Multidim Syst Sign Process (2019) 30:529–542
https://doi.org/10.1007/s11045-018-0568-2

ORIGINAL PAPER

A two-level secure data hiding algorithm for video steganography

S. Manisha[1] · **T. Sree Sharmila**[2]

Received: 7 November 2016 / Revised: 9 March 2018 / Accepted: 16 March 2018 /
Published online: 20 March 2018

Abstract Sensitive data is exchanged frequently through wired or wireless communication that are vulnerable to unauthorized interception. Cryptography is a solution to overcome this issue, but once decrypted the information secrecy does not exist. Apart from hiding data in an image, it can be extended for digital media. In this work, data hiding and extraction is proposed for Audio Video Interleave videos, that embeds the image in Bitmap Image File, that has the secret information in a frame of the video by segmenting the bytes of the secret image and placing them in the video frame providing a higher level of encryption. This novel method provides a two level encryption, thus to decipher the data, the way in which the secret image is originally decomposed and the frame in which it is embedded should be known. The quality of the secret image embedded and the size of the video is not altered before and after encryption of the secret data. The secret image may contain any multimedia data that can be further extracted and recognized.

Keywords Data hiding · Video steganography · Adaptive LSB · Randomized encoding

1 Introduction

The last decade has seen various advances in the area of data transmission that arises the concern regarding Steganography. Steganography is a method of information hiding techniques. It embeds secret messages into a host medium so as not to arouse suspicion by an eavesdropper (Mazurczyk and Caviglione 2015a; Saikia and Thakur 2013). A steganographic application

✉ S. Manisha
manishas@ssn.edu.in

T. Sree Sharmila
sreesharmilat@ssn.edu.in

[1] Department of Computer Science and Engineering, SSN College of Engineering, Chennai, Tamil Nadu, India

[2] Department of Information Technology, SSN College of Engineering, Chennai, Tamil Nadu, India

coverts the communication between the sender and the receiver whose existence is unknown to a possible attacker. The success depends on detecting the existence of this communication (Mazurczyk and Caviglione 2015a). In general, the host medium used in steganography includes meaningful digital media such as digital image, text, audio, video, 3D model etc (Mazurczyk and Caviglione 2015b). In Image Steganography the information is hidden exclusively in images. It encodes or embeds any secret information such that the existence of the information is invisible. The original medium is referred to as cover text, cover image, or cover audio. After inserting the secret message it is referred to as stego-medium. A stego-key is used for hiding/encoding process to restrict detection or extraction of the embedded data.

There are several similarities and differences between Cryptography and Steganography (Saikia and Thakur 2013). Both transforms information into a form that is incomprehensible to a third party, uses key to encrypt or decrypt. Cryptography does not hide the information or hide the fact of communication. It does not ensure anonymity of the communicating parties thus not needing any additional carrier. The amount of information transmitted in the communication process depends on the amount of the encrypted information. Steganography on the other hand hides the information and the fact of communication. It ensure anonymity of the communication parties and the amount of information transmitted is greater than the encrypted information. It needs an additional carrier.

Several steganographic systems uses different multimedia objects such as image, audio, video etc as a cover medium as people often transmit digital pictures over email or other communication. Modern steganography hides information into digital multimedia files at the network packet level (Saikia and Thakur 2013). Elements such as cover medium (C) to hold the secret message, stego-key (K) to encrypt or decrypt the message, the secret message (M) which can be plain text or a digital image and steganographic techniques are required to hide an information into any medium. Depending on the type of the cover medium, steganography can be classified as Text, Image, Audio, Video and Protocol Steganography. Text Steganography uses a text file as a cover to hide a secret message. When the secret message is embedded into an image and transmitted, an intruder can only notice an image but can't see the existence of the hidden message. Audio Steganography uses the technique, that makes an audible sound inaudible in the presence of another audible sound. This property can be used to select the channel to hide the information. Protocol Steganography allows to embed information within a network protocol in the TCP/IP header and other optional fields that are rarely used. Steganalysis is the process of breaking the steganography and detecting the stego message (Mazurczyk and Caviglione 2015b). Secret communications, feature tagging and copyright protection are several applications of steganography.

The rest of the paper is organized as follows. In Sect. 2, the existing approaches on video steganography are discussed. In Sect. 3, the state of the art steganographic methods are discussed to clearly differentiate the different steganographic methods. In Sect. 4, the proposed Adaptive LSB Encryption with Randomized Encoding Algorithm is discussed. Section 5, contains the experimental results and the comparison of the proposed method with the state-of-the art methods. The comparison in terms of MSE and PSNR is also done between the encoded secret image and the decoded secret image for the proposed method. In Sect. 6, the conclusion and future research directions are described (Fig. 1).

2 Related work

Monjul Saikia and Vandana Thakur in their paper Saikia and Thakur 2013, "Hiding Secret Image in Video", describe a data hiding technique to embed the secret message bits in Discrete

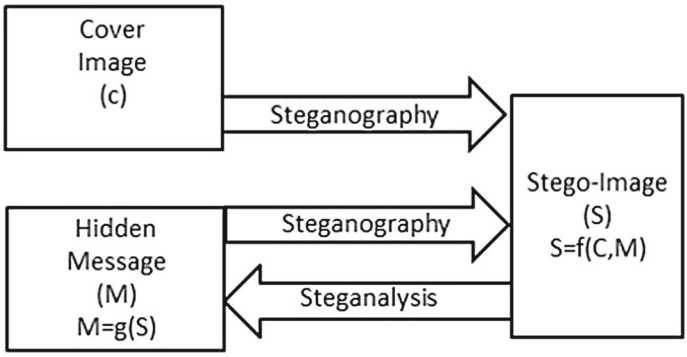

Fig. 1 Steganalysis

Cosine Transform (DCT) higher order coefficients. Here they propose a data hiding and extraction procedure for Audio Video Interleave (AVI) videos embedding the secret message bits in DCT higher order coefficients. The secret information taken here is a grayscale image pixel values. The grayscale pixel values are converted to binary values and are embedded in higher order coefficient value of DCT of AVI video frames. A method to hide data in more than one channel can be developed thereby increasing the full data holding capacity of the carrier video.

Saeed Sarreshtedari and Mohammad Ali Akhaee proposed the "One-third probability embedding: a new + 1 histogram compensating image least significant bit steganography scheme". In this, compared to the recently proposed image steganography techniques, the new method called one-third LSB embedding reduces the probability of change per pixel to one-third without sacrificing the embedding capacity (Sarreshtedari and Akhaee 2013). This improvement results in a better imperceptibility and also higher robustness against well-known Least Significant Bit (LSB) detectors. Bits of the message are carried using a function of three adjacent cover pixels. It is shown that no significant improvement is achieved by increasing the length of the pixel sequence employed. A closed-form expression for the probability of change per pixel in terms of the number of pixels used in the pixel groups has been derived. Another advantage of the proposed algorithm is to compensate, as much as possible, for any changes in the image histogram. A level k expression to predict the probability of change per pixel has to be developed thereby increasing the cost efficiency. Jiangqun Ni, Linjie Guo and Yun QingShi devised the "Uniform embedding for efficient JPEG steganography" in which a class of new distortion functions known as Uniform Embedding Distortion function (UED) is presented for both side-informed and non side-informed secure Joint Photographic Experts Group (JPEG) steganography (Ni et al. 2014). By incorporating the syndrome trellis coding, the best codeword with minimal distortion for a given message is determined with UED, which instead of random modification, tries to spread the embedding modification uniformly to quantized Discrete Cosine Transform (DCT) coefficients of all possible magnitudes. In this way, less statistical detectability is achieved, owing to the reduction of the average changes of the first-order and second-order statistics for DCT coefficients as a whole performance. While the UED proposed in their paper could by no means tackle all issues, it raises a quite challenging open question, that is how to evaluate the embedding costs of all possible DCT coefficients (including DCs, zero and non-zero ACs) based solely on the coefficients in the DCT domain for JPEG steganography, which remains as the topic of their future research effort.

Bin Li, Jiwu Huang, Ming Wang and Shunquan Tan, in their research "Investigation of cost assignment in spatial image steganography" devised a distribution. They analytically show that the cost-value distribution determines the change rate of cover elements. Furthermore, when the cost-values are specified to follow a uniform distribution, the change rate has a linear relation with the payload, which is a rare property for content-adaptive steganography (Li et al. 2014). In addition, they proposed some rules for ranking the priority profile for spatial images. Following such rules, they proposed a five-step cost assignment scheme. A method to calculate cost value distribution is to be found out. Wojciech Mazurczyk and Luca Caviglione proposed the "Steganography in modern smartphones and mitigation techniques". It surveys the state of the art of steganographic techniques for smartphones, with emphasis on methods developed over the period 2005 to the second quarter of 2014. The different approaches are grouped according to the portion of the device used to hide information, leading to three different covert channels, i.e., local, object and network (Mazurczyk and Caviglione 2015b). Also, it reviews the relevant approaches used to detect and mitigate steganographic attacks or threats. Lastly, it showcases the most popular software applications to embed secret data into carriers, this enables the creation of local covert channels for only android based devices. Wojciech Mazurczyk and Luca Caviglione proposed the "Information hiding as a challenge to malware detection". In this they exploit the uncertainity of motion estimation. They use motion vector based steganography that out performs the rest. Uses the uncertainty information for motion estimation. This has tiny degradation in coding efficiency, Peak Signal-to-Noise ratio (PSNR) and bit rate (Mazurczyk and Caviglione 2015a).

Shahrokh Ghaemmaghami, Soodeh Ahani and Z. Jane Wang, proposed the "Sparse Representation Based Wavelet Domain Steganography Method". In this, speech steganography method exploits the sparse representation to embed secret messages into higher semantic levels of the cover signal, resulting in increased undetectability Ghaemmaghami et al. 2015). The proposed method also yields improvements on both stego signal quality and embedding capacity, which are the two major requirements of a steganography algorithm. Experimental results illustrate that the stego signals generated by the proposed method are perceptually indistinguishable from the original cover signals, quantified by both Signal-to-Noise Ratio (SNR) and Perceptual Evaluation of Speech Quality (PESQ) quality measures. Chung-Ming Wang and Kuo-Chen Wu devised the "Steganography using reversible texture synthesis" in which they weave the texture synthesis process into steganography to conceal secret messages. In contrast to using an existing cover image to hide messages, their algorithm conceals the source texture image and embeds secret messages through the process of texture synthesis (Wang and Wu 2015). This allows to extract the secret messages and source texture from a stego synthetic texture. Only in texture images full capacity of this system be exploited. Dajun He, Qibin Sun and Qi Tian proposed "Remote authentication via biometrics: Robust Video Object Steganographic mechanism over wireless networks". This paper proposes a robust authentication mechanism based on semantic segmentation, chaotic encryption and data hiding. Assuming that user X wants to be remotely authenticated, initially X video object (VO) is automatically segmented, using a head-and-body detector. Next, one of X's biometric signals is encrypted by a chaotic cipher. Afterwards, the encrypted signal is inserted to the most significant wavelet coefficients of the VO, using its Qualified Significant Wavelet Trees (QSWTs). QSWTs provide both invisibility and significant resistance against lossy transmission and compression, conditions that are typical of wireless networks (Ntalianis and Tsapatsoulis 2016).

Finally, the inverse discrete wavelet transform is applied to provide the stego-object. Experimental results regarding:

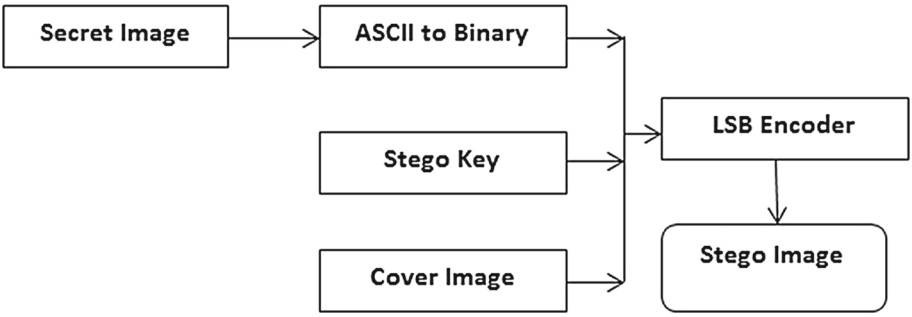

Fig. 2 LSB insertion

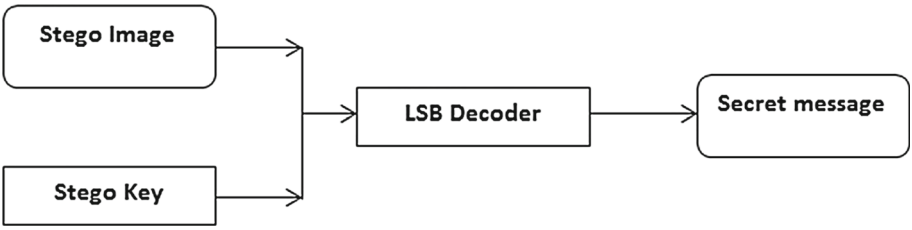

Fig. 3 LSB extraction

1. Security merits of the proposed encryption scheme.
2. Robustness to steganalytic attacks, to various transmission losses and JPEG compression ratios.
3. Bandwidth efficiency measures indicate the promising performance of the proposed biometrics-based authentication scheme. The encryption capacity is limited. It is designed for authentication purposes only and does not hold optimal capacity of data.

3 Steganographic methods

The Least Significant Bit Algorithm (LSB) is a major method used for Steganography. This method uses a simple strategy to embed the data into a cover medium that cannot be detected by a casual observer. The technique replaces some existing information in a given pixels with the information to be hidden. Although it is possible to embed data in any bit-plane, LSB embeds on the least significant bit of the pixel. This will reduce the variation in color without altering the data. For example, embedding into second bit-plane can change the color value by 2. When embedding is performed on the two least significant pixels, the color of the color medium varies by four. This technique avoids much variation as possible to minimize the likelihood of detection of any additional data. It is possible to use some information of the cover medium while using LSB technique. This is due to embedding directly into a pixel of the cover medium by discarding some information from them and replacing with the secret data. LSB insertion and extraction mechanism are shown below (Figs. 2, 3).

3.1 One bit LSB

This is the simplest way to hide information in an image. It blindly hides because it just starts at the top left corner of the image and works its way across the image (then down—in

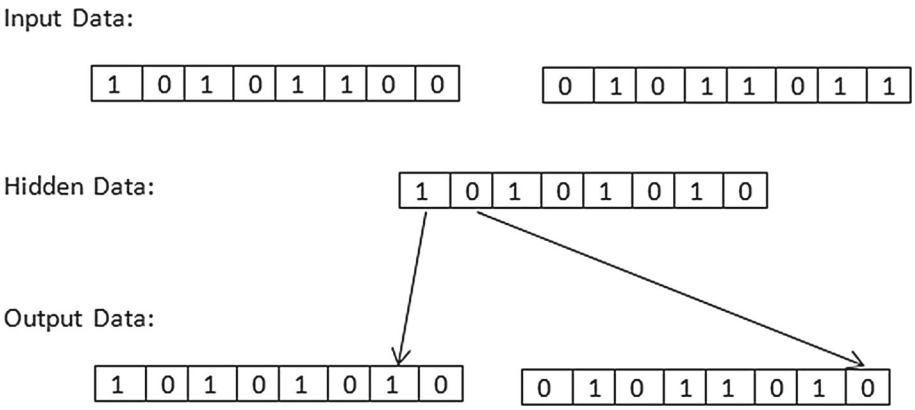

Fig. 4 Applying LSB using one least significant bit

scan lines) pixel by pixel. It changes the least significant bits of the pixel colors to match the message. To decode the process the least significant bits starting at the top left are read off. This is not very secure as it is really easy to read off the least significant bits. Also if the message doesn't completely fill up the possible space then just the top part of the image is degraded but the bottom is left unchanged making it easy to tell what is been changed. Using images in particular as the cover file, the choice of image and the format of the image are important not drawing attention to the fact that they could be concealing information. The format of the image is important. The use of a lossy file format such as JPEG on reconstruction of the embedded file, bits could be lost due to the compression. On the other hand, the use of lossless images is allowed for the hidden file to be reconstructed completely. However, the use of such an algorithm was limited as it allowed for up to only 12.5% (1/8) of the size of the image to be hidden (Fig. 4).

3.2 Two bit LSB

The use of the least significant bit algorithm was extended to increase hiding capacity by using not only the first least significant bit of each byte, but the second in order to hide information. Eg: Given a byte 1 0 1 1 0 1 1 0 the two least significant bits are the right most bits of the byte. The result of this was that twice the capacity of the least significant bit algorithm could be obtained (a maximum of 25% of the cover image). The possible problem with the two least significant bits algorithm is that since the second least significant bit was also changed to match bits of the secret file, there was a chance that the result would be noticeable to the human eye (Fig. 5).

3.3 Four bit LSB

This algorithm was an extension of the least significant bit algorithm similar to the two least significant bits algorithm. It uses the four least significant bits of each byte to hide information, thus it had a maximum capacity of 50%. The purpose of implementing this algorithm was to demonstrate the results when too much information is hidden. The resulting images provides a benchmark against those in which information had been successfully hidden (Fig. 6).

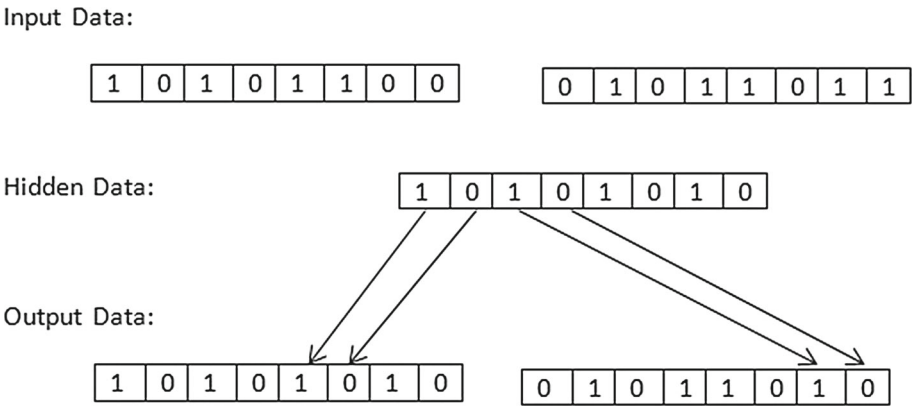

Fig. 5 Applying LSB using two least significant bit

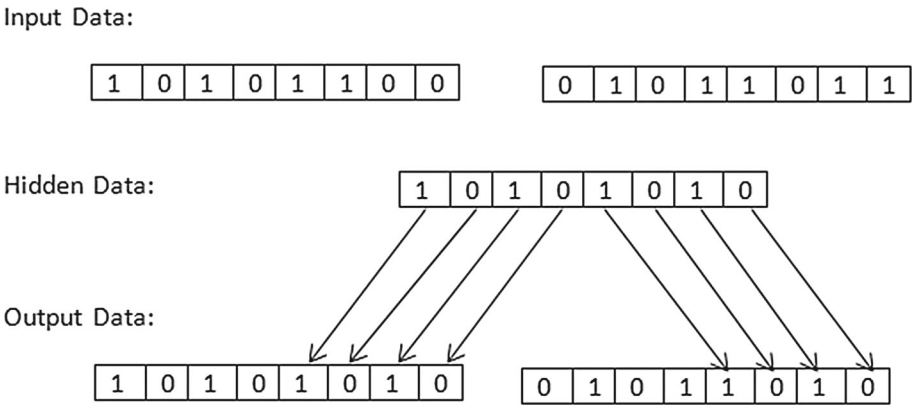

Fig. 6 Applying LSB using four least significant bit

3.4 Pixel swap algorithm

This algorithm randomly select 2 pixels from the cover image using a pseudo-random sequence. If the two pixels lie within a specified distance a ($a = 2$ or 3 generally), they are suitable for embedding, otherwise generate another set of pixels. If the message bit is zero or one, check if x1 is greater than x2 otherwise swap x1 and x2. Do the reverse operation for the message bit one (zero). For decoding, select the pixels using the same pseudo-random sequence. Check if the 2 pixels are within the pre-specified range a. If $x1 < x2$, the message bit is zero (one) otherwise the message bit is one (zero). This scheme preserves the first order statistic (histogram) inherently without applying separate restoration process. This scheme also does not add any visual distortion to the image since the threshold used for swapping of pixels is kept considerably small which only affects the least significant bit planes of an image (Fig. 7).

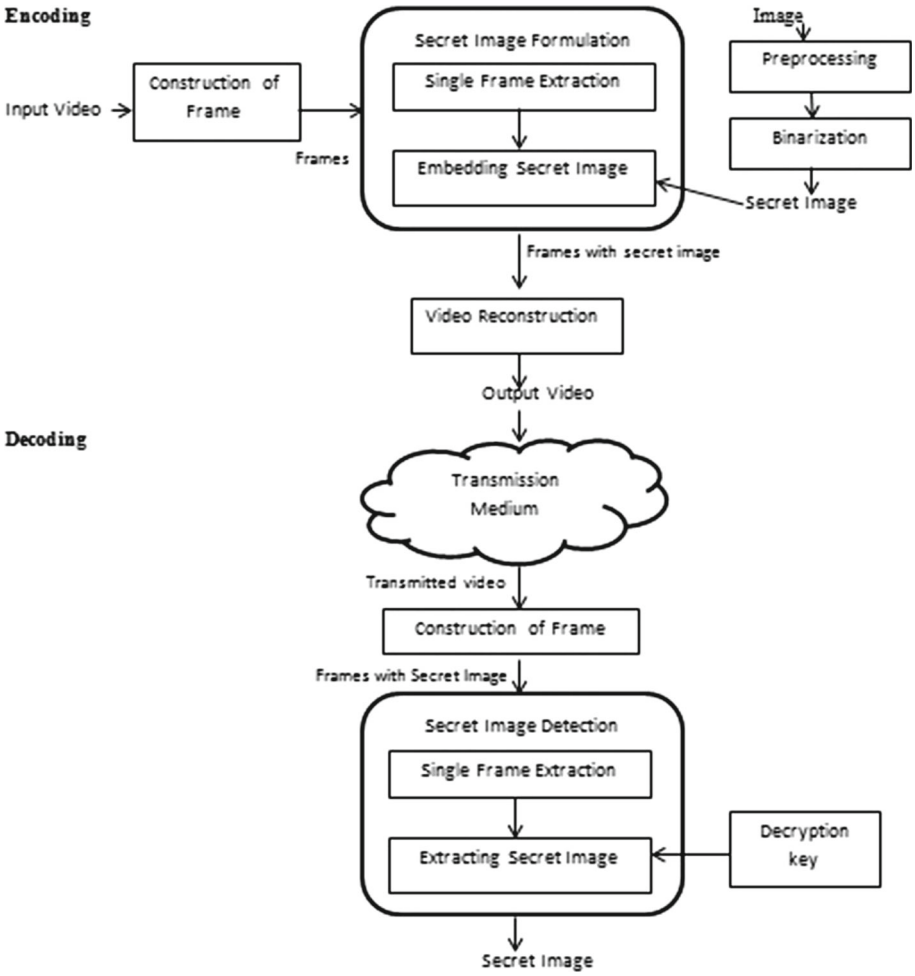

Fig. 7 Architecture of adaptive LSB encryption with randomized encoding

4 Proposed method: adaptive LSB encryption with randomized encoding

The Audio Video Interleave (AVI) video is used as the cover medium to carry the secret information. The secret image to be embedded is of bitmap (BMP) format. The bits in the original secret image are separated and placed at a particular video frame (Saikia and Thakur 2013). This technique of embedding bits in a secret frame itself provides a high level of encoding. The video object is then constructed to hold the frames and an uncompressed AVI video is obtained as the output video. This video can then be transmitted to the other party. The decoder converts the video s a series of frames and the encoded frame is decoded to get the secret information which is in the form of an image. The major reason for choosing the video format as AVI are:

1. Supported by most players and devices.
2. High Resolution frames

3. High Quality, as it supports synchronous audio-with-video playback.

The major reason for choosing the image format as BMP are:

1. High image Quality
2. Easy to edit or modify
3. No image loss throughout the process.

The proposed novel algorithm—adaptive LSB Encryption with Randomized Encoding is as follows:

1. The secret image and the high quality AVI video form the input for the encoding step. The secret image is pre-processed to remove noise and other discrepancies (Saikia and Thakur 2013).
2. The video is converted into a sequence of high resolution images called the frames. They are collected in a bitmap format. Each frame consists of three channels of RGB.
3. A particular frame is extracted and the secret image is resized to the maximum capacity of the extracted video frame so that the secret image fits within the video frame.
4. Each byte in the secret image is split into four pairs of bits (b1, b2, b3, b4) as shown in Fig. 8, to be encoded into the four quadrants of the frame (Saikia and Thakur 2013; Ni et al. 2014).
5. The last two bit locations in each byte of the frame is cleared so as to accommodate the bits from the secret image (Saikia and Thakur 2013). If the size of the secret frame is 1080 * 1920 then the number of bits to be encoded becomes 540 * 960 * 8 = 4,147,200 bits (Ni et al. 2014).
6. The total amount of data that can be accommodated in the extracted video frame becomes 1080 * 1920 * 2 = 4,147,200.
7. b1 is placed in the third quadrant, b2 is placed in the first quadrant, b3 is placed in the fourth quadrants and b4 is placed in the second quadrant (Mazurczyk and Caviglione 2015a; Ni et al. 2014).
8. The frames are sorted in order and a Video Object is defined to link the frames as a video.
9. In the decoder's side, the encoded frame forms the input and its last two bit location in each frame is cleared so that it results in the original extracted frame.
10. Each byte in the encoded frame is segregated (Saikia and Thakur 2013) into four pairs of bits (b1, b2, b3, b4).
11. All the four pairs of bits of each byte are assembled together to form the secret image.

The last two bit locations in each byte of the frame is cleared so as to accommodate the bits from the secret image (Saikia and Thakur 2013). If the size of the secret frame is 1080 * 1920 then the number of bits to be encoded becomes 540 * 960 * 8 = 4147200 bits (Ni et al. 2014). The total amount of data that can be accommodated in the extracted video frame becomes 1080 * 1920 * 2 = 4,147,200. b1 is placed in the third quadrant, b2 is placed in the first quadrant, b3 is placed in the fourth quadrants and b4 is placed in the second quadrant (Mazurczyk and Caviglione 2015a; Ni et al. 2014). The frames are sorted in order and a Video Object is defined to link the frames as a video. In the decoder's side, the encoded frame forms the input and its last two bit location in each frame is cleared so that it results in the original extracted frame. Each byte in the encoded frame is segregated (Saikia and Thakur 2013) into four pairs of bits (b1, b2, b3, b4). All the four pairs of bits of each byte are assembled together to form the secret image.

The novelty of the proposed work is twofold. The video is taken and it is converted into a series of frames (Ozcan and Kemal 2017). The number of secret images that can be embedded inside the video is equal to the number of frames in the video taken. The difference between

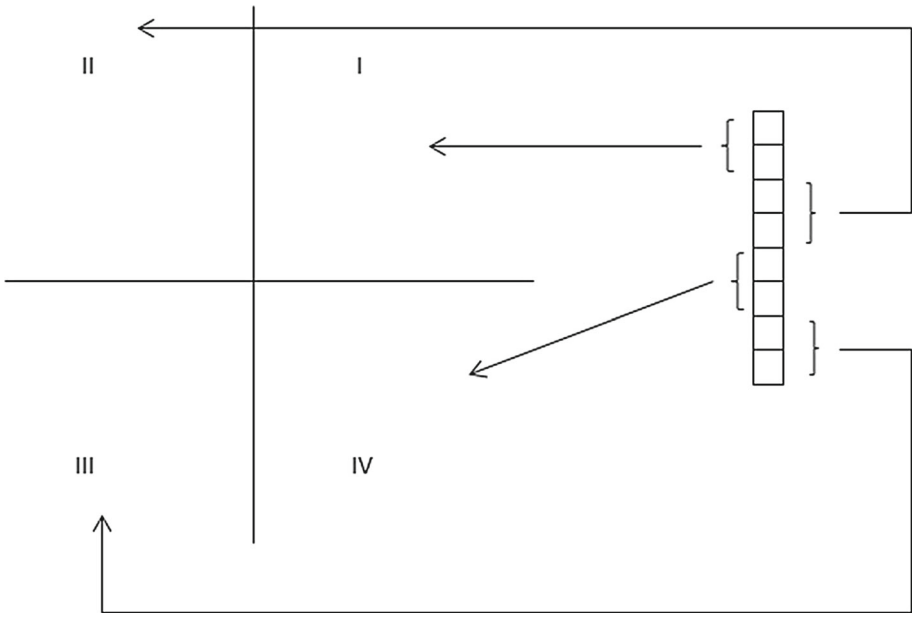

Fig. 8 Randomized algorithm to segregate bits in secret image

the encoded carrier frame taken and the same frame after decoding a secret image is calculated and tabulated in Sect. 5. It is clear the intruder cannot identify the frame in the video that actually contains the secret image without knowing the decryption key. Similarly, the secret image transmitted after decoding might lead to further processing. Therefore, it is important to maintain the quality of the secret image transmitted. Using the proposed method, the difference between the secret image before encoding (original image) and the decoded secret image is minimum ie., the data loss is minimum. Thus the proposed method outperforms the state-of-the art methods in encoding and decoding image in a video.

5 Experimental results

At the encoder side, the input video taken is an AVI video. All the n frames are processed and named in sequence in BMP format. A single image (frame) is selected from the n images to act as a cover medium. A secret image in BMP format is taken as the input to be embedded and resized to the maximum capacity of the video frame such that it fits completely in the selected cover frame (Li et al. 2014). The secret image bits are segregated into four pairs of bits using randomized algorithm. The last two significant bits of each pixel in the cover frame is replaced with each pair of bits from the secret image at different quadrants. The altered cover image is saved as BMP format image and the image stream is converted back into a video for the purpose of transmission. The transmission medium used is a virtual network that does not consist of any physical connection between the computer system and is implemented using network virtualization method (Fig. 9).

At the decoder's side, the input is the received AVI video file. This video file is converted into a series of frames. Using the stego key, the frame with the secret image is extracted and

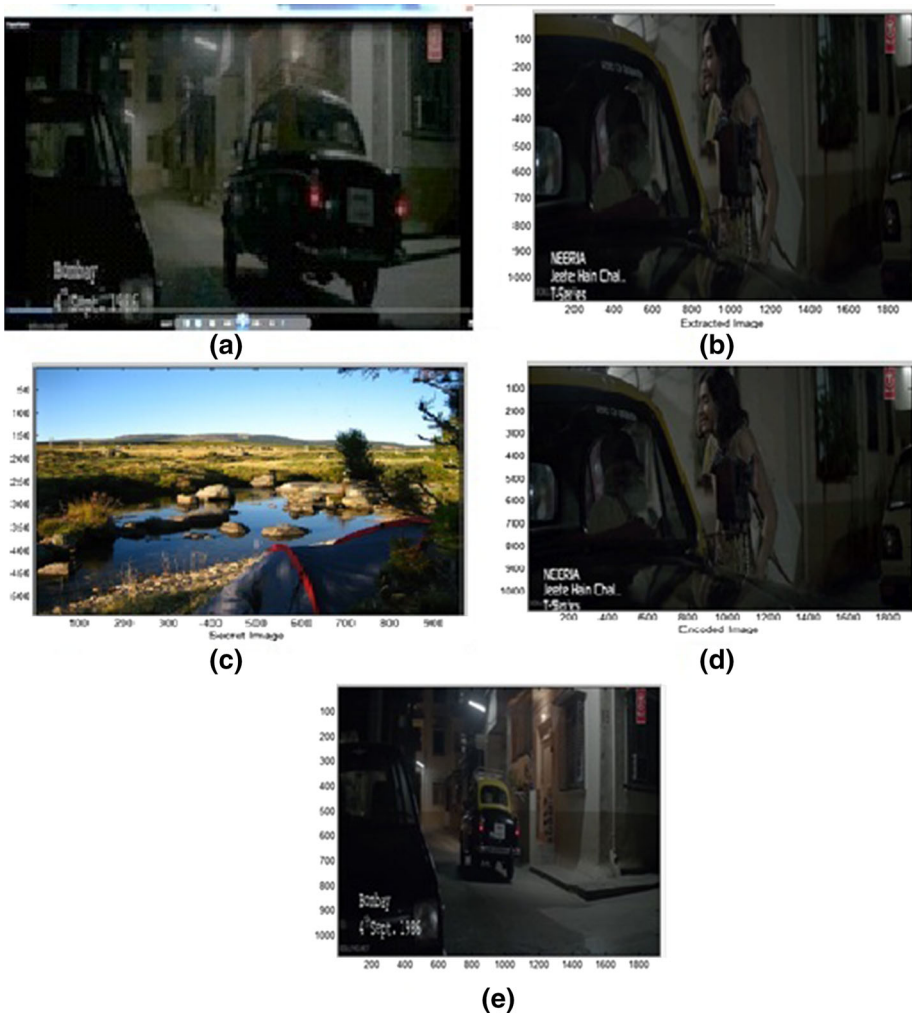

Fig. 9 **a** Sample AVI video, **b** Extracted frame, **c** Secret BMP image, **d** Encoded BMP frame, **e** Encoded video

decoded to get the four pairs of bits. The extracted pairs of bits are then assembled together to form the secret image and saved as BMP image format (Fig. 10). Two major requirements are: the size of the secret image encoded video should be the same as that of the original cover video and the quality of the secret image should be unaltered before and after encryption and decryption. To ensure this, the Mean Square Error (MSE) and the Peak Signal to Noise Ration (PSNR) are the metrics calculated. The MSE gives the cummulative squared error between the cover image and the stego image. The lower the value of MSE, the lower is the error and is calculated as,

$$MSE = \sum_{M,N} \frac{I_1(m,n) - I_2(m,n)^2}{M * N} \tag{1}$$

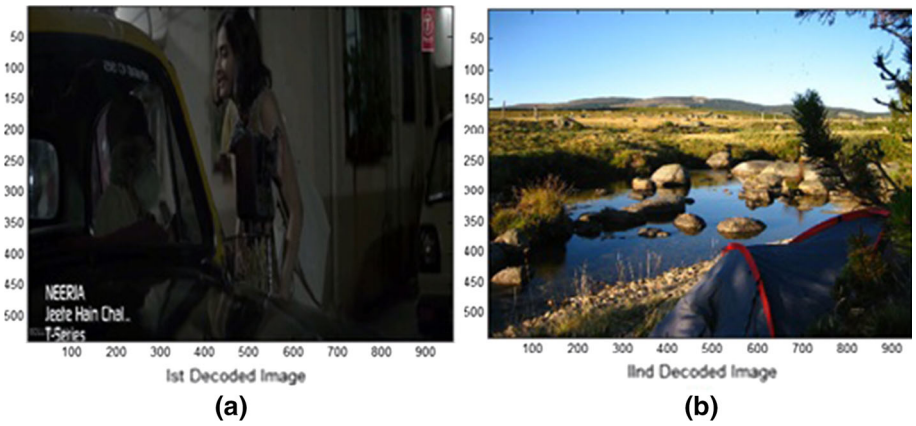

Fig. 10 **a** Extracted frame with secret image, **b** Secret BMP image

Table 1 Comparison of proposed method with state-of-the art methods for Lena and Baboon image (Bharadwaj and Sharma 2016; Ozcan and Kemal 2017)

Technique	MSE (in dB)	PSNR (in dB)	MSE (in dB)	PSNR (in dB)
One bit LSB	0.01	74.14	0.5239	50.97
Two bit LSB	0.0013	76.73	0.3074	53.28
4 bit LSB	0.0033	72.96	1.4405	46.57
Proposed method	0.0085	78.847	0.0650	60.035

where $I_1(m, n)$, intensity of pixel in cover image; $I_2(m, n)$, intensity of pixel in stego image.

M * N: size of image The PSNR calculates the statistical difference between the original secret image before encryption and the secret image after decryption using the formula,

$$PSNR = 10\log_{10}\frac{255^2}{MSE}\text{db} \tag{2}$$

where MSE: Mean Square Error of the secret image before and after decryption. The MSE and PSNR values of the video and image are compared after using our proposed approach and the size of the video or the quality of the secret image embedded remain unaltered.

The Lena image is taken as the carrier image and a sample image is encoded into it using all the exisiting methods and the proposed method. The difference between the original image (before encoding the secret image) and the encoded image (after encoding the secret image) is calculated. The same experiment is also done considering the Baboon image as the carrier image and a sample image as the secret image. Table 1, shows the comparison of the proposed approach with the existing state-of-the art approaches One bit LSB, Two bit LSB and Four bit LSB techniques for two different images. The comparison is made in terms of the performance metrics MSE and PSNR.

It is evident from the table that the difference between the original video frame and the encoded video frame is less (lower MSE) in the proposed method when compared to the existing methods, thus making it is impossible for an intruder to suspect any change in the encoded video. This suggests that the proposed work is novel and it outperforms the state-

Table 2 PSNR (in dB) for different images between the encoded image and the decoded image

PSNR (in dB) between encoded secret image and decoded secret image				
Lena	Baboon	Barbara	Cameraman	Goldhill
94.14	90.92	93.69	93.268	91.75

of-the art algorithms in terms of MSE and PSNR which are standard metrics for judging the image quality.

Apart from the carrier image, the secret image may contain secure data that can be further processed for various purposes. Therefore, it is important to maintain the quality of the secret image during transmission. A secret image is taken and the difference between the secret image before encoding (original image) and the secret image obtained after decoding (reconstructed image) is calculated. The experiment is repeated for five different images and the results are tabulated in Table 2. This table further ensures that the proposed work does not compromise also on the quality of the secret image embedded.

6 Conclusion

The proposed work is a novel data hiding technique that can be applied to an AVI video to insert a secret image within one of the frames of the video. When compared to the existing approaches, this technique does a two level encryption process that uses only two bit positions in a particular video frame. As a result of placing the secret image in four different quadrants, the size of the video or the quality of the secret image remains same before and after encryption thus providing novelty. Information in the form of a video, whose size is compatible with the size of the carrier video can also be encoded. This proposition considerably increases the quality of the secret data that a video can carry securely. This can be further extended by modifying the positioning of the secret bits which will significantly increase the randomization quotient. A hashing function can also be proposed to hash the bits of the secret image onto the carrier frame. The secret image thus extracted can contain text or image thus branching into lot more research areas in the field of multimedia.

References

Bharadwaj, R., & Sharma, V. (2016). Image steganography based on complemented message and inverted bit LSB substitution. *Proceedia Computer Science*, *93*, 832–838.

Ghaemmaghami, S., Ahani, S., & Wang, Z. J. (2015). Sparse representation based wavelet domain steganography method. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, *23*(1), 80–91.

Li, B., Huang, J., Wang, M., & Tan, S. (2014). Investigation of cost assignment in spatial image steganography. *IEEE Transactions on Information Forensics and Security*, *9*(8), 1264–1277.

Manisha, S., & Sharmila T. S. (2016). Text frame classification and recognition using segmentation technique. In *2016 2nd international conference on applied and theoretical computing and communication technology (iCATccT)*, IEEE, Bangalore (pp. 662–666).

Mazurczyk, W., & Caviglione, L. (2015a). Information hiding as a challenge for malware detection. *IEEE Security and Privacy*, *13*(2), 89–93.

Mazurczyk, W., & Caviglione, L. (2015b). Steganography in modern smartphones and mitigation techniques. *IEEE Communication Surveys Tutorials*, *17*(1), 334–357.

Ni, J., Guo, L., & QingShi, Y. (2014). Uniform embedding for efficient JPEG Steganography. *IEEE Transactions on Information Forensics and Security*, *9*(5), 814–825.

Ntalianis, K., & Tsapatsoulis, N. (2016). Remote authentication via biometrics: A robust video object steganographic mechanism over wireless networks. *IEEE Transactions on Emerging Topics in Computing*, *4*(1), 156–174.

Ozcan, C., & Kemal, T. (2017). Comparison of LSB image steganography technique in different color spaces. In *2017 international artificial intelligence and data processing symposium (IDAP)* (pp. 1–6). IEEE.

Saikia, M., & Thakur, V. (2013). Hiding secret image in video. In *International conference on intelligent systems and signal processing*, ISSP (pp. 150–153).

Sarreshtedari, S., & Akhaee, M. (2013). One-third probability embedding: A new +1 histogram compensating image least significant bit steganography scheme. *IET Image Processing*, *8*(2), 78–89.

Wang, C.-M., & Wu, K.-C. (2015). Steganography using reversible texture synthesis. *IEEE Transactions on Imapge Processing*, *24*(1), 130–139.

S. Manisha is working as Assistant professor in Department of Computer Science and Engineering at Sri Sivasubramaniya Nadar College of Engineering, Chennai. She completed her B.E. Computer science and Engineering in the year 2010 under Anna University, Chennai and M.E. (CSE) from Anna university, Chennai in the year 2012. She is currently pursuing her Ph.D. in Anna University under the guidance of Dr. T. Sree Sharmila, Associate professor, SSN College of Engineering. Her area of interest is Image and Video processing, Document Analysis and she has published papers on Text detection and recognition in IEEE and Springer conferences.

T. Sree Sharmila is working as an Associate Professor in the Department of Information Technology, SSN College of Engineering, Chennai, India. She received her B.E. in Information Technology from Manonmaniam Sundaranar University, Tirunelveli, in 2003, M.E. in Computer Science and Engineering from Annamalai University, Chidambaram, in 2005 and Ph.D. from Anna University, Chennai, in 2003. She has published more than 30 papers in well renowned International and National Journals and Conferences and is guiding 9 research scholars under Anna University. Her areas of interest are image preprocessing, texture analysis, satellite, ical and underwater image analysis.