ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 7, Issue 12, December 2020

Distributed LSB Embedding on Multiple Cover Image Files for Secured Steganography Communication

[1] Jagan Raj Jayapandiyan, [2] Kavitha.C, [3]Sakthivel.K
[1] Research Scholar, Department of Computer Science, Periyar University, Salem, Tamil Nadu, India
[2] Assistant Professor, Department of Computer Science, Thiruvalluvar Govt. Arts College, Rasipuram, Tamil Nadu, India
[3] Professor, Department of Computer Science and Engineering, K. S. Rangasamy College of Technology, Namakkal, Tamil Nadu, India

Abstract: In this proposed research work, an attempt has been made to use multiple image files for steganography encoding along with the capability of secret text recovery in the event of any image corruption during the transit. This algorithm is effective on the security factor of secret image since the embedded checksum will validate for any unauthorized users or intruders attempt to corrupt the picture in any aspect. If any of the stego image underwent any steganalysis or MiM attack, then this proposed algorithm can effectively regenerate the content of one stego image using other intact stego images received in the receiving end.

Keywords:- Steganography, Multi-cover image, secret message recovery

I. INTRODUCTION

From last few years growth in technology of computers and Steganography is a science of concealing a file, document, image or video inside another file, message, image or video, which continues to be an extremely flexible and powerful way to disguise or cover information in plain sight. Using steganography, there are several ways to hide records. The most popular technique is to insert data into digital images. We all know that digital images mean that there are many megabytes of pixel data. It allows space in the digital file for someone to embed steganographic secret data. A good programmer can alter the Least Significant Bits (LSB) of any media file with the use of steganographic applications and embeds a malicious code in the digital picture.

II. STEGANOGRAPHY

The first recorded uses of steganography can be traced back to 440 BC in Greece, when Herodotus mentions two examples in his Histories.[1] Histiaeus sent a message to his vassal, Aristagoras, by shaving the head of his most trusted servant, "marking" the message onto his scalp, then sending him on his way once his hair had regrown, with the instruction, "When thou art come to Miletus, bid Aristagoras shave thy head, and look thereon." Additionally, Demaratus sent a warning about a forthcoming attack to Greece by writing it directly on the wooden backing of a wax tablet before applying its beeswax surface. Wax tablets were in common use then as reusable writing surfaces, sometimes used for shorthand.

The second story [2] also came from Herodotus, which claims that a soldier named Demeratus needed to send a message to Sparta that Xerxes intended to invade Greece. Back then, the writing medium was text written on wax-covered tablets. Demeratus removed the wax from the tablet, wrote the secret message on the underlying wood, recovered the tablet with wax to make it appear as a blank tablet and finally sent the document without being detected. Romans

used invisible inks, which were based on natural substances such as fruit juices and milk. This was accomplished by heating the hidden text, thus revealing its contents.

III. TYPES OF STEGANOGRAPHY

Based on the type of cover file being used in the steganography technique, various types of steganography methods are as follows.

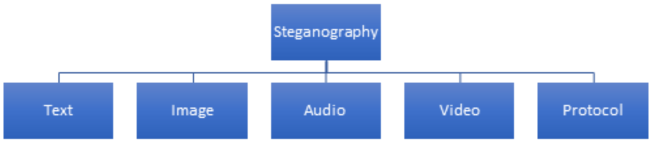

Fig 1. Type of Steganography

Text Steganography

The cover file used in the text steganography method would be in text format and the hidden message contained in the cover file would also be primarily text style. The embedding technique for text steganography is based on the number of characters, white spaces, capital letters, as used in the Morse language code in radio communication.

Image Steganography

Image steganography is a tool used to conceal a hidden message by taking the cover object as the image file. Graphic digital images are widely used as cover source in this steganography, and this cover file helps the user to embed a large volume of bits. The primary advantage of image steganography is that the attention of an intruder is not attracted by the cover image.

Audio Steganography

Audio steganography is a practice used to relay secret information in an imperceptible way by manipulating an audio signal. It is the science of hiding in a host message with any hidden text or audio content. The functions of the host message before steganography and the stego message after steganography are very identical. A more complicated method is embedding hidden messages in optical sound. Varieties of methods have been created for embedding information into digital audio.

Video Steganography

In multiple data hiding technology, video steganography is becoming a significant research field, which has become a promising technique. This is not only for the security necessity of secret message transmission becoming tighter, but video file also has enormous amount of data stream to leverage.

Video steganography is broken down into three groups as per the embedded location of the hidden message: intra-embedding, pre-embedding and post-embedding [3]. Intra-embedding techniques are classified according to the phases of video encoding, such as intra-prediction, motion vectors, interpolation of pixels, coefficients of transformation. On raw footage, pre-embedding methods are manipulated, which can be divided into spatial domains and converted. Post-embedding strategies rely mostly on bitstreams, meaning that the operation of embedding and removing video steganography is all manipulated on the compressed bit stream.

Protocol Steganography

The Protocol Steganography is a modern solution for data hiding, which are popular in recent days. The network layer protocol of the TCP/IP suite is used for data hiding in this steganography and not limited exclusively to network protocols.

For data hiding, covert channels are used in the network layer of the OSI architecture. Covert channels bypass the network system's security protocols. The intention is either used to steal information or use the network protocol to exchange hidden messages over a network. Example protocols used in the protocol steganography are TCP, IPv4, NFS, CIFS etc.,

IV. STEGANOGRAPHY PHASES

In order to complete the hidden message exchanging process from sender to receiver, every Steganography algorithm must come through various stages.

Sender: The prime objective of the sender is to embed the hidden message in the stego-medium and transmit it through the channel of communication.

Communication channel: A physical or wireless medium that holds an encoded cover picture across the network or some other distribution medium with a hidden message. The embedding strategy in the middle attacks should be sufficiently advanced to secure the hidden message for all potential intrusion.

Receiver: In this steganography process, it is the last stage where the cover medium is retrieved and extracted to see if the hidden text that was sent over the communication channel.

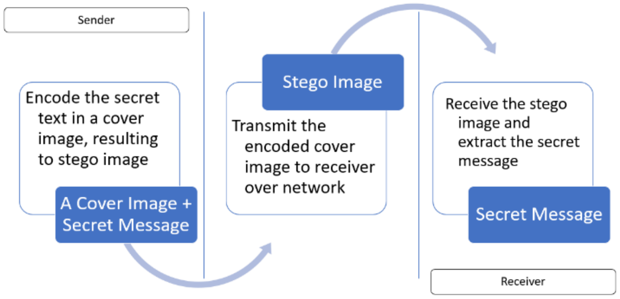
Sender
Encode the secret text in a cover image, resulting to stego image
A Cover Image + Secret Message
Stego Image
Transmit the encoded cover image to receiver over network
Receive the stego image and extract the secret message
Secret Message
Receiver

Fig 2. Phases of Steganography

Related Works

A succinct review based on the study of these papers related to our work is as follows. J. Homg et al. [4] and M. A. Hameed et al [5] described several image steganography techniques in spatial domain.

Along with existing techniques like LSB, layout management schemes and replacing only l's or only zero's, some more methods like replacing intermediate bit, raster scan principle, color-based data hiding and shape-based data hiding are also proposed.

M. C. Kasapbasi et al. [6] and S. D. R. I. Moses [7] developed an improved method for image-based steganography using LSB technique. All these techniques are primarily focused on the LSB steganography optimization and the steganography operation happens in one cover image. Though a high capacity focus is made on the research contributions in [4] to [7], these algorithms lack the ability to withstand the Steganalysis or man in the middle (MiM) attack on stego images during transmission with the intention of disrupting the transmission or acquiring the secret text that is in transit.

Existing Steganography Model – LSB

A typical Steganography system consists of following elements.

- Cover Object (C)
- Secret Message (M)
- Stego Object (S)

Cover Object

The cover objects in Steganography are those in which we are hiding secret messages. The cover object can be any digital files such as photos, audio, writing, images. The cover object that is most used is an image file to hide information. Most of the times the cover image stays as single file in a steganography cycle.

Secret Message

The actual hidden message in Steganography process, which has to be hidden in the cover object. It is important that the hidden message does not cause any visible quality degradation to the cover object.

Stego Object

After hiding the secret in a cover object, now the object is called as stego object. then, the stego object is transferred over public post or transferred over an email to the receiving end to complete the cycle.

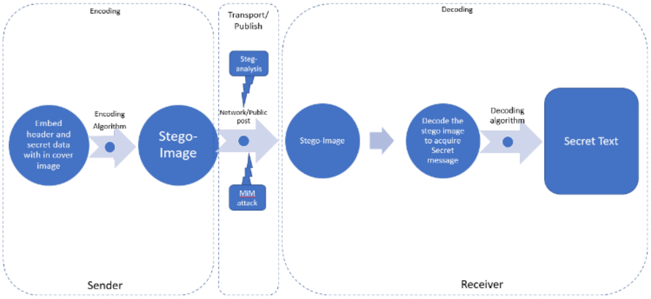

Fig 3. LSB Steganography model

Proposed multi-image cover object model

The proposed multi image cover model enables the receiver to send the secret text. This theoretical algorithm operates on the spatial domain of image steganography and places emphasis on maximizing the security of hidden message.

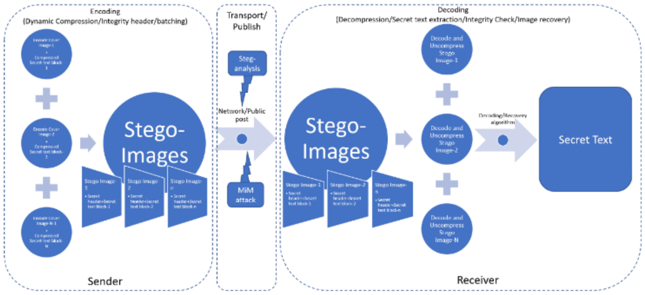
Encoding
(Dynamic Compression/Integrity header/batching)
Transport/ Publish
Decoding
(Decompression/Secret text extraction/Integrity Check/Image recovery)
Encode cover image-1 + Compressed Secret text block 1
Encode Cover image-2 + Compressed Secret text block-2
Encode cover image-N-1 + Compressed Secret text block N
Stego-Images
Stego Image 1 • Secret header+Secret text block-1
Stego Image 2 • Secret header+Secret text block-2
Stego Image n • Secret header+Secret text block-n
Steg-analysis
Network/Public post
MiM attack
Stego-Images
Stego Image-1 • Secret header+Secret text block-1
Stego Image-2 • Secret header+Secret text block-2
Stego Image-n • Secret header+Secret text block-n
Decode and Uncompress Stego Image-1
Decode and Uncompress Stego Image-2
Decode and Uncompress Stego Image-N
Decoding/Recovery algorithm
Secret Text
Sender
Receiver

Fig 4. Proposed Steganography model

Embedding Algorithm

As part of secret message embedding in given cover image first the algorithm reads the height (X) and width (Y) of the cover image and collects meta data of a secret message like number of words characters. The primary inputs to the algorithm are cover image list (C), secret message (M) and number of least significant bit (k) that must be used during the embedding process.

Algorithm-1: Multi cover image embedding model
Input: Cover images, Message and Random seed
Output: Stego image

```
procedure eLSB_embed(C, M, k)
Read cover images, C
Read secret message, M
$X \leftarrow Height\ of\ the\ cover\ image, C$
$Y \leftarrow Width\ of\ the\ cover\ image, C$
$W \leftarrow Number\ of\ words\ in\ the\ message, M$
$L \leftarrow Number\ of\ characters\ in\ the$
$message, M$
$E \leftarrow Equally\ divided\ secret\ message$
$based\ on\ n(c)$
$T \leftarrow String\ vector\ of\ words\ from\ the\ secret$
$message, M$
$S \leftarrow Secret\ message\ vector\ in\ binary\ form, T$
$Initialize, S \leftarrow [\ ]$
$for\ w \leftarrow 1\ to\ W\ in\ steps\ of\ 1\ do$
$S_{[w]} \leftarrow binary((T_{[w]}))$
end for
$H \leftarrow binary\ (StegoHeader(S))$
$S \leftarrow H + S$
$for\ e \leftarrow 1\ to\ E\ in\ steps\ of\ 1\ do$
$for\ i \leftarrow 1\ to\ Y\ in\ steps\ of\ 1\ do$
$for\ j \leftarrow 1\ to\ X\ in\ steps\ of\ 1\ do$
```

```
$for\ x \leftarrow 1\ to\ 8\ in\ steps\ of\ 1\ do$
$rb = resetFromNthBit(k)_b$
$C = \sum_{l=(8-k)}^{8} (c_{[i][j][x]} \& rb)$
$C = \sum_{l=(8-k)}^{8} (c_{[i][j][x]} \mid s_{[c++]})$
end for
end for
end for
end for
return list C, the secret text embedded stego images
end procedure
```

Extraction Algorithm

Extraction algorithm is similar to compression procedure, but the steps are orderly reversed to obtain original secret message from stego-image.

Algorithm-2: Multi cover image extraction algorithm
Input: Stego image
Output: Secret message

```
procedure_eLSB_extract(C)
Read Stego images, C
$H \leftarrow extracted\ header\ from\ stego\ images, C$
$X \leftarrow\ Height\ of\ the\ cover\ image\ from\ header, H$
$Y \leftarrow\ Width\ of\ the\ cover\ image\ from\ header, H$
$k \leftarrow number\ of\ LSB\ used\ from\ header\ data, H$
$l \leftarrow\ Length\ of\ secret\ message\ in\ bytes$
$from\ header\ data, H$
$E \leftarrow Equally\ divided\ secret\ message$
$based\ on\ n(c), from\ header\ data, H$
$Initialize, h\ \leftarrow 64$
$for\ i \leftarrow 1\ to\ Y\ in\ steps\ of\ 1\ do$
          $for\ e \leftarrow 1\ to\ E\ in\ steps\ of\ 1\ do$
$for\ j \leftarrow 1\ to\ X\ in\ steps\ of\ 1\ do$
$for\ x \leftarrow 1\ to\ 8\ in\ steps\ of\ 1\ do$
$if\ bytes \leq h$
$bytes\ \leftarrow bytes + 1$
$continue\ next;$
end if
$T = \sum_{l=(8-k)}^{8} (c_{[i][j][x]} \gg\ k)$
$S_{[i]} = S_{[i]} + T$
end for
end for
end for
end for
return T, the secret message
end procedure
```

Results:

Using the above proposed algorithm, a sample hello world can be encoded in five different cover images as below and the same can be sent over the network to the receiver to complete the communication cycle.

Table I. Sample text interpretation in the proposed algorithm using five cover images

Secret Text	ASCII	Binary Equivalent	Image-1	Image-2	Image-3	Image-4	Image-5(RS)
h	104	01101000	01	10	10	00	01
e	101	01100101	01	10	01	10	00
l	108	01101100	01	10	11	10	10
l	108	01101100	01	10	11	10	10
o	111	01101111	01	10	11	11	11
w	119	01110111	01	11	01	11	00
o	111	01101111	01	10	11	11	11
r	114	01110010	01	11	00	01	11
l	108	01101100	01	10	11	10	10
d	100	01100100	01	10	01	10	00

Let's assume that stego image-4 underwent an attack and the message digest mismatches with the content. On this contradiction, receiver will be able to identify that there was a corruption and recover the original text from other stego images as depicted below.

Table II. Secret text recovery on the loss of Image-3 due to steganalysis attack

Secret Text	ASCII	Binary Equivalent	Image-1 (I1)	Image-2 (I2)	Image-3 (I3)	Image-4 (I4)	Image-5 (RS)	Recovered Image-3 Ir = I1 ⊕ I2 ⊕ I3 ⊕ ...In ⊕)

h	104	01101000	01	10	10	00	01	10
e	101	01100101	01	10	01	10	00	01
l	108	01101100	01	10	11	10	10	11
l	108	01101100	01	10	11	10	10	11
o	111	01101111	01	10	11	11	11	11
w	119	01110111	01	11	01	11	00	01
o	111	01101111	01	10	11	11	11	11
r	114	01110010	01	11	00	01	11	00
l	108	01101100	01	10	11	10	10	11
d	100	01100100	01	10	01	10	00	01

The comparison of results with data transfer without corruption/intruder's intervention are shown in Table-4 with their corresponding checksums.

Table III. Comparison of checksum for the steganographed image, which transferred with corruption because of image color change

Image stage	**Checksum value(md5) (D)**	**Secret Text (ST)**
After Encoding	e64d69492b460cd25dbb42f970409f23	This is a secret text, which is hidden in an image file using steganography and having embedded checksum in it
After Decoding	e1c2e6f45c57978c86a78df764295972	Secret text got corrupted as the message digest are not identical

Table IV. Qualitative Comparison of proposed methodology

Parameters	**LSB methods (Existing)**	**Multi-image model Method (Proposed)**
Secret Text Recovery	No	Yes
Digest Inclusion on Stego files	No	Yes
Capability to identify MiM(Man In The Middle) attack	No	Yes
IPv4 Header Checksum check	Yes	Yes
Digest size used(md5)	0 bit	128 bits
Robustness	Less data loss	No Data loss
Integrity Check at receiving end	No	Yes

V. CONCLUSION

The proposed technique is effective on protecting secret message. Since the embedded checksum will validate for any unauthorized users or intruders corrupted the picture in any aspect. If any of the stego image underwent any steganalysis or MiM attack, then this proposed algorithm can regenerate the content of one stego image using other intact stego images received in the receiving end. Even if the attacker found the algorithm used for steganography in the stego picture by steganalysis and altered the quality of the hidden document, the tampering can be found by comparing the checksum if the same is obtained at the end of the recipient. Novelty of this approach, is the security of secret message is preserved and the model withstand a stego-image attack.

REFERENCES

[1] Petitcolas, FAP; Anderson RJ; Kuhn MG (1999). "Information Hiding: A survey" (PDF). Proceedings of the IEEE. 87 (7): 1062–78. CiteSeerX 10.1.1.333.9397. doi:10.1109/5.771065. Retrieved 2008-09-02.

[2] Dunbar, Bret. “A detailed look at Steganographic Techniques and their use in an Open-Systems Environment.” (2002).

[3] Y. Liu, S. Liu, Y. Wang, H. Zhao, S. Liu “Video steganography: a review” Neurocomputing, 335 (2019), pp. 238-250 (2019)

[4] J. Horng, C. Chang and G. Li, "Steganography Using Quotient Value Differencing and LSB Substitution for AMBTC Compressed Images," in IEEE Access, vol. 8, pp. 129347-129358, 2020, doi: 10.1109/ACCESS.2020.3009232.

[5] M. A. Hameed, M. Hassaballah, S. Aly and A. I. Awad, "An Adaptive Image Steganography Method Based on Histogram of Oriented Gradient and PVD-LSB Techniques," in IEEE Access, vol. 7, pp. 185189-185204, 2019, doi: 10.1109/ACCESS.2019.2960254.

[6] M. C. Kasapbaşi, "A New Chaotic Image Steganography Technique Based on Huffman Compression of Turkish Texts and Fractal Encryption with Post-Quantum Security," in IEEE Access, vol. 7, pp. 148495-148510, 2019, doi: 10.1109/ACCESS.2019.2946807.

[7] Setiadi De Rosal Ignatius Moses, “Payload Enhancement on Least Significant Bit Image Steganography Using Edge Area Dilation”, in Polish Academy of Sciences, Committee of Electronics and Telecommunication, Vol. 65, No. 2, pp. 287-292, 2019, DOI 10.24425/ijet.2019.126312

[8] Jagan Raj J and Prasath S. Article: Validating Data Integrity in Steganographed Images using Embedded Checksum Technique. IJCA Proceedings on National Conference on Research Issues in Image Analysis and Mining Intelligence NCRIIAMI 2015(1):5-8, June 2015

[9] J. R. Jayapandiyan, C. Kavitha and K. Sakthivel, "Enhanced Least Significant Bit Replacement Algorithm in Spatial Domain of Steganography Using Character Sequence Optimization," in IEEE Access, vol. 8, pp. 136537-136545, 2020, doi: 10.1109/ACCESS.2020.3009234.

[10] Jayapandiyan, Jagan Raj & C., Kavitha & Sakthivel, K.. (2020). Optimal Secret Text Compression Technique for Steganographic Encoding by Dynamic Ranking Algorithm. Journal of Physics: Conference Series. 1427. 012005. 10.1088/1742-6596/1427/1/012005.

[11] Jayapandiyan, Jagan Raj & C., Kavitha, “Coalesced Technique in Steganographic Images Using Encoded Conversion for Augmented Security” in International Journal of Innovative Technology and Creative Engineering, Vol.7, No.4, April 2017

[12] Jayapandiyan, Jagan Raj & C., Kavitha, “Enhancing the data security and data integrity in steganographed images by store bit randomization” in in International Journal of Innovative Technology and Creative Engineering, Vol.5, No.12, Dec 2015