An Overview of Text Steganography

R.Bala Krishnan[1*], Prasanth Kumar Thandra[2], M.Sai Baba[1]

[1]Indira Gandhi Centre for Atomic Research, HBNI, Kalpakkam, Tamil Nadu, India
[2]Indira Gandhi Centre for Atomic Research, Kalpakkam, Tamil Nadu, India
Correspondence E-Mail: balakrishnanr1987@yahoo.co.in

Abstract— **Steganography embeds a secret message inside an innocent looking cover medium, stealthily, without creating any attention. The cover medium used can be a text, image, audio, video, network packets, etc. To embed the secret, steganographic techniques rely on the redundant information of the used cover medium or the properties which human perceptual system fails to differentiate. Hence the choice of using text document as a cover medium is the most difficult one as they have less redundant information. However, as text documents are widely used in organizations, using text document as a cover medium might be a preferable choice in such environment. Considering this, this paper presents an overview of the existing techniques of text steganography and its classification. A comparison of the existing techniques is also provided.**

Index Terms — **cover medium, text steganography, stealthy**

I. INTRODUCTION

Internet is extensively used to exchange different kinds of information (text, image, audio and video). But the underlying protocols do not imply any strict rules for the security of data baring some cryptographic protocols. Hence, it is up to the end users to be cautious about the security of their data during transit. Although, these cryptographic techniques facilitate features like authentication, confidentiality, integrity and non-repudiation, it fails to provide secrecy. So, any third party who is observing the transmitted data can easily figure out that something important is being transmitted. To overcome this limitation, data hiding techniques like steganography is adapted. Steganography avoids such attraction by performing the communication in a stealthy manner [1] [2]. This policy encourages many sensitive organizations to adapt the steganographic techniques for a much better communication.

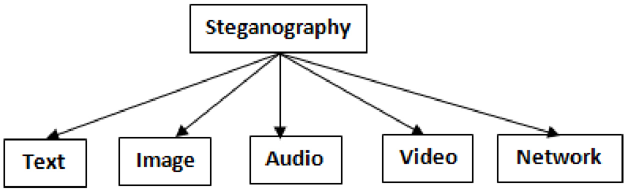


Fig. 1 Types of steganography

Steganography is not a recently evolved technique and its root lies in the ancient Greek period. Steganography means "Cover Writing". Steganography embeds a secret message inside a cover medium in an unnoticeable manner. While the ancient techniques exploited the tattoo of slaves, invisible inks, micro dots, wax tablets, semagrams, etc, to embed the secret, modern digital steganographic techniques either rely on the redundant information of the used cover medium or the properties which human perceptual system fail to differentiate. Hence the cover medium/work used can be a text, image, audio, video, etc. Based on the type of cover medium used, steganography can be divided into different categories (refer Fig. 1). Among these, text steganography is considered to be the harder one, as the amount of the redundant information present in the text document is low. In addition, even a small alteration in the text document is notable as the structural and visual appearance of text documents is directly related [3]. However, the wide usage of text documents in many organizations still insists the text documents as a better choice, while deciding a steganographic cover. Besides, the low processing power, due to the smaller file size, and low bandwidth during transmission adds more value to the decision.

II. CLASSIFICATION OF TEXT STEGANOGRAPHY

Based on the type of the embedding technique, text steganography can be divided into three broad categories as character-level, bit-level and mixed-type embedding [4] [5]. All these categories and its respective sub-categories are specified in Fig. 2 and are discussed below in detail.

1. *Character-level Embedding* [4] [5]

Character-level embedding technique directly embeds the secrets as characters inside the text document. Depending on the necessity of the cover work/document, this type of embedding can further be classified into two categories as Cover_Document_Necessary and Cover_Document_ Unnecessary [4] [5].

1.1 Cover_Document_Necessary [4] [5]

The various methods described under this category require a cover document and embeds the secrets as characters directly by marking them in it. Hence, it requires that all the alphabets that are present in the secret message must be present in the document and in the same order. This is the drawback of this method, as the occurrence frequency of alphabets in English text is not uniform [6].

1.1.1 Character Marking [4] [7] [8] [9] [10]

As a way of embedding the secrets, the respective characters in the cover document can be marked serially. That is, changing the font style or size, making a character bold, italic, underline, etc. The secret message will be exposed when the marked characters are identified and grouped together.

Eg: Technique**s** lik**e** **c**ryptog**r**aphy do not **e**nsure **t**he secrecy.

978-1-5090-4740-6/17/$31.00 ©2017 IEEE

Embedded Secret: secret

1.1.2 Character and String Mapping [4]

This method alters a font attribute, called the *character spacing*, in the cover document to embed the secret. It generates a list of 28 strings, each of length 7, using the 26 English alphabets, dot and space characters. All the 26 alphabets of English, dot and space characters are mapped to these generated 28 strings. To embed a secret character, the string that is mapped to the respective character is identified. The cover document is then scanned, serially, for the occurrence of any character in the identified string. The position of the occurred cover character, in the identified string, decides its character spacing. Since a single secret character can be embedded in any of the 7 characters of the identified string, this technique automatically handles the limitation of the non-uniform occurrence of characters in an English text.

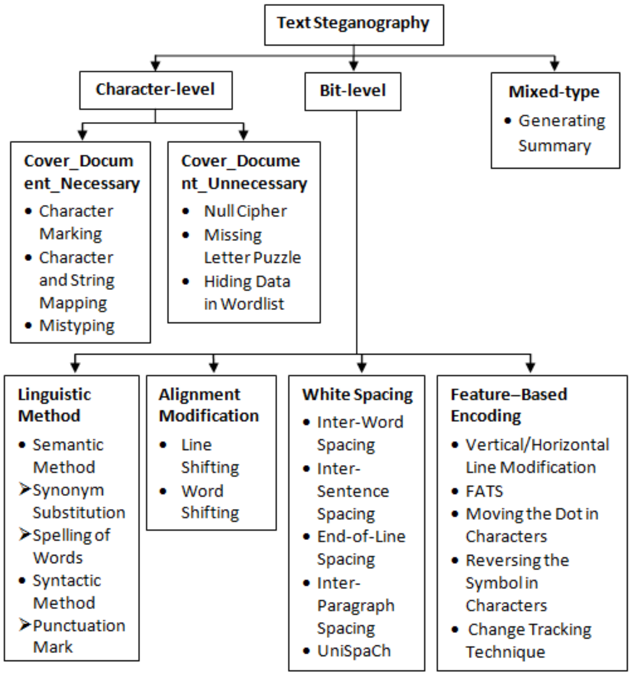


Fig. 2 Classification of text steganography

1.1.3 Mistyping

This technique embeds the secret message by intentionally creating some spelling mistakes [8] [11] [12] or by altering the position of characters (placing the characters slightly over or under the baseline[1]) in a text document [7]. Since this type of typing errors is very common in a text document, this method of concealing the secret won't attract any third party unless they are aware of the secret communication. However, as too many mistakes would raise suspicion, this method could be used only when the amount of information to be hidden is very small and the size of the used cover document is very large. The hidden secret will be exposed when the original characters of the misspelled word or the misplaced characters are identified and merged together. An advanced method of the misspelling technique, carefully mistypes the words such that the misspelled word also exists in the dictionary [12].

Eg:

Original Text: He jumped from the boat.

After embedding character "j": He dumped from the boat.

1.2 Cover_Document_Unnecessary [4] [5]

Instead of relying on a cover work to embed the secret, this method directly generates the stego work based on the secret. Hence there is no guarantee that the generated stego work, containing words or sentences, would be related.

1.2.1 Null Cipher [8] [13] [14]

This technique generates sentences in such a way that the particular position of a letter from each word (say second letter from each word) or sentence or paragraph or page carries the secret message. This method is very complicated and requires an experienced person to perform the task.

Eg: *Apparently neutral's protest is thoroughly discounted and ignored. Islam hard hit. Blockade issue affects pretext for embargo on by-products, ejecting suets and vegetable oils.* (From [14])

Embedded Secret: Pershing sails from NY June 1.

1.2.2 Missing Letter Puzzle [3]

This technique generates a list of words, of length between 6 and 15, as a stego work. The words are generated with respect to the decimal value of the secret character to be hidden. Then, in order to disguise the stego work as a puzzle, one or more characters in each generated word are substituted by question mark. The length of each word along with the position and number of question marks together represents the character hidden in it.

1.2.3 Hiding Data in Wordlist [3]

Similar to Missing Letter Puzzle, this technique also generates a list of words of length between 6 and 15. But this method does not use special characters like question mark to represent the secret. Instead the length and first character of each generated word, that are decided by the decimal value of each secret character, itself represents it.

2. Bit-level Embedding [4] [5]

This method first converts the secret message into bits, by encrypting or using binary representation of ASCII value or using some other representation, and then embeds it.

2.1 Linguistic Method

This method embeds the secrets by altering the cover document without changing its original meaning. This method can further be divided into two categories as semantic and syntactic methods.

2.1.1 Semantic Method

This method replaces one word with another, retaining the original meaning of the used cover document.

[1] In a text document, all the characters of a line are placed over a straight horizontal line knows as the baseline [32]

978-1-5090-4740-6/17/$31.00 ©2017 IEEE

2.1.1.1 Synonym Substitution [15]

In this method, synonyms of words from dictionary are used to encode the secret message (refer Table I). As a substitute for a word, usage of the first synonym represents "0" whereas the second synonym represents "1". It should be noted that both the sender and receiver must have the complete list of words and their respective synonyms for encoding and decoding purpose.

TABLE I SAMPLE WORDS AND THE RESPECTIVE SYNONYMS

Words	Synonyms
Leave	Depart, Go away
Subsequent	Successive, Later
Port	Harbour, Dock
Consequence	Result, Effect

2.1.1.2 Spelling of Words [16]

The spellings of certain words differ from one country to another (refer Table II). For example, representing a word using the UK spelling would encode "0" whereas the US spelling would encode "1". Thus, the stego work produced by this method will contain a mixture of two different word spellings. This simple clue might create suspicion for the third party.

TABLE II SAMPLE US AND UK SPELLING OF WORDS [17]

UK Spelling	US Spelling
Ageing	Aging
Colour	Color
Colonise	Colonize
Computerise	Computerize

2.1.2 Syntactic Method

This method changes the structure of the cover document without changing its meaning.

2.1.2.1 Punctuation Mark [15]

In English language, there exist some sentences where the occurrence of the punctuation mark, like comma, becomes optional. This method explores this flexibility to embed the secret bits. Their presence encodes '0' and the vice-versa encodes '1'. Eg: "Milk, Bread, and Butter" and "Milk, Bread and Butter" both convey the same.

2.2 Alignment Modification

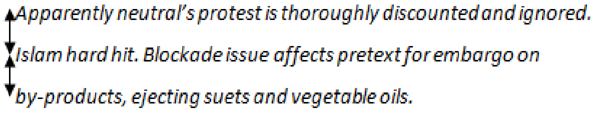
As the name implies, this method alters the alignment of text in the cover document to embed the secret bits.

2.2.1 Line Shifting [7] [18]

This technique shifts a line up or down, by some inches, to embed the secret bits. Three consecutive lines are considered together as a group and a line is marked only if all the lines in the considered group are sufficiently long. In each group, out of these 3 lines, the middle line is alone moved up to encode the bit "0" or down to encode the bit "1". The other two neighbor lines are considered as the control lines and are left undisturbed. During the decoding process, these control lines are used to check whether the middle line has been shifted or not.

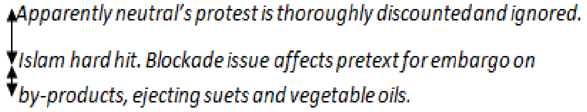
Eg:

Original text (From [14]) *before embedding:*



Apparently neutral's protest is thoroughly discounted and ignored.
Islam hard hit. Blockade issue affects pretext for embargo on
by-products, ejecting suets and vegetable oils.

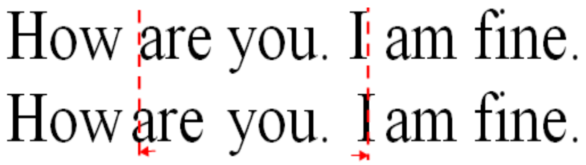
After embedding the bit "1":



Apparently neutral's protest is thoroughly discounted and ignored.
Islam hard hit. Blockade issue affects pretext for embargo on
by-products, ejecting suets and vegetable oils.

2.2.2 Word Shifting [7] [18]

This method is similar to the line shifting technique mentioned above. One difference between these two techniques is that, instead of considering the lines, this method uses words to embed the secrets. The words in each line are divided into groups each consisting of three words. Keeping the first and last word in each group constant, the middle word is alone moved left to encode the bit "0" or right to encode the bit "1" (refer Fig. 3). The document is justified so that the marked lines won't create any suspicion.

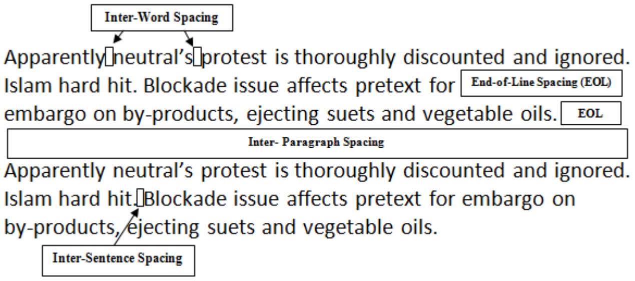


How are you. I am fine.
How are you. I am fine.

Fig. 3 Example for word shifting

2.3 White Spacing

This method adds extra white spaces in between words, paragraphs and sentences or at the end of the lines to embed the secrets (refer Fig. 4). The main problem with these techniques is that some word processing software automatically removes the extra space that is present in the document.



Apparently neutral's protest is thoroughly discounted and ignored.
Islam hard hit. Blockade issue affects pretext for
embargo on by-products, ejecting suets and vegetable oils.

Apparently neutral's protest is thoroughly discounted and ignored.
Islam hard hit. Blockade issue affects pretext for embargo on
by-products, ejecting suets and vegetable oils.

Fig. 4 Illustration of the white spacing's in text document [14]

2.3.1 Inter-Word Spacing [19] [20] [21]

This method uses white spaces between words to encode the secret message. The extra presence of a space encodes

978-1-5090-4740-6/17/$31.00 ©2017 IEEE

"1" and their absence encodes "0". However, in a justified text, all the inter-word space cannot be interpreted as data. Hence, in such justified texts, to differentiate the ordinary space between words from intentionally created ones, Manchester-like Encoding[2] is used. "01" encodes bit "1", "10" encodes bit "0" whereas "00" and "11" are considered as dummies.

Eg: Hai·how··are·you (For understanding purpose, the white space in the text is highlighted using the character "•")

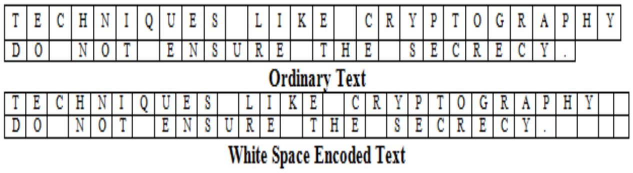
Embedded Secret: 010

2.3.2 *Inter-Sentence Spacing* [21]

The white space between two sentences could be used to encode the bit "0" or "1". The presence of an extra space represents "1" and their absence represents "0". Since the number of sentences in a normal text document is believed to be less, this method cannot be used to hide more data. However, this method is far more superior when compared with the line shifting method. This is mainly because the latter method requires minimum three lines to encode one bit of information.

2.3.3 *End-of-Line Spacing* [21]

This method encodes the secret message by inserting white space or tabs at the end of each line (refer Fig. 5). Since nobody cares about the extra white space at the end of a line, this method might not create any suspicion to the third party. This method can encode much amount of data and uses two spaces to encode one bit (either "0" or "1"), four spaces to encode 2 bits (00, 01, 10, 11), and so on.



TECHNIQUES LIKE CRYPTOGRAPHY
DO NOT ENSURE THE SECRECY.

Ordinary Text

TECHNIQUES LIKE CRYPTOGRAPHY
DO NOT ENSURE THE SECRECY.

White Space Encoded Text

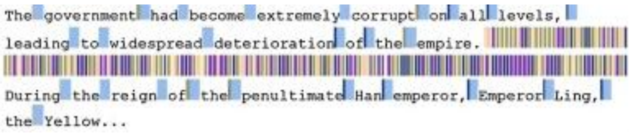
Fig. 5 Example for End-of-Line spacing [21]

2.3.4 *Inter-Paragraph Spacing* [22]

The space between two paragraphs can be used to encode the bits secretly. Injecting space or tab characters in these inter-paragraph spaces will go unnoticed.

2.3.5 *UniSpaCh* [14]

This technique is an improved version of the white spacing techniques that are mentioned above. To embed the bits secretly, this technique inserts characters like Punctuation, Thin, En Quad, Em Quad, Figure, three-per-Em, Six-per-Em, and Hair in inter-sentence, inter-word, inter-paragraph and end-of-line spacing's (refer Fig. 6). The advantage of these space characters over the ordinary white space is that the width of these characters is too small. Hence more space characters can be injected subsequently increasing the amount of information that can be hidden in the cover document.



The government had become extremely corrupt on all levels, leading to widespread deterioration of the empire.

During the reign of the penultimate Han emperor, Emperor Ling, the Yellow...

Fig. 6 Unicode Space Characters (color-coded for understanding purpose) [14]

2.4 *Feature-Based Encoding*

This method explores the features of characters or document to embed the secret bits. Some techniques that are mentioned under this method are restricted to the documents of a particular language.

2.4.1 *Vertical/Horizontal Line Modification* [23]

The vertical/horizontal lines of characters (b, d, h, k, l, t, etc) are modified (expand or shrink) to encode the bits "0" or "1" respectively.

2.4.2 *FATS* [5]

Using the vector representation, this method converts an image into its respective code. The codes are then embedded inside the font attributes like *character spacing*, *kerning* and *color*. One notable advantage of this method is that the embedded codes reflect the original structure of the hidden image. Hence, even when some of the embedded codes are corrupted or destroyed, the original image can still be retrieved.

2.4.3 *Moving the Dot in Characters* [23]

Similar to the lower-case letters of the English alphabets "i" and "j", Arabic and Persian alphabets also have dots. In Persian language, out of the 32 alphabets 18 have dots and in Arabic language, out of the 28 alphabets 15 have dots. These dots are moved up or left undisturbed to embed the bits secretly (refer Fig. 7). Besides, the points and extensions in the Arabic/Persian characters are also exploited.

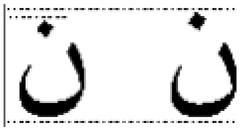


Fig. 7 Vertical displacement of dot in the Persian character Noon [23]

2.4.4 *Reversing the Symbol in Characters* [15]

Arabic language use Araabs like Fatha, Kasra and Damma. Fatha is a slash like symbol written over the character, kasra is a slash like symbol written below the character and damma is a number nine like symbol written over the character [15]. These Araabs are present in every character of the Arabic language of which Fatha is reversed to encode the bits "0" and "1" (refer Fig. 8).

2.4.5 *Change Tracking Technique* [24]

Some document formats like Microsoft Word facilitates the user to keep track of the modifications that are

[2] In Computer Networks, Manchester Encoding [31] is used to encode the binary bit as well as for synchronization purpose. The middle transition from "0" to "1" and "1" to "0" is used for clock synchronization as well as to encode the bits "1" and "0" respectively

performed in the document. This technique explores this feature to embed the bits secretly. First, the sender purposefully inserts some mistakes in a chosen text document. Then with the help of the available commenting tools, the sender corrects the mistakes by himself and sends it to the receiver. Using the above-mentioned facility, the receiver extracts the hidden bits by recognizing the deliberately created mistakes.

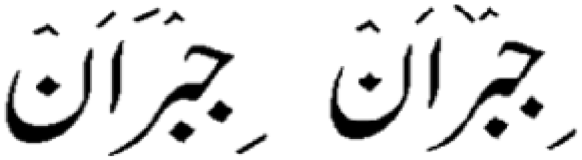


Fig. 8 Usage of the regular and reverse Fatha [15]

3. *Mixed-type Embedding* [4] [5]

This method is a mixture of the character-level and bit-level embedding techniques. Like bit-level embedding, it first converts the secret message into its equivalent binary bits. It then maps these bits to English alphabets using some property. Finally, the mapped alphabet is embedded inside the cover work as similar to the character-level embedding.

3.1 Generating Summary [25]

This technique embeds the secret bits, as pairs, by generating the summary of the used cover work. It partitions the 26 English alphabets into 4 groups using the vertical and horizontal reflection symmetry property. These groups are then mapped to the 4 possible bit pairs. Depending on each bit pair to be embedded, sentences whose first alphabet (not an article) that matches with any one of the alphabets of the respective group are selected to generate a summary.

III. Evaluation

The comparison of the existing methods is provided in Table III [4] [5]. The methods are compared based on their embedding capacity. Embedding capacity is nothing but the amount of information that can be hidden in a chosen cover medium [26]. For calculation purpose, it has been considered that the average length of a word is 4.50 (not including space) [27], the average number of characters in a line is 60 (including space) [28], the average number of words in a sentence (should be between 15 and 20 [29]) is 15, and the number of sentences per paragraph (150 words [30]) is 10 for English language [4] [5]. From Table III, it can be observed that the method Character and String Mapping stands best as it requires the least number of cover characters to embed a secret character.

IV. Conclusion

Digital steganography, which is inspired by the ancient secret communication techniques like tattooed slaves, invisible inks, micro dots, wax tablets, semagrams, etc, is the art of hiding a secret message inside a cover medium in an unnoticeable manner. Due to the recent evolvement of the digital communication, steganography has got a new paradigm with the help of the digital cover medium like text, image, audio, video, etc. Although media types other than text can be used as a cover medium, organizations might prefer text documents as they are widely used in such environment. Considering these facts, this paper highlighted both the pitfalls and importance of text steganography, in addition to introducing the various existing techniques and its respective classification. A comparison of the existing techniques, based on their embedding capacity, illustrated that the character-level embedding technique Character and String Mapping stands best as it requires the least number of cover characters to embed a secret character.

TABLE III Comparison of the Existing Techniques [4] [5]

Technique	Type of Embedding	Embedding Capacity (Approximate)
Character Marking [4] [7] [8] [9] [10], Mistyping [7] [8] [11] [12]	Character-level	Variable (due to the non-uniform occurrence)
Character and String Mapping [4]	Character-level	2-bits/cover-character
Missing Letter Puzzle [3]	Character-level	8-bits/10.5-cc
Hiding Data in Wordlist [3]	Character-level	8-bits/10.5-cc
Synonym Substitution [15]	Bit-level	1-bit/4.5-cc (best case)
Spelling of Words [16]	Bit-level	1-bit/4.5-cc (best case)
Line Shifting [7] [18]	Bit-level	1-bit/180-cc
Word Shifting [7] [18]	Bit-level	1-bit/15.5-cc
Inter-Word Spacing [19] [20] [21]	Bit-level	1-bit/10-cc
Inter-Sentence Spacing [21]	Bit-level	1-bit/166-cc
End-of-Line Spacing [21]	Bit-level (2-bits)	2-bits/60-cc
UniSpaCh [14]	Bit-level (2-bits)	1.046-bits/cover-character
Vertical/Horizontal Line Modification [23]	Bit-level	Variable (due to the non-uniform occurrence)
Change Tracking Technique [24]	Bit-level	0.33-bits/4.5-cc
Generating Summary [25]	Mixed-type (2-bits)	2-bits/82.5-cc

where, cc – cover-characters

References

[1] N.F. Johnson and S. Jajodia, "Exploring steganography: seeing the unseen," IEEE Computer, vol. 31, no. 2, pp. 26-34, 1998.

[2] Zhi-Hui Wang, Chin-Chen Chang, Chia-Chen Lin, and Ming-Chu Li, "A reversible information hiding scheme using left–right and up–down chinese," The Journal of Systems and Software, pp. 1362–1369,

2009.

[3] M Agarwal, "Text steganographic approaches: a comparison," International Journal of Network Security & its Applications, vol. 5, pp. 91-106, 2013.

[4] R.Bala Krishnan, Prasanth Kumar Thandra, and S.A.V. Satya Murty, "Text steganography: a novel character-level embedding algorithm using font attribute," Security and Communication Networks, 2017.

[5] R.Bala Krishnan, Prasanth Kumar Thandra, S.A.V. Satya Murty, and P. Thiyagarajan, "Font arrtibutes based text steganographic algorithm (FATS) for communicating images: a nuclear power plant perspective," Kerntechnik, in press.

[6] Behrouz A Forouzan and Debdeep Mukhopadhyay, Cryptography And Network Security.: Tata McGraw-Hill Education, 2011.

[7] Jack T. Brassil, Steven Low, and Nicholas maxemchuk F., "Copyright protection for the electronic distribution of text documents," Proceedings of the IEEE, vol. 87, no. 7, pp. 1181-1196, 1999.

[8] Abdelrahman Desoky and Mohamed Younis, "Chestega: chess steganography methodology," Security and Communication Networks, vol. 2, no. 6, pp. 555–566, 2009.

[9] Sabu M. Thampi, "Information hiding techniques: a tutorial review," in ISTE-STTP on Network Security & Cryptography, LBSCE, 2004.

[10] Fabien A. P. Petitcolas, Ross J. Anderson, and Markus G. Kuhn, "Information hiding—a survey," Proceedings of the IEEE, pp. 1062–1078, 1999.

[11] Krista Bennett, "Linguistic steganography: survey, analysis, and robustness concerns for hiding information in text," Purdue University, Technical 2004-13.

[12] Mercan Topkara, Umut Topkara, and Mikhail J. Atallah, "Information hiding through errors: a confusing approach," in Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents IX, vol. 6505, 2007.

[13] David Kahn, The Code-Breakers: The Comprehensive History of Secret Communication from Ancient Times to the Internet, 2nd ed.: Scribner, 1996.

[14] Lip Yee Por, KosSheik Wong, and Kok Onn Chee, "UniSpaCh: a text-based data hiding method using unicode space characters," The Journal of Systems and Software, pp. 1075-1082, 2012.

[15] Jibran Ahmed Memon, Kamran Khowaja, and Hameedullah Kazi, "Evaluation of steganography for Urdu/Arabic text," Journal of Theoretical and Applied Information Technology, pp. 232-237, 2005.

[16] Mohammad Shirali-Shahreza, "Text steganography by changing words spelling," in 10th International Conference on Advanced Communication Technology (ICACT 2008), 2008.

[17] [Online]. http://www.tysto.com/uk-us-spelling-list.html [Last accessed on 3.3.2017]

[18] Jack T. Brassil, Steven Low, and Nicholas F. Maxemchuk, "Electronic marking and identification techniques to discourage document copying," IEEE Journal on Selected Areas in Communications, vol. 13, no. 8, pp. 1495-1504, 1995.

[19] Souvik Bhattacharyya, Indradip Banerjee, and Gautam Sanyal, "A novel approach of secure text based steganography model using word mapping method (WMM)," International Journal of Computer and Information Engineering, pp. 96-103, 2010.

[20] Prem Singh, Rajat Chaudhary, and Ambika Agarwal, "A novel approach of text steganography based on null spaces," IOSR Journal of Computer Engineering, vol. 3, no. 4, pp. 11-17, 2014.

[21] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding," IBM Systems Journal, vol. 35, no. 384, pp. 313-336, 1996.

[22] L. Y. POR and B. Delina, "Information hiding: a new approach in text steganography," in 7th WSEAS International Conference on Applied Computer & Applied Computational Science (ACACOS '08), pp. 689-695, 2008.

[23] M. Hassan Shirali-Shahreza and Mohammad Shirali-Shahreza, "A new approach to Persian/Arabic text steganography," in 5th IEEE/ACIS International Conference on Computer and Information Science, 2006.

[24] Tsung-Yuan Liu and Wen-Hsiang Tsai, "A new steganographic method for data hiding in microsoft word documents by a change tracking technique," IEEE Transactions on Information Forensics and Security, vol. 2, pp. 24-30, 2007.

[25] A. Majumder and S. Changder, "A novel approach for text steganography: generating text summary using reflection symmetry," in International Conference on Computational Intelligence: Modeling Techniques and Application, pp. 112-120, 2013.

[26] Esra Satir and Hakan Isik, "A compression-based text steganography method," Journal of Systems and Software, vol. 85, no. 10, pp. 2385–2394, 2012.

[27] John R. Pierce, An Introduction to Information Theory: Symbols, Signals and Noise, 2nd ed.: Dover Publications, 1980.

[28] Isabelle de Ridder, Reading from the Screen in a Second Language: Empirical Studies on the Effect of Marked Hyperlinks on Incidental Vocabulary Learning, Text Comprehension and the Reading Process.: Garant, 2003.

[29] Martin Cutts, Oxford Guide to Plain English.: Oxford University Press, 2013.

[30] Janice R. Matthews, John M. Bowen, and Robert W. Matthews, Successful Scientific Writing Full Canadian Binding: A Step-by-Step Guide for the Biological and Medical Sciences.: Cambridge University Press, 2000.

[31] Curt White, Data Communications and Computer Networks: A Business User's Approach, 8th ed.: Cengage Learning, 2015.

[32] [Online]. https://en.wikipedia.org/wiki/Baseline_(typography). [Last accessed on 3.3.2017]

978-1-5090-4740-6/17/$31.00 ©2017 IEEE