Jurnal Mantik
Volume 4, Number 3, Nov 2020, pp. 1676-1687 E-ISSN 2685-4236
https://iocscience.org/ejournal/index.php/mantik

Analysis of T-Code Compression Methods to Increase Capacity of SSCE Method

Marzuki Leo[1], Charles Halim[2], Iola[3], Jeremy Valtino[4], Insidini Fawwaz[5]

[12345]Faculty of Technology and Computer Science
[12345]Universitas Prima Indonesia, Indonesia, Jl. Sekip simp. Sikambing, Medan

E-mail: marzukileo2000@gmail.com

ARTICLE INFO

Article history:
Received: 12/07/2020
Revised: 22/08/2020
Accepted: 30/09/2020

Keywords:
steganography, text documents, SSCE method, data compression, T-Code method

ABSTRACT

The process of hiding classified information can take advantage of the steganography method. However, media such as images, audio or video is not efficient, due to the large size of the media, making text media a great solution for hiding confidential information. In 2011, a new steganographic method was introduced that uses special code generation based on the use of unlimited articles or on conjunctions with nouns that are not specific or specific to English. A new code representation method Secret Steganography Code for Embedding (SSCE) was also used to increase security. However, there is a problem with the application of this text steganography method, namely that the secret message has not been compressed, where if the cover text is smaller than the secret message, the secret message cannot be inserted into the cover text. To solve this problem, a text compression method can be applied. One of the compression methods that can be applied is the T-Code text compression method. The T-Code method uses the concept of iterative code construction and self-synchronization to compress and decompress text. From the testing process, information was obtained that every change of letter or word in the position where the secret message bit is attached will cause the secret message that is extracted to become chaotic and the T-Code compression method can increase the capacity of the cover document by 10.61%.

1. Introduction

During the development of computer and communication technology, the process of sending electronic messages is prone to wiretapping. This causes the transmission of confidential information using computer technology to be secured before sending.

Steganography is an art and technique that will be used to insert messages into a medium. This steganography technique is very different from cryptography, which only scrambles messages so that they cannot be understood, but third parties can detect the presence of data (ciphertext), because the results of cryptography themselves are data that are different from the original form and as if the data is messy, but can be returned. into shape. Meanwhile, steganography discusses how a message can be inserted into a media, both images, audio and video, so that third parties cannot realize this because steganography techniques take advantage of the limitations of the human sensory system such as the eyes and ears. One type of steganography method that can be used is the text steganography method (Darwis and Kisworo, 2017).

One of the text steganography methods that can be used is a new steganography method that uses special code generation and the new Secret Steganography Code for Embedding (SSCE) code representation method. The process begins by encoding each character of the secret message using the SSCE value. The encoded result will be pasted into the cover text. This method is an integrated approach of secret code generator with text steganography method. Combining the two approaches to the paste algorithm will increase the security of the pasting of secret messages (Banerjee, Bhattacharyya and Sanyal, 2011).

However, there is a problem with the application of this text steganography method, namely that the secret message has not been compressed, where if the cover text is smaller than the secret message, the secret message cannot be inserted into the cover text. To solve this problem, a text compression method can be

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

applied. The compression process aims to reduce the size of the data by considering the quality of the data which is still sufficient to be enjoyed. One of the compression methods that can be applied is the T-Code text compression method. The T-Code method uses the concept of iterative code construction and self-synchronization to compress and decompress text.

Based on the description above, it is known that this text steganography method is able to increase the security of the embedding of secret files and this method is still relatively new. Meanwhile, to increase the capacity of the cover text, the T-Code text compression method will be applied.

2. Methods

2.1. Steganography

Steganography is the art and science of writing hidden messages or hiding messages in a way so that apart from the sender and the recipient, no one knows or realizes that there is a secret message. The word "steganography" comes from the Greek steganos, which means "hidden or covert", and graphein, "to write" (Ardiansyah and Kurniasih, 2018).

A. Steganography Media

According to Nosrati et al (2011), the type of media used for steganography is divided into three categories, text, image and sound (Figure 1).

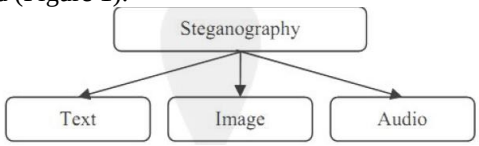

Fig 1. Steganographic media type diagram

B. Text Steganography

Text steganography is a method that uses written natural language to hide a secret message. The advantages of text steganography when compared to using images or audio are the small memory requirements and the easier communication process.

Text steganography can be classified into three types, namely format-based, random and statistical generation and linguistic method, as shown in the following figure:

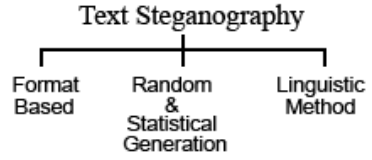

Fig 2. Three Types of Categories of Text Steganography

Format based method uses and changes the format of the cover text to hide data. This method doesn't change any word or sentence, so it doesn't change the value of the cover text. In this method, a blank space can be added behind each word, sentence or paragraph. A single space will be represented as '0' and two consecutive spaces will be represented as '1'. Although only small data can be hidden in a document, this method can be applied to most types of text.

Random and statistical generation methods are used to generate cover texts automatically based on the statistical characteristics of the language. This method uses sample grammar to produce cover text in a specific natural language. Probabilistic context-free grammar (PCFG) is a commonly used language model where each transformation rule of a context-free grammar has a probability.

The linguistic method considers the linguistic features of the text in order to modify it. This method uses the linguistic structure of the message as a place to hide information.

2.2. SSCE Text Steganography Method

This technique will first encrypt a message using the SSCE table and then paste the ciphertext on a cover file by inserting articles a or an with non-specific nouns in English using certain mapping techniques. The position of the attachment is encrypted using the same SSCE table and stored in a different file which will be transmitted to the recipient securely together with the stego file (Monika Agarwal, 2013).

The block diagram of a text steganographic model with a secret key can be seen in the following figure:

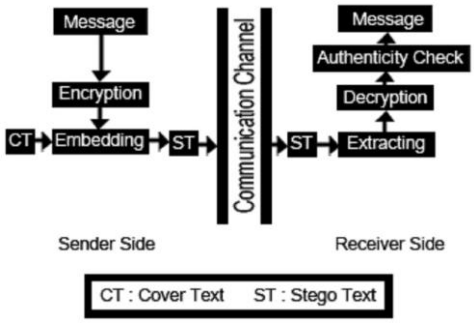

Fig 3. Text Steganography Model

The input message can be of any digital form and will be treated as a string of bits. First, the input message will be encrypted using a new code generation technique (Secret Steganography Code for Embedding / SSCE). This encrypted message generates a secret key. To increase the level of security, SSCE code representations are used to encrypt messages and secret messages will be attached to the cover text by entering the word 'a' or 'an' in the conjunction based on the mapping information in Figure 4.To obtain stego text (stego text). On the receiving end, a reverse operation will be performed to obtain classified information.

Words		Bit Sequence
a	consonant	00
an	vowel	11
a	vowel	10
an	consonant	01

Fig 4. Mapping Techniques

A. Special Code Generation Algorithm for Message Encryption / Decryption

The following details the special code generation algorithm for encryption and decryption of the entered secret message:

a. Select the message and take characters one by one.
b. Convert to ASCII Code form.
c. Convert ASCII code to SSCE value using the following Table 1:

Table 1
Convert ASCII code to SSCE

Secret Steganography Code for Embedding(SSCE) Table

ASCII	SSCE	ASCII	SSCE	ASCII	SSCE	ASCII	SSCE	ASCII	SSCE	ASCII	SSCE	ASCII	SSCE	ASCII	SSCE	ASCII	SSCE	ASCII	SSCE
10	1	1	26	2	52	3	78	4	104	5	130	6	156	7	181	8	206	9	231
20	2	11	27	12	53	13	79	14	105	15	131	16	157	17	182	18	207	19	232
30	3	21	28	22	54	23	80	24	106	25	132	26	158	27	183	28	208	29	233
40	4	31	29	32	55	33	81	34	107	35	133	36	159	37	184	38	209	39	234
50	5	41	30	42	56	43	82	44	108	45	134	46	160	47	185	48	210	49	235
60	6	51	31	52	57	53	83	54	109	55	135	56	161	57	186	58	211	59	236
70	7	61	32	62	58	63	84	64	110	65	136	66	162	67	187	68	212	69	237
80	8	71	33	72	59	73	85	74	111	75	137	76	163	77	188	78	213	79	238
90	9	81	34	82	60	83	86	84	112	85	138	86	164	87	189	88	214	89	239
100	10	91	35	92	61	93	87	94	113	95	139	96	165	97	190	98	215	99	240
110	11	101	36	102	62	103	88	104	114	105	140	106	166	107	191	108	216	109	241
120	12	111	37	112	63	113	89	114	115	115	141	116	167	117	192	118	217	119	242
130	13	121	38	122	64	123	90	124	116	125	142	126	168	127	193	128	218	129	243
140	14	131	39	132	65	133	91	134	117	135	143	136	169	137	194	138	219	139	244
150	15	141	40	142	66	143	92	144	118	145	144	146	170	147	195	148	220	149	245
160	16	151	41	152	67	153	93	154	119	155	145	156	171	157	196	158	221	159	246
170	17	161	42	162	68	163	94	164	120	165	146	166	172	167	197	168	222	169	247
180	18	171	43	172	69	173	95	174	121	175	147	176	173	177	198	178	223	179	248
190	19	181	44	182	70	183	96	184	122	185	148	186	174	187	199	188	224	189	249
200	20	191	45	192	71	193	97	194	123	195	149	196	175	197	200	198	225	199	250
210	21	201	46	202	72	203	98	204	124	205	150	206	176	207	201	208	226	209	251
220	22	211	47	212	73	213	99	214	125	215	151	216	177	217	202	218	227	219	252
230	23	221	48	222	74	223	100	224	126	225	152	226	178	227	203	228	228	229	253
240	24	231	49	232	75	233	101	234	127	235	153	236	179	237	204	238	229	239	254
250	25	241	50	242	76	243	102	244	128	245	154	246	180	247	205	248	230	249	255
		251	51	252	77	253	103	254	129	255	155								

d. Convert back to character form.
e. Convert characters to binary ASCII Code form.

(Indradip Banerjee, et.al., 2011)

B. Special Code Generation Algorithm for Message Embedding

The following details the special code generation algorithm for pasting secret messages on the cover text document:

a) Select a message and encrypt messages with SSCE value.
b) Select a cover text to paste the message. Check whether the encrypted messages could be embedded or not. If not, then reselect the cover text.

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

c) Check the message string and select the first two bits of the message sequence (MSG).
d) Start from the first word of the cover text (TX).
   1) If MSG = '11' then find the word 'an' from TX and check if the first character of the next word is a vowel character.
   2) Otherwise, if MSG = '10' then find the word 'an' from TX and check if the first character of the next word is a vowel character. Change 'an' to 'a'.
   3) Otherwise, if MSG = '01' then find the word 'a' from TX and check if the first character of the next word is a consonant character. Change 'a' to 'an'.
   4) Otherwise, if MSG = '00' then find the word 'a' from TX and check if the first character of the next word is a consonant.
   5) Repeat the above steps for all message bit sequences taking two bits for one run.
e) Save the embedding position in a separate file and encode the file with SSCE values and send it to the recipient separately. The encoding process is done by changing each character in the position file to an ASCII Code value, then converting it to an SSCE value.

C. Special Code Generation Algorithm for Message Extraction

The following details the special code generation algorithm for extracting secret messages from stego text documents:

a) Input the stego text file and the embedding position.
b) Decode the embedding position to ASCII value. The decode process is done by converting each SSCE value into an ASCII value. Then each ASCII value is converted into a positional character.
c) Select the embedding position on the TX.
   1) If there is the word 'an' and the first character of the next word is a vowel character, then MSG = '11'.
   2) Otherwise, if there is the word 'a' and the first character of the next word is a vowel character, then MSG = '10'.
   3) If there is the word 'an' and the first character of the next word is a consonant, then MSG = '01'.
   4) If there is the word 'a' and the first character of the next word is a consonant, then MSG = '00'.
d) Combine all the resulting binary bits.
e) Group into 8 bit sub-blocks and convert them to decimal numbers. The value obtained is the SSCE value.
f) Convert the value obtained to the form of the ASCII value.
g) Convert each ASCII value to character form.

2.3. T-Code Compression Method

The T-Codes algorithm is an augmentation algorithm for variable length codes that have self-synchronization properties developed by M.R. Titchener in 1995. The T-Codes algorithm will encode the input message where the codes that are generated for each character have a different length (variable length). The T-Codes algorithm consists of three stages, namely:

A. Simple-level Simple T-Augmentation

Given a set S $\subset A^+$ and p $\in$ S, augmentation process of S, represented by S_0 could be formulated as follow:

$$S_{(p)} = S \backslash \{p\} \cup pS$$

Symbol p represents prefix.

All non-prefix code-string on S, ie. u $\in$ S\{p}, mark the end of code-word on $S_{(p)}$ as shown in Table 2 below:

Table 2

T-Augmented Construction Determines $S_{(p)}$ Value Where $S = \{S_1, S_2, ...S_n\}$ And $p = S_j$

S	$\rightarrow$	$S_{(p)}$
$S_{(1)}$		$S_{(1)}$
$\vdots$		$\vdots$
$S_{(j-1)}$		$S_{(j-1)}$
$S_{(j)}$		~~$S_{(j)}$~~
$S_{(j+1)}$		$S_{(j+1)}$
$\vdots$		$\vdots$
$S_{(n)}$		$S_{(n)}$
		$S_{(j)}S_{(1)}$

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

S	$\rightarrow$	$S_{(p)}$
		$\vdots$
		$S_{(j)}S_{(j)}$
		$\vdots$
		$S_{(j)}S_{(n)}$

B. Iterative Simple T-Augmentation

An extension of the single step simple T-augmentation process can be obtained by applying equation (4) iteratively, with the result set of each level of T-augmentation taken as input for the next successive T-augmentation.

Given S, result set of n level T-augmentation with prefix p_1, p_2, ..., p_n represented as $S_{(p1,\ p2,\ ...,\ pn)}$. Iterative application of equation (4) implied that:

$$S_{(p_1,p_2,\ldots,p_{i+1})} = (S_{(p_1,p_2,\ldots,p_i)})_{(p_i+1)}$$

For i = 1, 2, ..., n – 1.

C. Self-Synchronization

Table 3 below changes code-words on $S_{(0,\ 1,\ 101)}$ to character in alphabet symbols

Table 3

T-Augmented Construction Determines $S_{(0,1,101)}$ Value Where $S = \{0,1\}$

S		$S_{(0)}$		$S_{(0,1)}$		$S_{(0,1,101)}$	symbol
0		0					
1	$\rightarrow$	1		1			
		00	$\rightarrow$	00	$\rightarrow$	00	a
		01	$\rightarrow$	01	$\rightarrow$	01	e
				11	$\rightarrow$	11	i
				100	$\rightarrow$	100	b
				101	$\rightarrow$	101	
						10100	c
						10101	d
						10111	f
						101100	g
						101101	H

The encoding of a message can be obtained by substituting code words for letter symbols. The decoding process through a prefix-free code can be implemented with a decoding tree.

3. Metode Penelitian

The following image shows a use case diagram of the application being made.

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

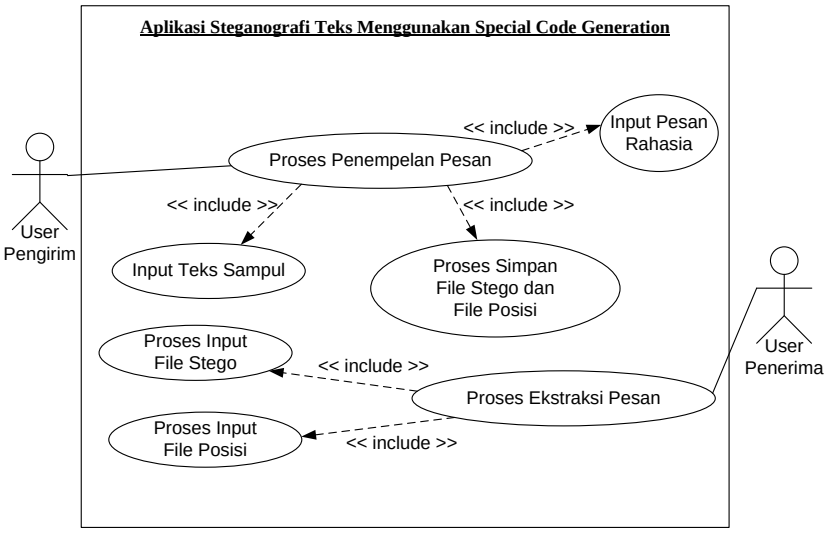

Fig 5. Use Case Diagram of a Text Steganography Application Using Special Code Generation

The work process of the system can be divided into 2 parts, namely the message pasting process and the message extraction process.

3.1 Process of Embedding Message using the Special Code Generation Method

The process of embedding a secret message to the cover document starts from the encryption process of the secret message using the SSCE method. The encryption result from the SSCE method will be embedded to the cover document using algorithmic provisions. The work procedure of the message embedding process using the Special Code Generation method can be seen in the following figure:

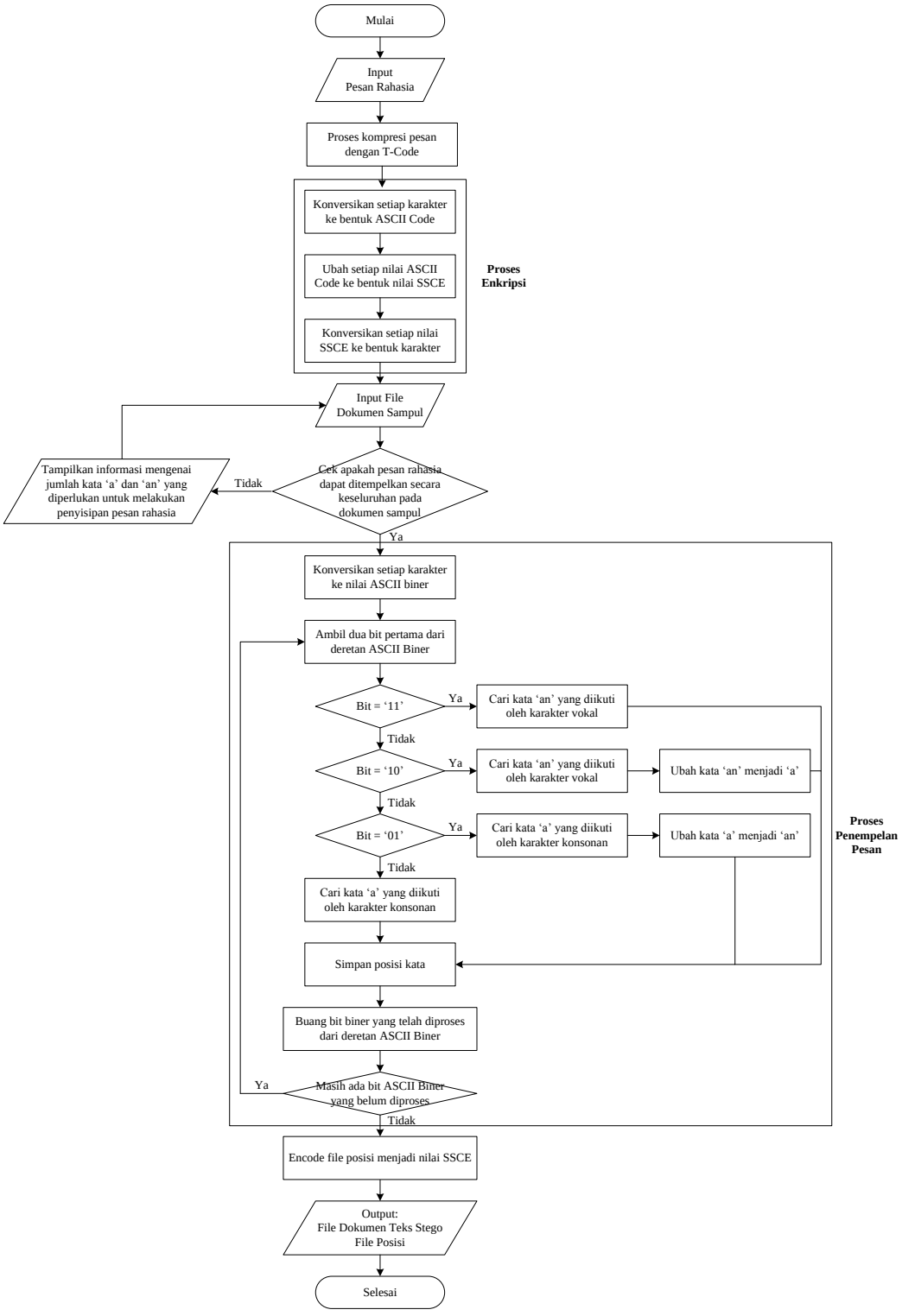

Fig 6. Flowchart of Message Embedding Process with Special Code Generation Method

3.2 Message Extraction Process with Special Code Generation Method

The process of extracting secret messages from the stego document will begin with the extraction of secret bits from the stego document. Then the set of bits will be grouped into 8 bits and converted into decimal values. This series of decimal values will be decrypted using the SSCE method. The results of the decryption value obtained will be converted into characters so that the original secret message is obtained. The work procedure of the message extraction process using the Special Code Generation method can be seen in the following figure:

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

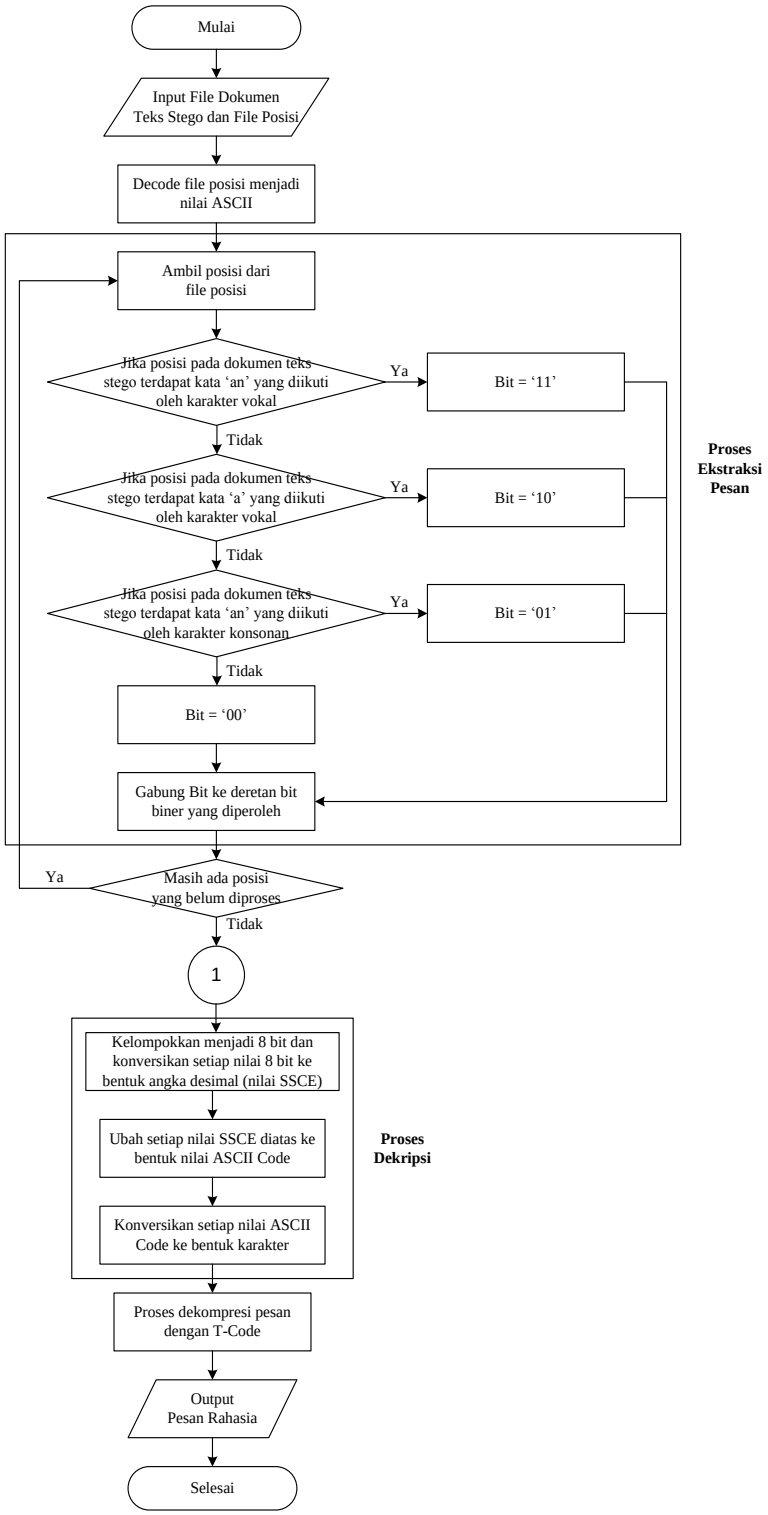

Fig 7. Flowchart of Message Extraction Process with Special Code Generation Method

3.3 Testing

Test cases that can be done include:

A. Omitting a word 'a' at the beginning of a paragraph.

This test is done by deleting the word 'a' which was first discovered, as shown in the following illustration:

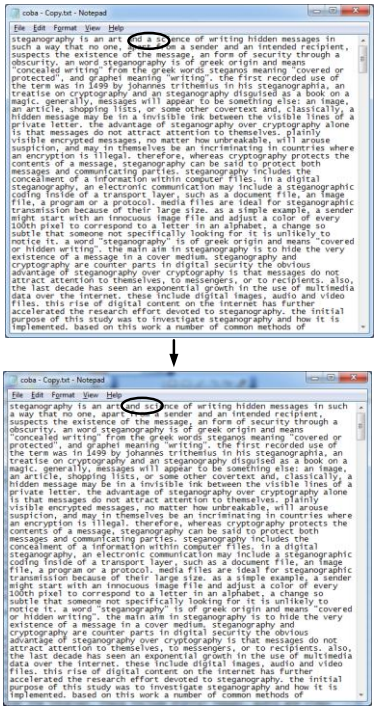

Fig 8. Display of Word Deletion Process

The results of the extraction process can be seen in the following figure:

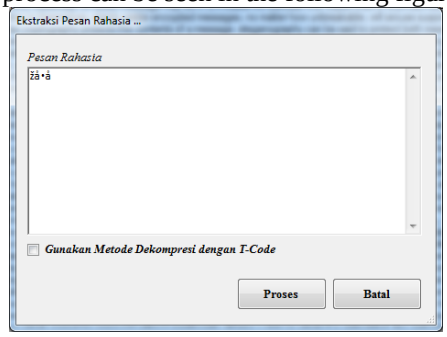

Fig 9. Extraction Process View for First Test

B. Omitting a word 'a' at the end of a paragraph.

This test is done by deleting the word 'a' at the end of the paragraph, as shown in the following illustration:

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

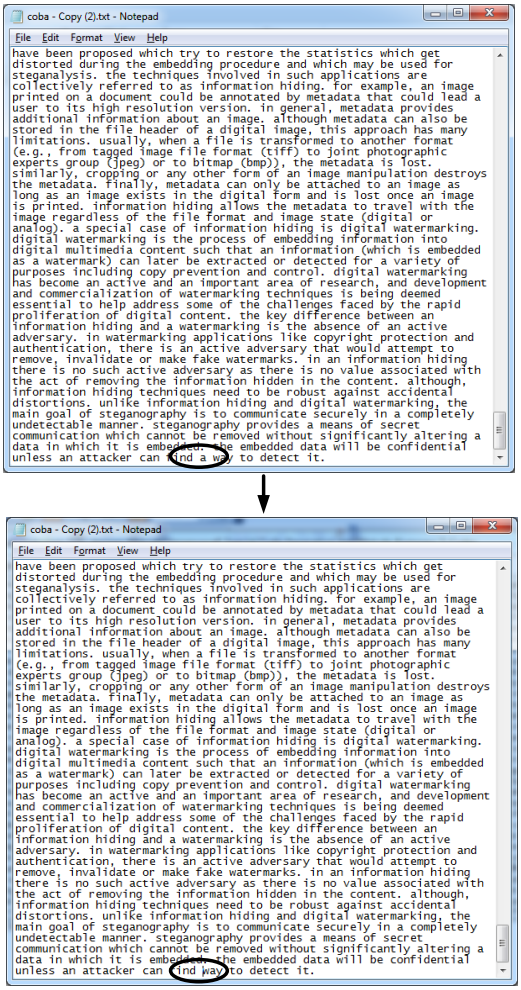

Fig 10. Display of Word Deletion Process at the End of Paragraph

The results of the extraction process can be seen in the following figure:

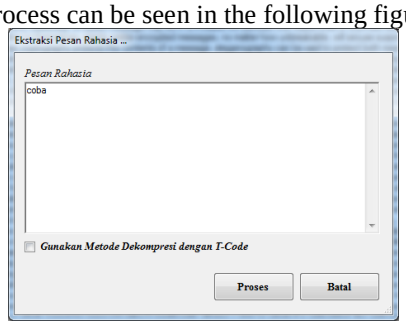

Fig 11. Extraction Process View for Second Test

C. Change the word 'an' to 'a'.

This test is done by changing the word 'an' to 'a' which is at the beginning of the paragraph, as shown in the following illustration:

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

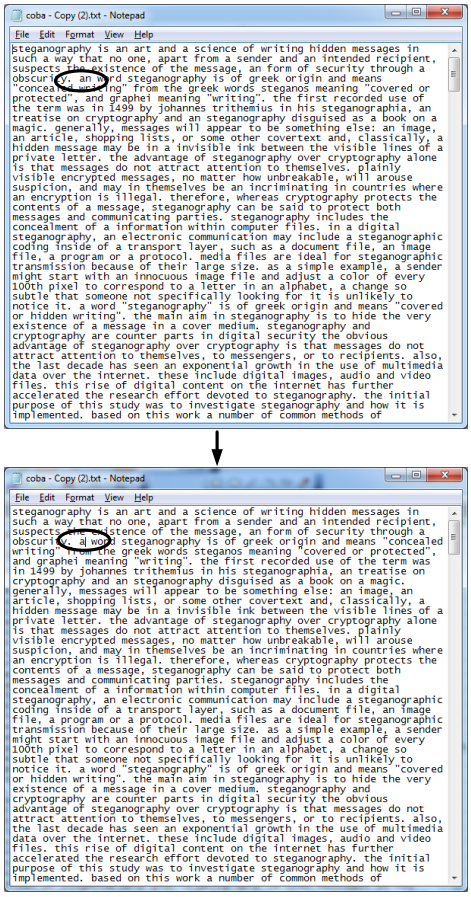

Fig 12. Display of the Word Change Process at the Beginning of a Paragraph

The results of the extraction process can be seen in the following figure:

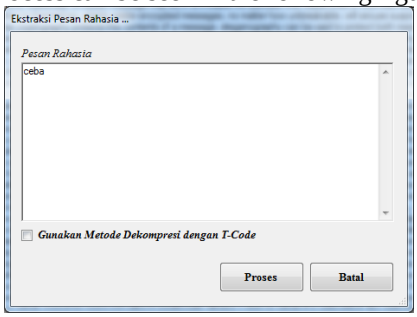

Fig 13. Extraction Process View for Third Test

From the test results above, the following information can be obtained:

a) The process of deleting words at the beginning of a paragraph will cause secret messages to fail to extract out.
b) The process of deleting words at the end of paragraphs will not cause changes to the secret message if the secret message is not too long.
c) The process of changing words at the beginning of a paragraph will cause the secret message to be changed.

Another test was carried out to test the additional capacity before and after the use of the T-Code compression method in the SSCE method. The details of the test results obtained are as follows Table 4.

Tabel 4
Details of The Test Results

Cover Text Size	Without T-Code compression method		With T-Code compression method	
	Total bit	Total characters	Total bit	Total characters
1 KB	30 bit	3	32 bit	4
2 KB	60 bit	7	64 bit	8
5 KB	98 bit	12	108 bit	13

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

Cover Text Size	Without T-Code compression method		With T-Code compression method	
	Total bit	Total characters	Total bit	Total characters
10 KB	168 bit	21	182 bit	22
15 KB	244 bit	30	274 bit	34
20 KB	312 bit	39	348 bit	43
26 KB	370 bit	46	410 bit	51

Total bit without T-Code compression method: 1282 bit.
Total bit with T-Code compression method: 1418 bit.
Difference of bit = 1418 – 1282 = 136 bit
Mean of increasing capacity:
$^{136}/_{1282}$ * 100 % = 10.61 %

4. Conclusions

After completing the analysis of the T-Code compression method to increase the capacity of the SSCE method in hiding this secret file, the authors can draw the following conclusions:

A. Any change of letters or words in the position where the secret message bit is attached will cause the extracted secret message to become messy.
B. The T-Code compression method can increase the embedding capacity of cover documents by 10.61%.

5. References

[1] Achmad Ardiansyah dan Mepa Kurniasih, (2018). Penyembunyian Pesan Rahasia Pada Citra Digital Dengan Teknik Steganograpi Menggunakan Metode Least Significant Bit, Vol. XIII Nomor 3 November 2018 – Jurnal Teknologi Informasi, ISSN: 1907-2430.
[2] Bhattacharyya, S., I. Banerjee dan G. Sanyal (2010). *A Novel Approach of Secure Text Based Steganography Model using Word Mapping Method (WMM)*, International Journal of Computer and Information Engineering 4:2.
[3] Endang Ratnawati Djuwitaningrum, Melisa Apriyani, (2016). Teknik Steganografi Pesan Teks Menggunakan Metode Least Significant Bit dan Algoritma Linear Congruential Generator (Text Message Steganography Using Least Significant Bit Method and Linear Congruential Generator Algorithm), JUITA ISSN: 2086-9398 Vol. IV Nomor 2.
[4] Frida Effelyanti Naibaho, (2020). Implementasi Algoritma J-Bit Encoding Pada Kompresi File PDF, Jurnal Sistem Komputer dan Informatika (JSON) Volume 1, Nomor 3, Mei 2020, e-ISSN 2548-8368, DOI 10.30865/json.v1i3.2153.
[5] Indradip Banerjee, Souvik Bhattacharyya, Gautam Sanyal (2011). *Novel Text Steganography through Special Code Generation*, International Conference on Systemics, Cybernetics and Informatics.
[6] Kibbee D.Streetman (2010). *Steganography, Art of Covert Communications*.
[7] Mohammed Al-Mualla and Prof. Hussain Al-Ahmad, *Information Hiding: Steganography and Watermarking*. [Online]. Available: http://www.emirates.org/ieee/information_hiding.pdf.
[8] Sofia Saidah, Nur Ibrahim, Mochammad Haldi Widianto, (2019). Pengamanan Pesan pada Steganografi Citra dengan Teknik Penyisipan Spread Spectrum, ELKOMIKA, ISSN (p): 2338-8323, ISSN (e): 2459-9638.
[9] William Steven, Viki Afriyandi, Kristien Margi Suryaningrum, (2019). Implementasi Algoritma Ezstego Untuk Menyembunyikan Pesan Terenkripsi Dengan Playfair Cipher Pada Citra GIF, Jurnal Teknologi Informasi, Vol. 5, No. 2, Desember 2019, E-ISSN 2623-1700.
[10] Souvik Bhattacharyya, Indradip Banerjee and Gautam Sanyal: "Data Hiding Through Multi Level Steganography and SSCE" Journal on "Journal of Global Research in Computer Science, Volume 2, No. 2, February 2011".