TELKOMNIKA, Vol.17, No.3, June 2019, pp.1417~1424
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
DOI: 10.12928/TELKOMNIKA.v17i3.9513

Optimization of video steganography with additional compression and encryption

Dwi Arraziqi*[1], Endi Sailul Haq[2]
[1]STIKOM PGRI Banyuwangi/Department of Informatics Engineering, Banyuwangi, Indonesia
[2]State Politechnic of Banyuwangi/Department of Informatics Engineering, Banyuwangi Indonesia
*Corresponding author, e-mail: dwi.arraziqi@stikombanyuwangi.ac.id[1], endi@poliwangi.ac.id[2]

Abstract

Currently, data such as text, images and video are very important. Therefore, data must be secured from unauthorized parties. In this paper, we propose a number of security levels, first using compression techniques on the data that will be hidden to reduce the size of the data, second using encryption techniques on data that has been compressed so that data is more secure, third using video steganography techniques on compressed and encrypted data so that unauthorized parties are increasingly difficult to extract data. Measurement of differences in quality of cover-video and stego-video using MSE (Mean Square Error), PSNR (Peak Signal to Noise Ratio), evolution histogram of video tags, and video playback. The results show that the proposed algorithm gives better results than the previous algorithm which has a smaller MSE, larger PSNR, smaller histogram evolution of video tags, and play video without distortion.

Keywords: *compression, distribution, encryption, PSNR, steganography*

1. Introduction

Nowadays the development of communication network technology is very fast which makes information exchange very easy. Information is the wealth of an organization that must be sent safely through an unsecured communication network. Steganography serves to guard from extracting confidential information by unauthorized parties. Steganography is the art of hiding confidential information into cover media where the unauthorized party is very difficult to detect the existence of confidential information [1-28]. Recently, many data hiding algorithms have been developed on FLV [1, 2, 5, 6]. Data hiding is very important in identification, authentication, and copyright protection of digital media [1]. Sometimes confidential information has a size that larger than the cover media, cause distortion in the cover media. This can cause suspicion by unauthorized parties to the existence of confidential information. These problems can be minimized by compression.

Compression is the conversion of input data streams into other data streams that have a smaller size [29]. Stream can be a file, buffer in memory, or individual bits sent. Compression on confidential information can reduce file size [8, 12-14, 17, 19, 23, 24, 26, 27]. So that confidential information is much safer, encryption needs to be done [7-9, 11, 13-16,18, 20-22, 24, 25, 28]. Encryption is an algorithm that performs various substitutions and transforms the plaintext [30]. The original information is called plainteks while the encrypted form is called chipertext. Original information that has become a chiperteks has a high level of security [9]. Chipertext is divided and distributed at the end of each video tag evenly so that it has a layered security level.

The structure of FLV file consists of a short header, followed by a meta tag and then alternates between the audio tag and the video tag as shown in Figure 1 [1, 5, 6]. The header consists of 4 parts, the signature with the hex value "46 4c 56" which is translate to "FLV" in the hex string value; version with hex values "01"; flags with hex values "04" (audio), "01" (video), or "05" (audio + video); and offset the hex value "00 00 00 09". Meta tags, audio tags, and video tags have the same 6 parts namely type, body length, timestamp, timestampextended, streamed, and body (actual data). Type tags are 0x08 (audio), 0x09 (video), and 0x12 (meta).

Received April 6, 2018; Revised January 28, 2019; Accepted February 12, 2019

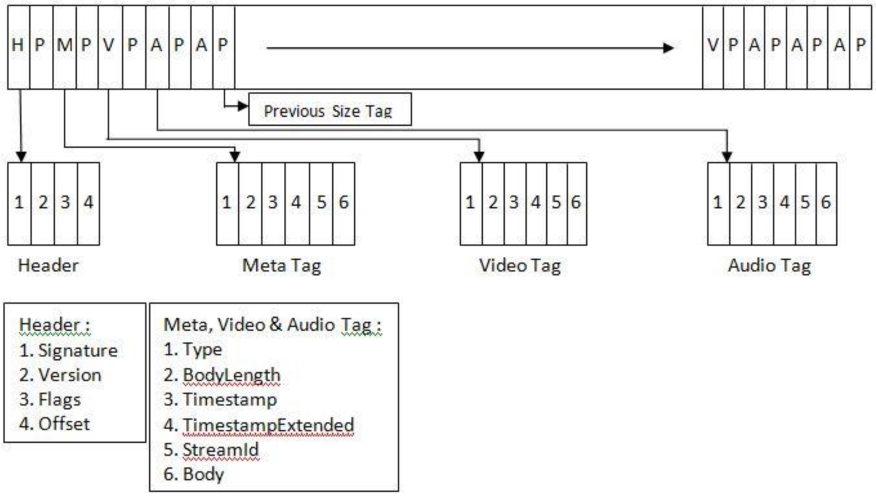


Figure 1. Structure of FLV file

Cruz et al. [1] discuss security for digital records through the application of steganography. The proposed effective steganographic methods are as follows: (1) injection at the EOF; (2) embedding at a video tag inside the FLV; (3) embedding at the metadata; and (4) distributing the stego file among all of the video tags. Each method has its corresponding strengths and weaknesses, and the stego FLVs created using the proposed techniques were analyzed qualitatively by using auditory-visual perception tests and quantitatively by using video tags evolution graphs and histograms and RGB averaging analysis.

The methods of hiding the stego file at the end of a video tag and distributing the stego file among all of the video tags of a carrier are recommended as the most effective steganographic methods for hiding information inside an FLV carrier for the following reasons: (1) no noticeable distortions can be observed in the video and audio quality of the stego FLV, making it looks like a typical FLV out in the open; (2) the stego file is hidden inside the FLV and not at the EOF or header, making it less prone to suspicion or attack; (3) the stego file cannot be altered even if the stego FLV is converted to a different file format and when its header or metadata is changed or removed; (4) checking for suspicious content (the video tags evolutions and histograms) via software, such as automated FLV analysis software, is difficult for a typical third party or observer who obtains the stego FLV.

Mozo et al. [2] presented his experiments on how FLV file structures can be manipulated to store confidential data. The FLV file characteristics can store all of data types in it and extract the same hidden information. Embedding data is safer at the end of the video tag. Very promising results include lossless compression, perfect original images and sound quality when modified FLV is transferred via the internet. The application has been applied to medical privacy records. Chang et al. [3] presented a data hiding algorithm using the (Discrete Sine Transform) DCT and (Discrete Cosine Transform) DST method in High-Efficiency Video Coding (HEVC). HEVC intra coded frame is used to hide data without spreading errors from neighboring blocks. Blocks from the quantized DCT (QDCT) and DST coefficients are selected to hide confidential data using specific intra prediction mode. The combination modes from neighboring blocks will produce three patterns of error distribution direction from data hiding consisting of vertical, horizontal, and diagonal. Each error distribution pattern has a range of intra prediction modes that protect a group of pixels in a particular direction. Range starts at 0 and ends 34.

Ma et al. [4] presented a data hiding algorithm using the QDCT (Quantized Discrete Cosine Transform) method in H.264/advanced video coding (AVC). To maintain the spread of distortion, the author has developed three conditions to determine the direction of intraframe prediction modes. Several paired coefficients from the 4x4 DCCT block to accumulate embedding that cause distortion. The proposed method can achieve low visual distortion. However, this method has the disadvantage of low insertion capacity because data hiding is only carried out on luminance from the block intraframe that meets the three selected conditions.

2. Research Method

2.1. Specification of Cover FLV

The specifications of the cover FLV include duration, width, height, frame rate, video codec, audio codec, size, number of video tag, number of audio tag, and average body length of video tag as shown in Table 1. Cover FLV are used 1.flv, 2.flv, and 3.flv. Cover FLV is obtained from the internet.

Table 1. Specification of Cover FLV

Cover FLV	1.flv	2.flv	3.flv
Duration	00:16	00:06	00:18
Width	360	480	300
Height	288	320	240
Frame Rate	25	29.916	25
Video Codec	On2 VP6	Sorenson's H.263	Sorenson's H.263
Audio Codec	MPEG layer 3	MPEG layer 3	MPEG layer 3
Size (byte)	669036	626325	773575
Number of audio tag	649	234	693
Number of video tag	424	185	451
Average body length of video tag (byte)	1337	3084	1352

2.2. Proposed Algorithm

The proposed algorithm is the development of the previous algorithm [1]. The algorithm previously [1] said that the distribution of secret data at the end of the video tags is the best steganography algorithm in FLV. The proposed algorithm adds a compression and encryption algorithm to secret data before being distributed. There are 2 proposed algorithms, embedding and extracting algorithms. Embedding algorithm is an algorithm that embeds secret data into the cover video so that it produces stego video. Extracting algorithm is an algorithm that extracts secret data from stego video. Design of embedding algorithm as shown in Figure 2. The algorithm process is explained as follows:

- Determine secret data and cover FLV.
- Split the cover FLV so that it produces frames (video tags) and audio tags.
- Compress secret data using ZIP algorithm.
- Encrypt secret data that has been compressed using AES algorithm.
- Embed secret data that has been compressed and encrypted at the end of the frame evenly
- Merge between non stego frames, stego frames, and audio tags so as to produce stego FLV.

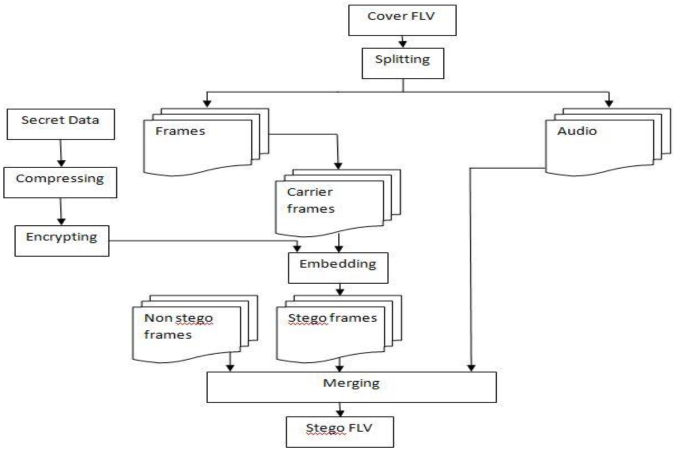


Figure 2. Design of embedding algorithm

Design of extracting algorithm as shown in Figure 3. The algorithm process is explained as follows:

- Determine cover FLV and stego FLV.
- Split cover FLV and stego FLV to produce frames (video tags) and audio tags.
- Subtract frames between stego FLV with cover FLV generating secret data that has been compressed and encrypted.
- Decrypt secret data that has been compressed and encrypted using the AES algorithm to produce secret data that has been compressed.
- Decompress secret data that has been compressed using the ZIP algorithm to generate secret data.

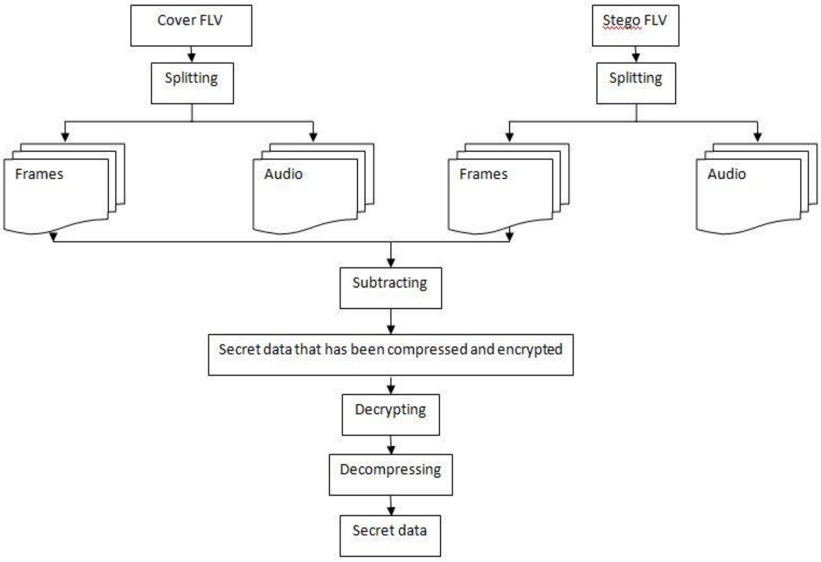


Figure 3. Design of extracting algorithm

3. Results and Analysis

Several trials were conducted to prove the video quality of the proposed algorithm compared to the previous algorithm [1]. Quality measurement is very difficult to use the naked eye. Two commonly used quality measurements are MSE and PSNR [7-10, 12, 14, 16, 18, 20, 22-25, 27]. Other quality measurements are playback and histogram of the video tag evolution. PSNR is a comparison between the maximum value of a signal measured by the amount of noise that affects the signal. PSNR is usually measured in decibels (dB). PSNR is used to find out the FLV cover quality comparison with FLV stego. To determine the PSNR, the MSE value (Mean Square Error) must first be determined as given in (2). MSE is the error value of the average square between FLV cover and FLV stego as given in (1).

$$MSE = \frac{1}{M*N}\sum_{i=1}^{M}\sum_{j=1}^{N}[C(i,j) - S(i,j)]^2 \quad (1)$$

where, M is the row while N is the column. C (i, j) is the pixel value of FLV cover in row i and column j. S (i, j) is the pixel value of FLV stego in row i and column j.

$$PSNR = 10 * \log\left(\frac{P^2}{MSE}\right) \quad (2)$$

where, P = max(C(i,j), S(i,j)).

3.1. Based on MSE and PSNR

The proposed algorithm has better quality than the previous algorithm [1] as shown in Table 2. The proposed algorithm has a smaller MSE value. The proposed algorithm also has a greater PSNR value. The smaller the MSE value means the better the quality. The greater the value of the PSNR means the better the quality.

Table 2. Comparison of Proposed Algorithm with Previous Algorithm

Cover FLV	Screet data and size	Method	ZIP compression (byte)	AES encryption (byte)	The average that was embedded to the video tag (byte)	MSE	PSNR (dB)
1.flv	Test.accdb 839680 byte	The proposed algorithm	21538	21552	50	0	∞
		The previous algorithm			1980	0	∞
2.flv		The proposed algorithm	21538	21552	116	0	∞
		The previous algorithm			4538	309	23
3.flv		The proposed algorithm	21538	21552	47	0	∞
		The previous algorithm			1861	880	18

3.2. Based on Histogram of The Video Tag Evolution

The proposed algorithm has better quality with the previous algorithm [1] as shown in Figure 4. Figure 4 (a) is a comparison of the histogram evolution of video tags from cover FLV (1.flv), stego FLV from proposed algorithm, and stego FLV from previous algorithm [1]. Figure 4 (b) is a comparison of histogram evolution of video tags from cover FLV (2.flv), stego FLV from proposed algorithm, and stego FLV from previous algorithm [1]. Figure 4 (c) is a comparison of histogram evolution tag video from cover FLV (3.flv), stego FLV from proposed algorithm, and stego FLV from previous algorithm [1]. The x axis is the order of the video tag while the y axis is the number of data (bytes) of the video tag. The proposed algorithm has a smaller evolution of video tags means the better the quality.

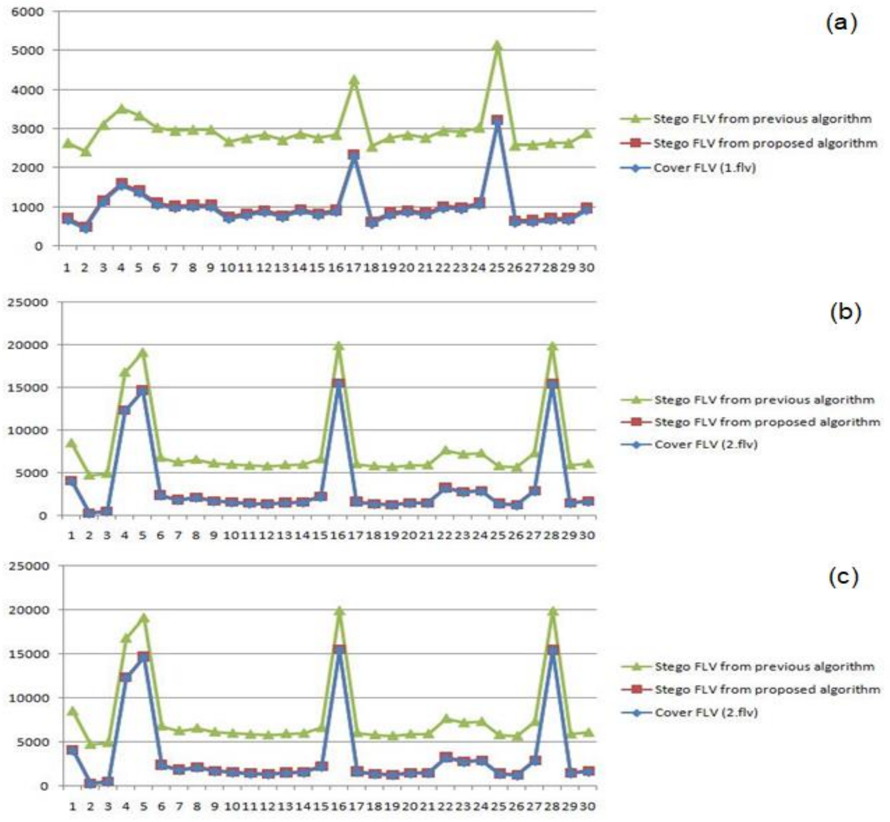
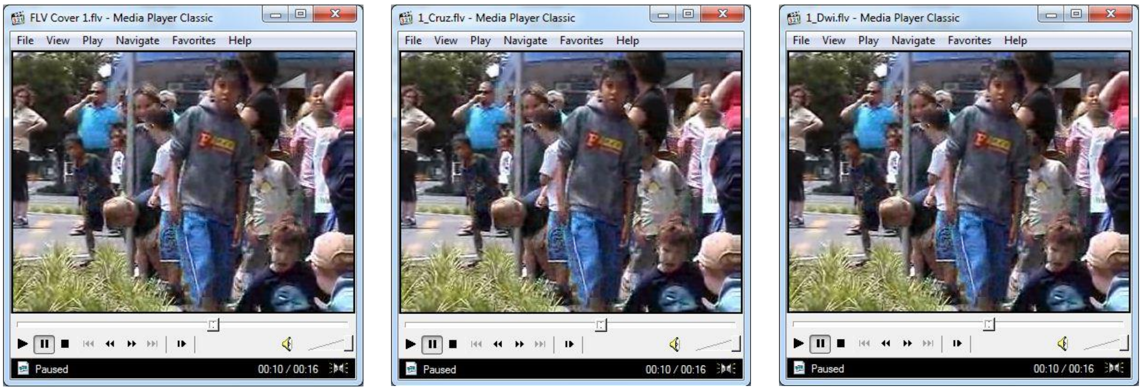


Figure 4. Histogram evolution of video tags from the proposed algorithm; previous algorithm; and cover FLV (a) 1.flv, (b) 2.flv, (c) 3.flv

3.3.Based on Playback

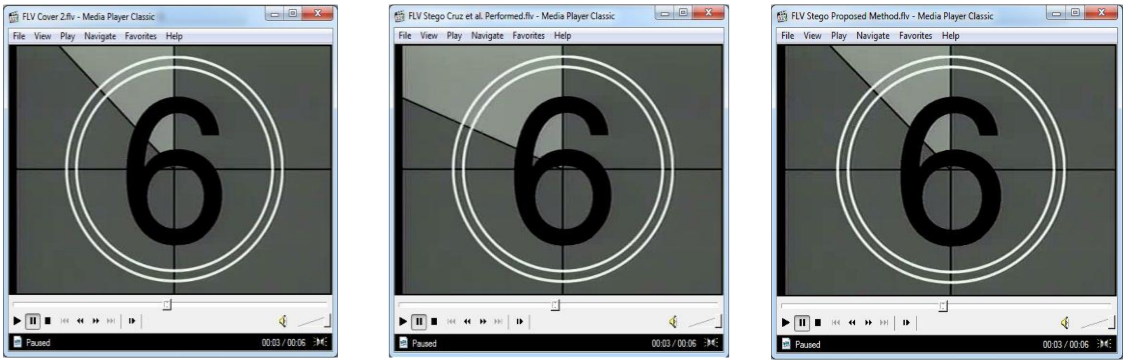
The proposed algorithm has the same good quality as the previous algorithm [1] as shown in Figure 5. Figure 5 (a) is a screenshot of cover FLV (1.flv). Figure 5 (b) is a FLV stego screenshot from the previous algorithm [1]. Figure 5 (c) is a FLV stego screenshot of the proposed algorithm. If we look at the naked eye at 10th second, there is no distortion seen from either the previous algorithm [1] or the proposed algorithm.


FLV Cover 1.flv - Media Player Classic
File View Play Navigate Favorites Help
Paused
00:10 / 00:16
1_Cruz.flv - Media Player Classic
File View Play Navigate Favorites Help
Paused
00:10 / 00:16
1_Dwi.flv - Media Player Classic
File View Play Navigate Favorites Help
Paused
00:10 / 00:16

(a) (b) (c)

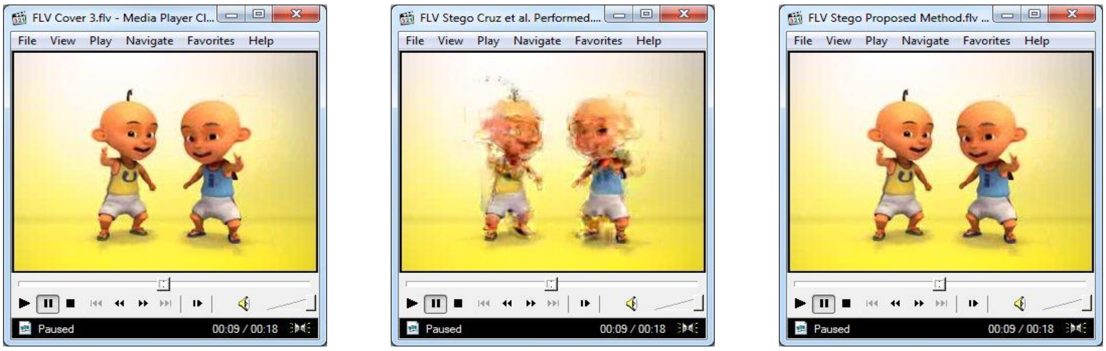
Figure 5. (a) Screenshots from the Cover FLV 1.flv;
(b) Stego FLV from previous algorithm and; (c) Stego FLV from proposed algorithm

The proposed algorithm has has better quality than the previous algorithm [1] as shown in Figure 6. Figure 6 (a) is a screenshot of cover FLV (2.flv). Figure 6 (b) is a FLV stego screenshot from the previous algorithm [1]. Figure 6 (c) is a FLV stego screenshot of the proposed algorithm. If we look at the naked eye at 3th second, there is no distortion seen from the proposed algorithm while the previous algorithm [1] looks distortion. The proposed algorithm has has better quality than the previous algorithm [1] as shown in Figure 7. Figure 7 (a) is a screenshot of cover FLV (3.flv). Figure 7 (b) is a FLV stego screenshot from the previous algorithm [1]. Figure 7 (c) is a FLV stego screenshot of the proposed algorithm. If we look at the naked eye at 9th second, there is no distortion seen from the proposed algorithm while the previous algorithm [1] looks distortion.


FLV Cover 2.flv - Media Player Classic
File View Play Navigate Favorites Help
6
Paused
00:03 / 00:06
FLV Stego Cruz et al. Performed.flv - Media Player Classic
File View Play Navigate Favorites Help
6
Paused
00:03 / 00:06
FLV Stego Proposed Method.flv - Media Player Classic
File View Play Navigate Favorites Help
6
Paused
00:03 / 00:06

(a) (b) (c)

Figure 6. (a) Screenshots from the cover FLV 2.flv; (b) Stego FLV from previous algorithm, and
(c) Stego FLV from proposed algorithm


FLV Cover 3.flv - Media Player Cl...
File View Play Navigate Favorites Help
Paused
00:09 / 00:18
FLV Stego Cruz et al. Performed....
File View Play Navigate Favorites Help
Paused
00:09 / 00:18
FLV Stego Proposed Method.flv ...
File View Play Navigate Favorites Help
Paused
00:09 / 00:18

(a) (b) (c)

Figure 7. (a) Screenshots from the Cover FLV 3.flv, (b) Stego FLV from previous algorithm, and (c) Stego FLV from proposed algorithm

4. Conclusion

The proposed algorithm has better quality both in terms of security and video quality. In terms of security, the proposed algorithm encrypts confidential data using the AES algorithm. In terms of video quality, FLV stego from the proposed algorithm has a smaller MSE, a larger PSNR, smaller histogram evolution of video tags, and plays videos without distortion. For the next research, the process of embedding secret data is done randomly so that secret data is more secure.

References

[1] Cruz JP, Libatique NJ, Tangonan G. *Steganography and data hiding in flash video (FLV)*. TENCON 2012 IEEE Region 10 Conference. Cebu. 2012: 1-6.

[2] Mozo AJ, Obien ME, Rigor CJ, Rayel DF, Chua K,Tangonan G. *Video steganography using Flash Video (FLV)*. 2009 IEEE Instrumentation and Measurement Technology Conference. Singapore. 2009: 822-827.

[3] Chang PC, Chung KL, Chen JJ, Lin CH, Lin TJ. A Dct/Dst-Based Error Propagation-Free Data Hiding Algorithm for Hevc Intra-Coded Frames. *Journal of Visual Communication and Image Representation*. 2014; 25(2): 239-253.

[4] Ma X, Li Z, Tu H, Zhang B. A Data Hiding Algorithm for H.264/Avc Video Streams without Intra-Frame Distortion Drift. *IEEE Transactions on Circuits and Systems for Video Technology*. 2010; 20(10): 1320-1330.

[5] Bawaneh MJ, Obeidat AA, M.Al-kofahi M. An Adaptive FLV Steganography Approach Using Simulated Annealing. *International Journal of Communication Networks and Information Security*. 2018; 10(1): 56-66.

[6] Atiea MA, Mahdy YB, Hedar AR. Hiding data in FLV video file. *Advances in Intelligent and Soft Computing*. 2012; 167: 919-925.

[7] Jothimani S, Shaheen H. Securing Video files using Steganography Method in Android Mobile. *International Journal of Scientific and Engineering Research*. 2016; 7(4): 46-49.

[8] Sarmah DK, Kulkarni AJ. Improved Cohort Intelligence-A high capacity, swift and secure approach on JPEG image steganography. *Journal of Information Security and Applications*. 2019; 45: 90-106.

[9] Pandey A, Singh B, Saini BS, Sood N. A novel fused coupled chaotic map based confidential data embedding-then-encryption of electrocardiogram signal. *Biocybernetics and Biomedical Engineering*. 2018: 1-19.

[10] Liu Y, Liu S, Wang Y, Zhao H, Liu S. Video steganography: A review. *Neurocomputing*. 2018.

[11] Chikouche, SL, Chikouche N. *An improved approach for lsb-based image steganography using AES algorithm*. 2017 5th International Conference on Electrical Engineering-Boumerdes (ICEE-B). Boumerdes. 2017: 1-6.

[12] Khan S, Irfan MA, Ismail M, Khan T, Ahmad N. *Dual lossless compression based image steganography for low data rate channels*. 2017 International Conference on Communication Technologies (ComTech). Rawalpindi. 2017: 60-64.

[13] Malathi M, Rahul M, Kumar NS., Thamaraiselvan R. *Enhanced image steganography using AES &amp;amp; SPIHT compression*. 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS). Coimbatore. 2017: 1-5.

[14] Mumthas S, Lijiya A. Transform Domain Video Steganography Using RSA, Random DNA Encryption and Huffman Encoding. *Procedia Computer Science*. 2017; 115(C): 660-666.

[15] Prasetyadi GC, Mutiara AB, Refianti R. *File encryption and hiding application based on advanced encryption standard (AES) and append insertion steganography method*. 2017 Second International Conference on Informatics and Computing (ICIC). Jayapura. 2017: 1-5.

[16] Houssein EH, Ali MAS, Hassanien AE. *An image steganography algorithm using Haar Discrete Wavelet Transform with Advanced Encryption System*. 2016 Federated Conference on Computer Science and Information Systems (FedCSIS). Gdansk. 2016: 641-644.

[17] Dhanawe SA., Doshi SV. *Hiding file on Android mobile and sending APK file through whatsapp using steganography and compression techniques*. 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES). Paralakhemundi. 2016: 106-110.

[18] Jain M, Lenka SK, Vasistha SK. Adaptive circular queue image steganography with RSA cryptosystem. *Perspectives in Science*. 2016; 8: 417-420.

[19] Malik A, Sikka G, Verma HK. A high capacity text steganography scheme based on LZW compression and color coding. *Engineering Science and Technology, an International Journal*. 2017; 20(1): 72-79.

[20] Muhammad K, Sajjad M, Mehmood I, Rho S, Baik SW. Image steganography using uncorrelated color space and its application for security of visual contents in online social networks. *Future Generation Computer Systems*. 2018; 86: 951-960.

[21] Reddy MIS, Kumar APS. Secured Data Transmission Using Wavelet Based Steganography and Cryptography by Using AES Algorithm. *Procedia Computer Science*. 2016; 85: 62-69.

[22] Sabnis SK, Awale RN. Statistical Steganalysis of High Capacity Image Steganography with Cryptography. *Procedia Computer Science*. 2016; 79: 321-327.

[23] Tang M, Zeng S, Chen X, Hu J, Du Y. An adaptive image steganography using AMBTC compression and interpolation technique. *Optik*. 2016; 127(1): 471-477.

[24] Mishra R, Mishra A, Bhanodiya P. *An edge based image steganography with compression and encryption*. 2015 International Conference on Computer, Communication and Control (IC4). Indore. 2015: 1-4.

[25] Xiang T, Hu J, Sun J. Outsourcing chaotic selective image encryption to the cloud with steganography. *Digital Signal Processing*. 2015; 43(C): 28-37.

[26] Zhang Y, Luo X, Yang C, Ye D, Liu F. *A JPEG-Compression Resistant Adaptive Steganography Based on Relative Relationship between DCT Coefficients*. 2015 10th International Conference on Availability, Reliability and Security. Toulouse. 2015: 461-466.

[27] Lin CY, Wu SC, Wang JJ. *VQ Image Compression Steganography Based on Section-Based Informed Embedding*. 2014 International Symposium on Computer, Consumer and Control. Taichung. 2014: 111-114.

[28] Ren-Er Y, Zhiwei Z, Shun T, Shilei D. *Image Steganography Combined with DES Encryption Pre-processing*. 2014 Sixth International Conference on Measuring Technology and Mechatronics Automation. Zhangjiajie. 2014: 323-326.

[29] Salomon D, Motta G. Data Compression the Complete Reference. Fifth Edition. New York: Springer. 2010.

[30] Stallings W. Cryptography and Network Security Principles and Practice. Sixth Edition. New York: Prentice Hall. 2014.