99409
h	i k s g o j m u	31.03365	u	v q m t y f z e	29.67409
i	y a t u w k b p	30.39403	v	f p a v t x g h	30.99381
j	x m h i p l t j	31.01373	w	w r c h a y p l	30.914
k	d l r b f m w i	29.51421	x	b h p q e z r d	31.21382
l	m n v l d o c y	30.27412	y	p v u x g a i o	30.99381
m	h d n m u q s k	31.07374	z	r c e y q b a v	31.91371

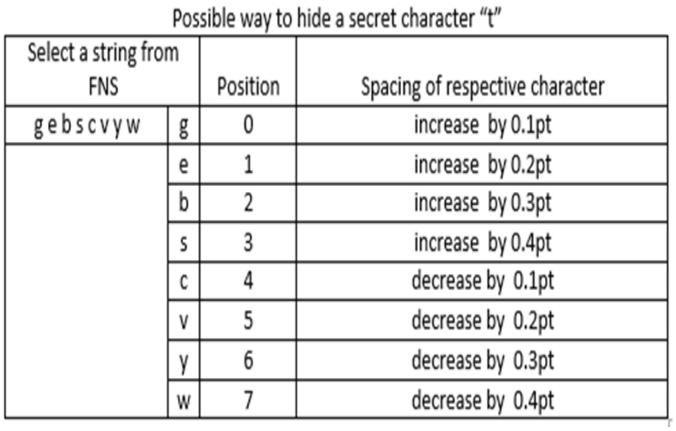
English character and string mapping

Character level embedding techniques require a certain character to embed a secret message. Occurrence of alphabets in English documents is not uniform, with some alphabets more frequently used than others. This causes the data hiding to be non-uniform. For example, a low occurring character Q needs 2128 cover characters to get embedded. Frequency normalization gives eight possible ways to hide Q in string [k, y, l, w, s, t, f, and m]. In character level embedding technique occurrence value of Q is 0.0470% while after frequency normalization occurrence value of Q is 30.79384 %. Thus after frequency normalization we need a maximum of three cover characters to hide one character. The cumulative probability is given by equation 1. P= (100/26)*8=30.77 (1) Where the number of characters is 8 and the number of alphabets is 26.

Character and String Mapping

The 8 character normalized strings Map generated 26 strings as shown in table 2

TABLE 2



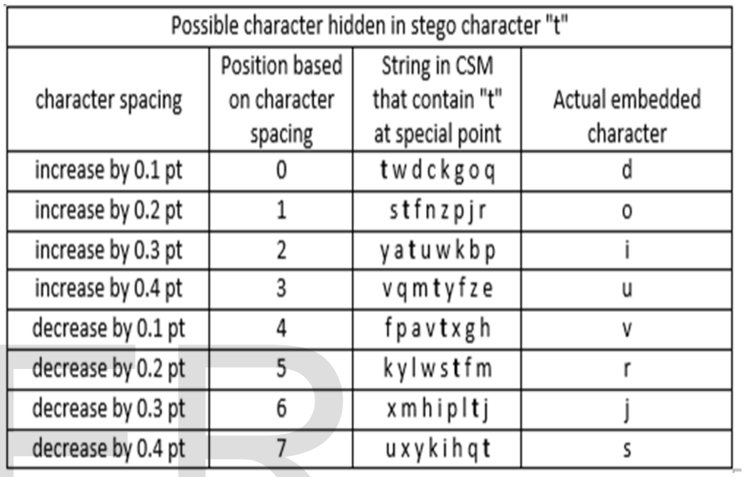
Possible way to hide a secret character "t"

Select a string from FNS		Position	Spacing of respective character
gebscvyw	g	0	increase by 0.1pt
	e	1	increase by 0.2pt
	b	2	increase by 0.3pt
	s	3	increase by 0.4pt
	c	4	decrease by 0.1pt
	v	5	decrease by 0.2pt
	y	6	decrease by 0.3pt
	w	7	decrease by 0.4pt

Mapping increases low occurring character probability and makes the probabilities close to uniform and also increases the hiding capacity as compared to existing character level embedding techniques. Let "t" be the first character in stego-document then there are eight possible ways to hide secret character t in cover document character string mapping (CSM) that change attributes of character space with respect to CSM as given in table **4.**

TABLE 4

Possible Character hiding using csm



Possible character hidden in stego character "t"			
character spacing	Position based on character spacing	String in CSM that contain "t" at special point	Actual embedded character
increase by 0.1 pt	0	twdckgoq	d
increase by 0.2 pt	1	stfnzpjr	o
increase by 0.3 pt	2	yatuwkbp	i
increase by 0.4 pt	3	vqmtyfze	u
decrease by 0.1 pt	4	fpavtxgh	v
decrease by 0.2 pt	5	kylwstfm	r
decrease by 0.3 pt	6	xmhipltj	j
decrease by 0.4 pt	7	uxykihqt	s

Embedding Process

In this study, a secret message is embedded in cover document using the word attributes namely, character space and position. To hide data we changed character space value (Increased/decreased by 0.1, 0.2, 0.3, 0.4). The embedding process starts from the first letter of cover document and increases position 1 when secret message ends.

Embedding Capacity (= No of hidden characters) / (*total number of character used for data hiding*)

Input: Cover document(c) character string mapping table (2) Secret message (sm);
Output: Stego text (SM)

- *Read the cover document*
- *Isolate each word of the cover document.*
- *Read the attribute character space position for the each character*
- *Make character space to default value 0*
- *Read sm*
- *Take each element of the encrypted sm and Find the code from the csm (table2)*
- *Changed the attributes of the cover document character as per above code*
- *Continue till end of secret message sm*
- *Output: SM*

End

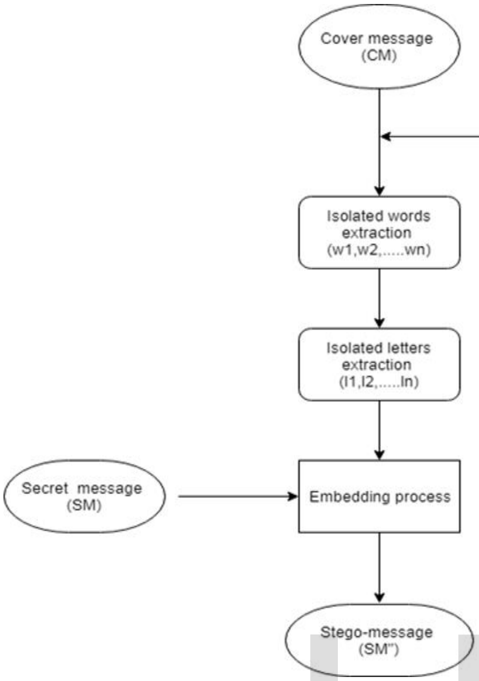

Cover message (CM)
CSM
Isolated words extraction (w1,w2,.....wn)
Isolated letters extraction (l1,l2,.....ln)
Secret message (SM)
Embedding process
Stego-message (SM")

FIGURE 2 Block Diagram for Embedding Process

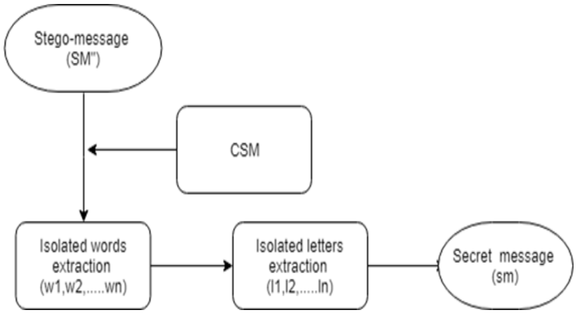

Stego-message (SM")
CSM
Isolated words extraction (w1,w2,.....wn)
Isolated letters extraction (l1,l2,.....ln)
Secret message (sm)

FIGURE 3 BLOCK DIAGRAM FOR EXTRACTING PROCESS

Extracting Process

Check character space for each character and compare to character string mapping table. The steps below show the reverse of embedded process also given in figure 3

Input Parameters: Stego message (SM) Character string mapping (CSM)

Output: secret message(SM)

- open: SM
- Separate words
- Isolated each letter of each count word
- Retrieved code w.r.t character from the CSM
- If the character space same then end of sm is reached
- Else. Found corresponding secret symbol using code End.

Hiding operation

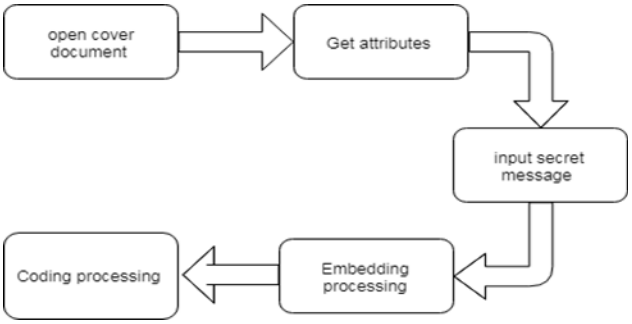
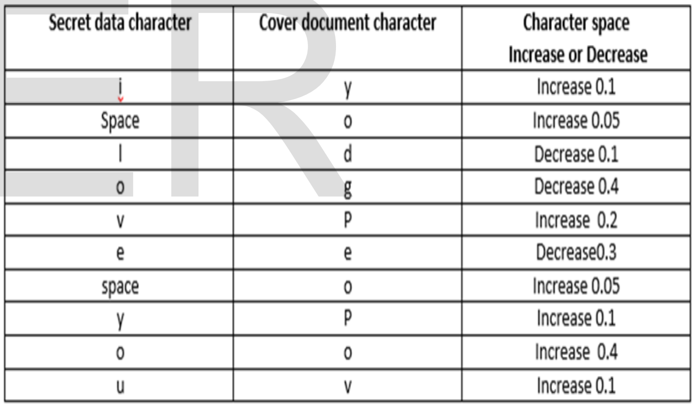
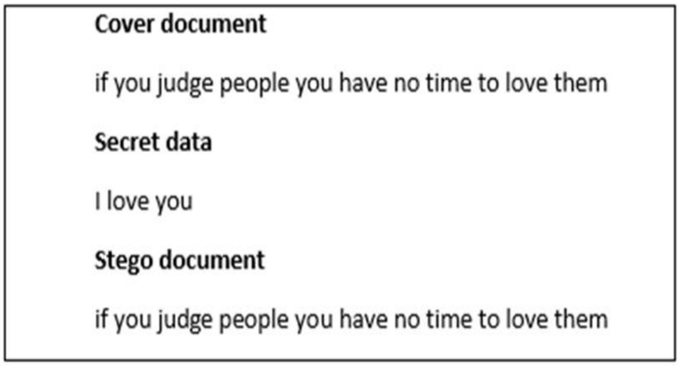


FIGURE 3

Table 5 Example OF proposed method



Secret data character	Cover document character	Character space Increase or Decrease
i	y	Increase 0.1
Space	o	Increase 0.05
l	d	Decrease 0.1
o	g	Decrease 0.4
v	P	Increase 0.2
e	e	Decrease0.3
space	o	Increase 0.05
y	P	Increase 0.1
o	o	Increase 0.4
u	v	Increase 0.1



Cover document

if you judge people you have no time to love them

Secret data

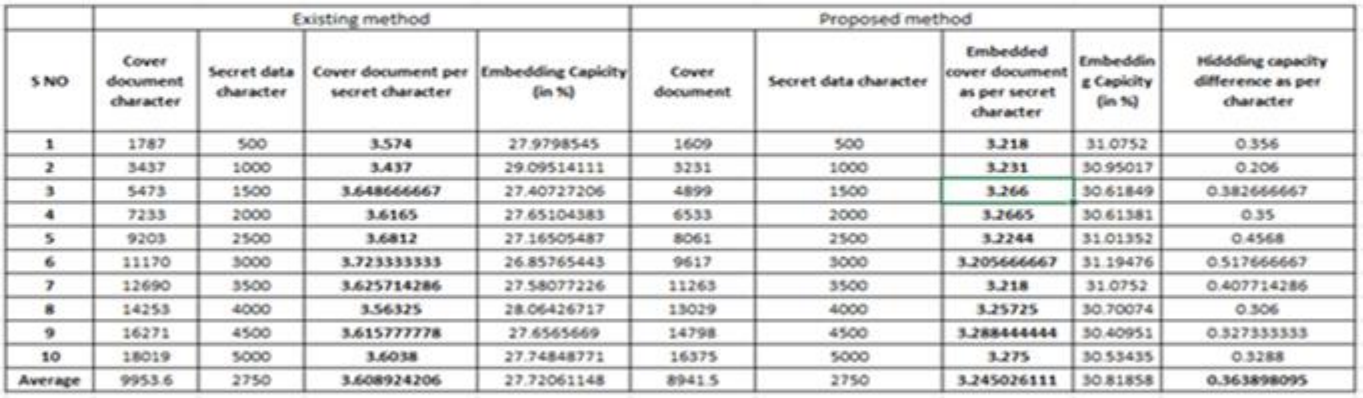
I love you

Stego document

if you judge people you have no time to love them

FIGURE 4 Experimental results

TABLE 5 EXPERIMENTAL COMPARISON



	Existing method				Proposed method				
S NO	Cover document character	Secret data character	Cover document per secret character	Embedding Capicity (in %)	Cover document	Secret data character	Embedded cover document as per secret character	Embeddin g Capicity (in %)	Hiddding capacity difference as per character
1	1787	500	**3.574**	27.9798545	1609	500	**3.218**	31.0752	0.356
2	3437	1000	**3.437**	29.09514111	3231	1000	**3.231**	30.95017	0.206
3	5473	1500	**3.648666667**	27.40727206	4899	1500	**3.266**	30.61849	0.382666667
4	7233	2000	**3.6165**	27.65104383	6533	2000	**3.2665**	30.61381	0.35
5	9203	2500	**3.6812**	27.16505487	8061	2500	**3.2244**	31.01352	0.4568
6	11170	3000	**3.723333333**	26.85765443	9617	3000	**3.205666667**	31.19476	0.517666667
7	12690	3500	**3.625714286**	27.58077226	11263	3500	**3.218**	31.0752	0.407714286
8	14253	4000	**3.56325**	28.06426717	13029	4000	**3.25725**	30.70074	0.306
9	16271	4500	**3.615777778**	27.6565669	14798	4500	**3.288444444**	30.40951	0.327333333
10	18019	5000	**3.6038**	27.74848771	16375	5000	**3.275**	30.53435	0.3288
Average	9953.6	2750	**3.608924206**	27.72061148	8941.5	2750	**3.245026111**	30.81858	**0.363898095**

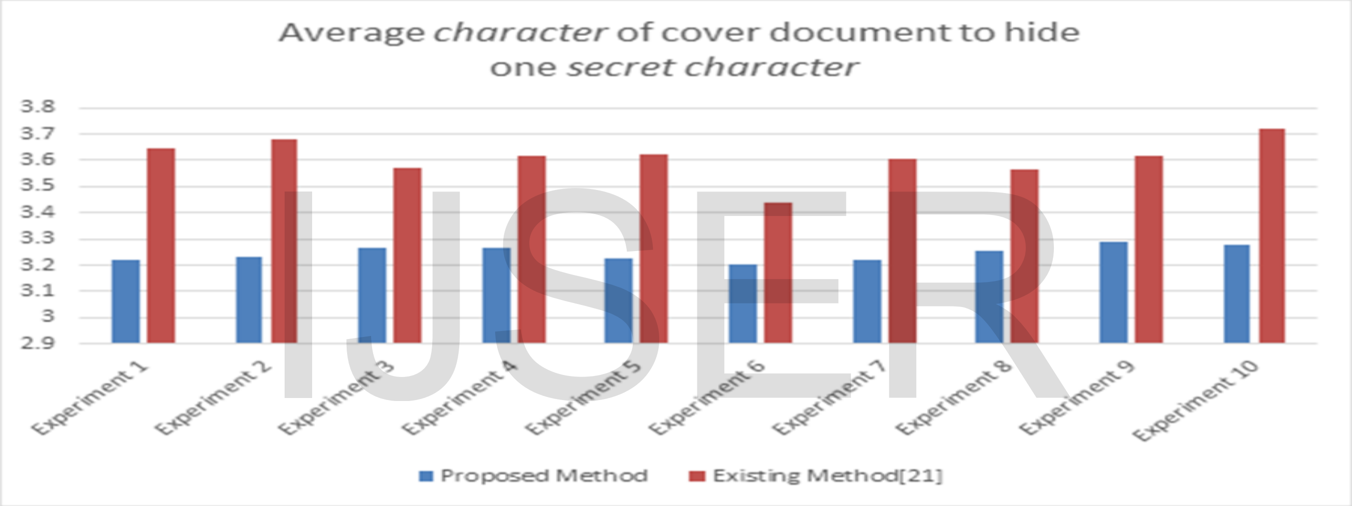


FIGURE 5 GRAPHIC REPRESENTATION COMPARISON OF PROPOSED AND EXISTING METHOD

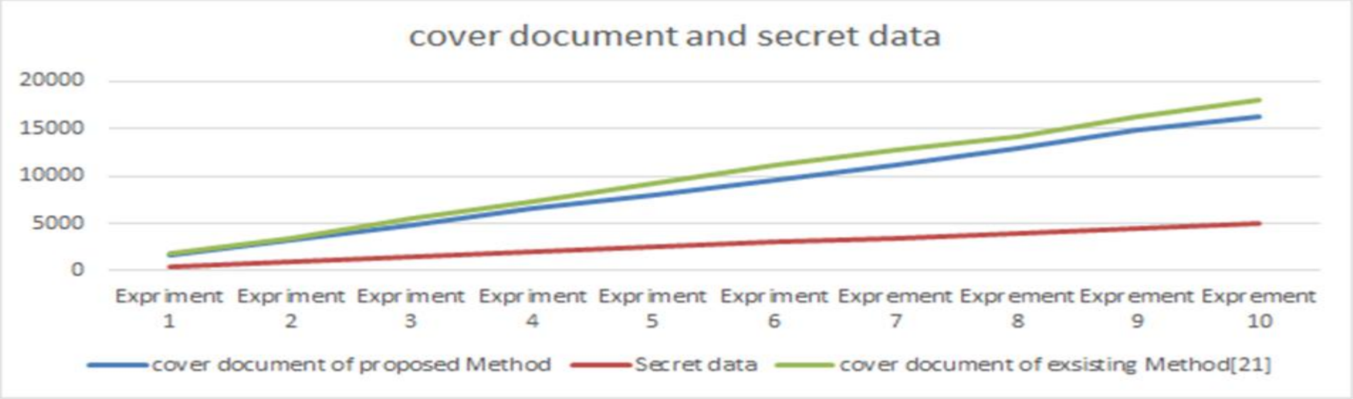
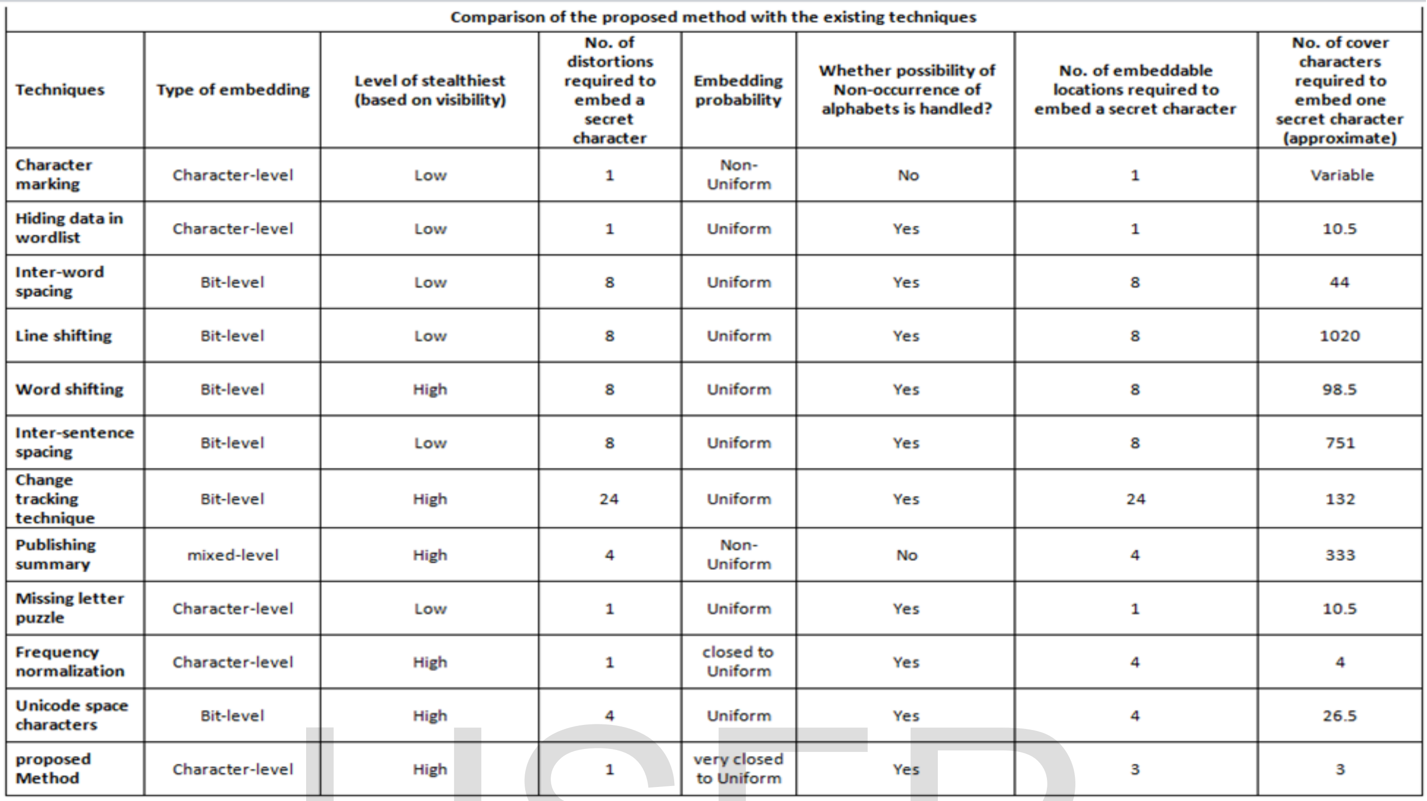


Figure 6 Graphic representation of cover document size and secret data



Comparison of the proposed method with the existing techniques

Techniques	Type of embedding	Level of stealthiest (based on visibility)	No. of distortions required to embed a secret character	Embedding probability	Whether possibility of Non-occurrence of alphabets is handled?	No. of embeddable locations required to embed a secret character	No. of cover characters required to embed one secret character (approximate)
Character marking	Character-level	Low	1	Non-Uniform	No	1	Variable
Hiding data in wordlist	Character-level	Low	1	Uniform	Yes	1	10.5
Inter-word spacing	Bit-level	Low	8	Uniform	Yes	8	44
Line shifting	Bit-level	Low	8	Uniform	Yes	8	1020
Word shifting	Bit-level	High	8	Uniform	Yes	8	98.5
Inter-sentence spacing	Bit-level	Low	8	Uniform	Yes	8	751
Change tracking technique	Bit-level	High	24	Uniform	Yes	24	132
Publishing summary	mixed-level	High	4	Non-Uniform	No	4	333
Missing letter puzzle	Character-level	Low	1	Uniform	Yes	1	10.5
Frequency normalization	Character-level	High	1	closed to Uniform	Yes	4	4
Unicode space characters	Bit-level	High	4	Uniform	Yes	4	26.5
proposed Method	Character-level	High	1	very closed to Uniform	Yes	3	3

TABLE 3

Our proposed method has high data hiding capacity as compared to other frequency normalization techniques [21] because frequency normalization needs 4 cover document characters to hide one secret character on the average while our proposed method needs only 3 character of cover document on the average. To hide one character in our proposed method occurrences of English alphabets is near to uniform as compared [21] to existing frequency normalization techniques because cumulative frequency value of character string of proposed method is very close to each other and its robustness as high as compared to other docs [9]

5 Conclusion

In this research, our proposed technique provided concealed data in word document using word font attributes called character space. In character space, data hiding rate is very high. Hiding capacity depends upon the number of characters in the cover document. A novel technique, named Frequency normalization System in combination with the Character String Mapping has been proposed to achieve the uniformity in embedding probability and thereby to increase the embedding capacity occurrence of character is closed to uniform each character of each word can hide one character by our tniqu and does not depend upon the case sensitive. A further improvement may be accomplished in this domain. The efficiency and effectiveness of the proposed technique can be
Improved and enhanced though Security, robustness, and capacity

6 Future Works

In proposed method, we have utilized only English alphabet, and in our method secret message is not case sensitive, future method may be case sensitive embedding special character and numerical..

References

[1] A.A. Mohamed, "An improved algorithm for information hiding based on features of Arabic text: A Unicode approach," in Egyptian Informatics Journal (2014) (pp 79–87)

[2] Shirali-Shahreza, M., & Shirali-Shahreza"An Improved Version of Persian/Arabic Text Steganography Using" La" Word" in IEEE (2008) (pp. 372-376).

[3] Mohan, M., & Anurenjan, P. R. "A new algorithm for data hiding in images using contourlet transform" in Recent Advances in Intelligent Computational Systems (RAICS), IEEE 2011 (pp. 411-415)

[4] Shirali-Shahreza, M., & Shirali-Shahreza. "Persian/Arabic Unicode text steganography in Information Assurance and Security," in Fourth International Conference IEEE. 2008. (pp. 62-66).

[5] A.A. Mohamed, "An improved algorithm for information hiding based on

features of Arabic text: A Unicode approach," Egyptian Informatics Journal (2014) (PP 79–87)

[6] Changder, S., Ghosh, D., &Debnath, N. C.. "Linguistic approach for text steganography through Indian text", 2nd International Conference in computer technology Development (IEEE 2010 (pp. 318-322).

[7] Al-Azawi AF & Fadidhil MA. "Arabic text steganography using Kashida extensions with Huffman code" in Journal of Application Science; (2010). (PP 436–9.)

[8] Elkamchouchi, H. and M. Negm,. "Hiding English Information in Extended Arabic Characters" in Proceedings of the 20th National Radio Science Conference, Mar. 18-20, IEEE Explore Press, (2003) pp: C12-1-8.

[9] WesamBhaya, Abdul MonemRahma and Dhamyaa AL-Nasrawi, "Text Steganography Based on Font Type in MS-Word "in Journal of Computer Science (2013) 9 (7): 898-904

[10] M. H. S. Shahreza, and M. S. Shahreza, "A new synonym text steganography," in Int. Conf. on Intelligent Information Hiding and Multimedia Signal Processing, (2006), pp. 1524-1526.

[11] Christian Grothoff "Translation-Based Steganography" Department of Computer Science, University of Denver, (2010).

[12] Sangita Roy and Manini Manasmita "A Novel Approach to Format Based Text Steganography" ICCCS'11, , Rourkela, Odisha, India. February 12–14, 2011 pp511-516

[13] Al-Haidari, F., Gutub, A., Al-Kahsah, K., & Hamodi, J. Improving security and capacity for Arabic text steganography using 'Kashida' extensions" in Computer Systems and Applications, IEEE/ACS International Conference on IEEE 2009. (pp. 396-399).

[14] Susmita Mahato "A Novel Approach to Text Steganography Using Font Size of Invisible Space Characters in Microsoft Word Document" Intelligent Computing, Networking, and Informatics January 2014 pp 1047-1054|

[15] Aabed, M., Awaideh, S. M., Elshafei, A. R. M., & Gutub, A. "Arabic diacritics based steganography" in Signal Processing and IEEE Communications. IEEE International Conference on (2007, November). (pp. 756-759).

[16] Li, L., Huang, L., Zhao, X., Yang, W., & Chen, Z. "A statistical attack on a kind of word-shift text-steganography" in Intelligent Information Hiding and Multimedia Signal Processing, IIHMSP International Conference on ' IEEE 08 (pp. 1503-1507).

[17] Rajeev Kumar , Satish Chand , and Samayveer Singh, "An efficient text steganography scheme using Unicode Space Characters", in International Journal of FORENSIC COMPUTER SCIENCE (2015) PP (8-14

[18] Bhaya, W.S. "Text hiding in mobile phone simple message service using fonts " in Advanced Statistical Steganalysis. 2nd Ed., Springer, Berlin (2010) PP 285

[19] Khair ullah, M. "A novel text steganography system using font color of the invisible characters in Microsoft Word documents" in Computer and Electrical Engineering, Second International Conference on (2009, December). IEEE (Vol. 1, pp. 482-484)

[20] Lingyun xiang, Xingming sun,can gan , "research on steganalysis for text steganography based on font format" in third international symposium on information assurance and security (2007) pp 490-495

[21] Bala Krishnan, "Text steganography a novel character-level embedding algorithm using font attribute" in Wiley Online Library (wileyonlinelibrary.com). (2017)

[22] Li, L., Huang, L., Zhao, X., Yang, W., & Chen, Z. "A statistical attack on a kind of word-shift text-steganography" in Intelligent Information Hiding and Multimedia Signal Processing, 2008. IIHMSP'08 International Conference on (pp. 1503-1507). IEEE.

[23] Behrouz A Forouzan, Debdeep Mukhopadhyay. "Cryptography and network security". Tata McGraw-Hill Education, India, 2011.