Circuits Syst Signal Process (2014) 33:3901–3919
DOI 10.1007/s00034-014-9830-5

Steganography Between Silence Intervals of Audio in Video Content Using Chaotic Maps

Muhammad Fahad Khan · Faisal Baig · Saira Beg

Received: 29 June 2013 / Revised: 17 May 2014 / Accepted: 22 May 2014 /
Published online: 17 June 2014

Abstract Steganography is the art of hiding data, in such a way that it is undetectable under traffic-pattern analysis and the data hidden is only known to the receiver and the sender. In this paper, new method of text steganography over the silence interval of audio in a video file is presented. In the proposed method, first the audio signal is extracted from the video. After doing audio enhancement, the data on the audio signal are steganographed using new technique and then audio signal is rewritten in video file. To enhance the security level, we apply chaotic maps on arbitrary text. Furthermore, the algorithm in this paper gives a technique which states that undetectable stegotext and cover text have same probability distribution and no statistical test can detect the presence of the hidden message. Moreover, hidden message does not affect the transmission rate of video file at all.

Keywords Video steganography · Voice steganography · Audio silence interval · Audio analysis · Chaotic maps · Signal processing

M. F. Khan (✉)
Foundation University Islamabad, Rawalpindi Campus, Rawalpindi, Pakistan
e-mail: mfahad.bs@gmail.com

F. Baig
Federal Urdu University of Arts, Science and Technology, Islamabad, Pakistan
e-mail: engr.fsl.baig@gmail.com

S. Beg
COMSATS Institute of Information Technology, Islamabad, Pakistan
e-mail: sairabegbs@gmail.com

1 Introduction

Steganography is the art of secret writing [10]. The word steganography originates from two Greek words i.e., "stegano" (which means covered) and "graphy" (which means writing). This method hides the private message within some carrier medium which is known as cover medium. Generally, carrier medium is video, audio, images, text, etc. Such method is different from cryptography, where message contents are masked rather than message itself [7].

Usually the stego algorithm finds the redundant bits in carrier medium and then replaces those bits with secret messages [10]. Steganography could be applied in various fields, such as military and industrial applications to copyright and intellectual property rights (IPR) [7].

A number of methods and tools are available which improvise the degree of security via hiding data approaches [25]. For text steganography, Texto application is used. Such application produces plain text after encoding the original message. Problem with such application is that they do not rely on line or word shifting, so there is a chance of producing plain text which invites investigation [22]. MP3Stego is one of the effective watermarking tools used for audio MP3 files. It also hides the arbitrary information and such process takes place at the heart of the layer III encoding process name in the inner loop [22,25].

The most common audio-based steganography techniques are parity coding, phase coding, spread spectrum, echo hiding and LSB, etc. [14,41]. Similarly, video-based steganography methods are also available. Generally, a video file is composed of audio and images, so techniques of images and sound can be applied on video files directly. The biggest advantage of video-based steganography is that long message/files could be easily hidden without being observed. The reason behind this is that video files have large size and normally it is just moving stream of images and sound [16,24].

This paper presents the video-based steganography technique. In this method, first the audio signal is extracted from the video after doing audio enhancement using Wiener filter to remove any background noise during recording. Then, we applies steganography method which used two methods: one method is used to convert secrete text into arbitrary characters and other method applies chaotic maps on arbitrary text. Finally, text data are steganographed between the silent intervals of audio signal, which one embedded in the video. The algorithm in this paper focuses on hiding information over silence interval of video; Video that carry hidden information and the video that do not carry hidden information cannot be statistically distinguished and also transmission rate of messages with hidden information is not affected at all.

The rest of the paper is organized as follows. Sections 2 and 3 present the related work and proposed methodology, respectively. Section 4 is about results and discussions and lastly we conclude the paper.

2 Related Work

Video and image steganography techniques belong to two main categories called spatial and temporal domains. Stanescu et al. [34] presented a steganographic technique

to hide data in DCT coefficient using spatial characteristic of video. Spatial domain-based techniques usually use intensity of pixels in LSB positions to hide secrete message. Usually, LSB-based methods are easily prone to attack [19]. Few authors used temporal features of video to hide information. Fang et al. [18] proposed a steganographic method, in which motion vectors with large magnitude are selected for data hiding. Then, compute phase angle of these motion vector; data are hidden in horizontal component of motion vector for acute angle and for the obtuse angle data are embedded in the vertical component.

Video steganography methods can also be categorized into two classes: one is about hiding information in raw video and then compressing it later [5,6]; the other is hiding information directly in the compressed video files [7,40].

Advance LSB video steganography is presented in [10]. In this author describes the hash-based LSB technique. In this method, they divide the 8 bits secret message as 3, 3, 2 and embed them into the RGB pixel values of the cover frames, respectively where insertion position in LSB is selected through hash function. Results are analyzed in terms of peak signal to noise ratio (PNSR) and mean square error (MSE). Moreover, they also calculate image fidelity (IF) as well.

A secure compressed video steganography method is proposed in [7]. Proposed method has four parts: video sequence parser, the scene change detector, the secret message embedded, and the video steg-analysis. In this method, DCT coefficients could be represented by 0 and 1. Positive even and negative odd are represented by 0, whereas positive odd and negative even are represented by 1. I-Frame pixel value variance with DCT coefficient is calculated and allotment of message is dependent upon the variance of each cover frame. They also perform steg-analysis and prove that their proposed algorithm works well and can be applied on compressed video without losing image degradation or quality. A dynamic cover generation is proposed in [31]. This paper provides video cover generation method which proves that utilization of personalized video cover can improvise data security.

3 Proposed Methodology

Figure 1 shows the system architecture of the proposed methodology. In this, first we derive a method which extracts the audio signal from the video file. Now, apply audio enhancement procedure by using Wiener filter in order to remove background noise which may be stored during recording. At the end, proposed system performs steganography method which hides the text in the audio signal and reconstructs the video file again.

Step 1: Acquisition of Video Frames and Audio

At first we acquire the video file which is stored on the hard media or record it using camera. Then, read the given file in format (wmv, mpeg, avi, mov). After reading the video file, we extract the video frames and audio data and store it in different arrays as shown in Table 1.

Step 2: Removing the Background Noise

To increase the efficiency of the system, background noise must be removed from audio signal. Here, we use Wiener filter [12,18,37,38]. For implementing, the filter

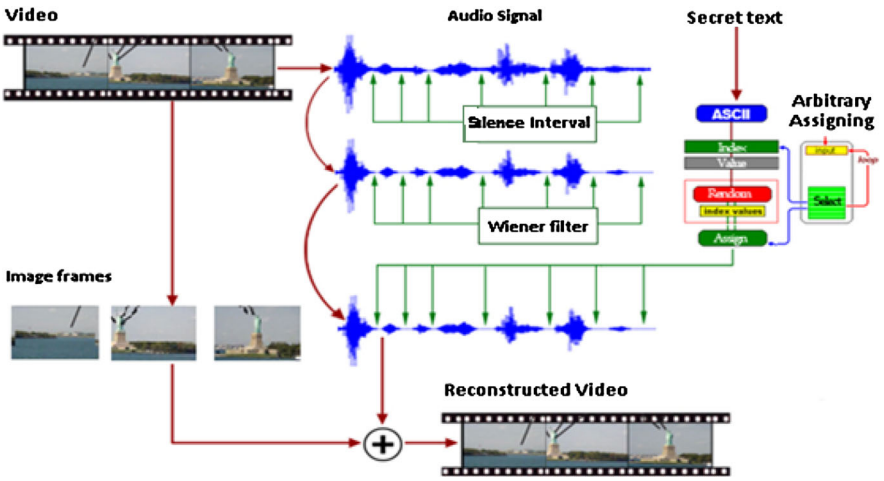

Fig. 1 System architecture

Table 1 Different arrays used to store video and audio data

Video		Audio	
Width	640	nrChannels	2
Height	480	Rate	44100
nrFramesTotal	167	Bits	16
Frames	[1*167 struct]	nrFramesTotal	35
Rate	30	Data	[245760*2 double]
Total duration (s)	5.566	Frames	[1*35 cell]
Times	[1*167 double]	Total duration	5.566
Skipped-frame	0	Times	[1*35 cell]

following Eqs. (1) and (2) has been used.

$$\emptyset(f) = \frac{|\text{Signal}(f)|^2}{|(\text{Signal}(f) + \text{Noise}(f))|^2} \tag{1}$$

$$\text{Since}|\,(\text{Signal}(f) + \text{Noise}(f))\,|^2 = |\text{Mixed}(f)|^2 \text{ we get}$$

$$\emptyset(f) = \frac{|\text{Signal}(f)|^2}{|\text{Mixed}(f)|^2} \tag{2}$$

where $\text{signal}(f)$, $\text{noise}(f)$, $\text{mixed}(f)$ are the Fourier transform of the signal $\text{signal}(t)$, $\text{noise}(t)$, $\text{mixed}(t)$, respectively. The following Fig. 2 shows the output after applying Wiener filter.

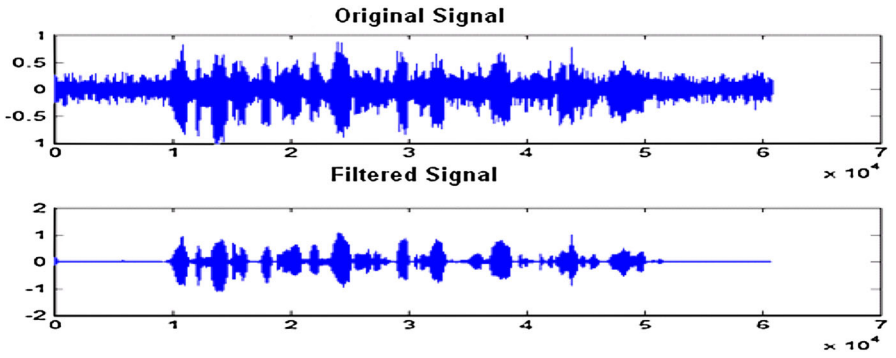

Fig. 2 Background noise removal of input signal using Wiener filter

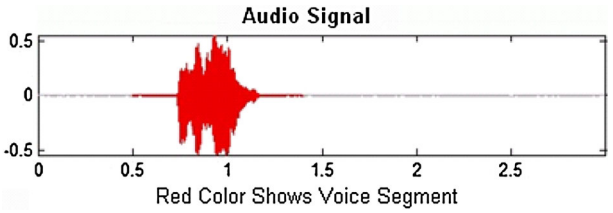

Fig. 3 Voice segments of audio signal

Step 3: Feature Extraction from Audio Signal

a. Root Mean Square Energy (RMS)and Audio activity

After the removal of noise, the signal is further subjected for feature extraction, the first part which came in feature extraction is to find voiced segment of the signal and in order to find voiced segment of the signal we first have to divide this signal in to small parts, which is done by the process of windowing. Windowing is done to reduce the effect of the spectral artifacts. A Hamming window [29], which tapers at its edges rather than having a sharp discontinuity, introduces fewer artifacts and is, therefore, used. Equation (3) for Hamming window is shown below.

$$W[k+1] = 0.54 - 0.46\cos\left(\frac{k}{n-1}\right), \quad k = 0, 1, \ldots, n-1 \tag{3}$$

After applying Hamming window we find voiced segment over the entire range of audio signal by calculating energy of the signal. So after finding energy now it is easy to distinguish between voiced and unvoiced segment of audio as shown in Fig. 3.

The energy of the signal is computed using the following Eq. (4) [26].

$$E(k) = \sum_{n=1}^{N} |m_k(n)|^2 \tag{4}$$

where $E(k)$ is the energy of the signal and $m_k(n)$ is audio sample of the k-th frame, of length N. After finding energy for audio signal we found RMS energy of input audio signal.

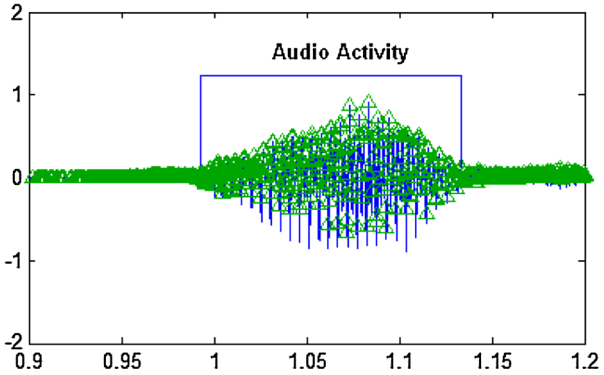

Fig. 4 Input signal audio activity with peak plots

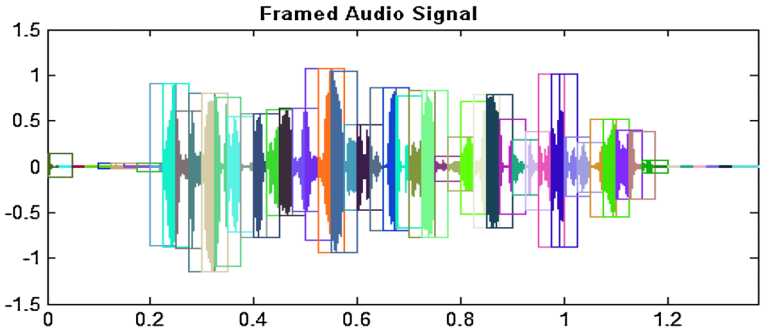

Fig. 5 Framed audio signal

As by finding energy now it is easy to distinguish between voiced and unvoiced segment of audio so we apply audio activity to find where audio level is active to separate voiced and unvoiced segment, Fig. 4 shows audio activity with peaks plot.

b. Audio Segmentation

Now it is easy to extract the silence zone from audio signal. In this step, the continuous audio signal has been blocked into frames of N samples as shown in Fig. 5. Each adjacent frames is being separated by M ($M < N$).

c. Cepstrum

After this we find cepstrum of input signal is shown in Fig. 6. The cepstrum is taken by taking inverse Fourier transform of signal estimated spectrum. This code for cepstrum is given below.

```
If (signal)
Spectral = Fourier transform of signal
Abs = square magnitude of spectral signal
Logged = Log of Abs
Inverse Fourier of Logged
```

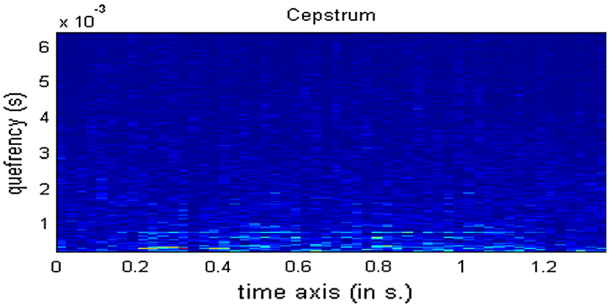

Fig. 6 Cepstrum

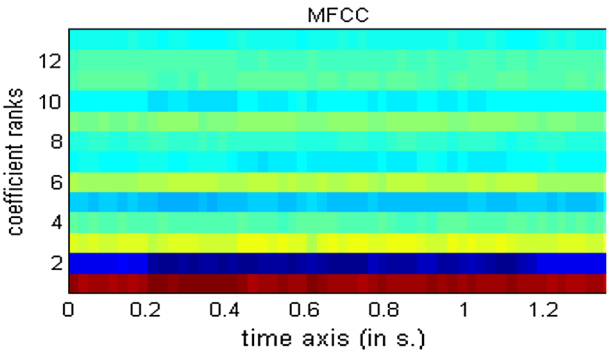

Fig. 7 Mel frequency cepstrum coefficient

d. Mel Frequency Cepstral Coefficient (MFCC)

Mel frequency cepstrum is the sound power spectrum of short term, while MFCCs are derived from cepstral representation of sound. MFCC of filtered signal is shown in Fig. 7. Mel frequency analysis is based on human perception of speech. MFCC is found as given below

If (signal)
Spectral = Fourier transform of the signal
Mapping = using triangular overlapping windows spectrum obtained above mapped onto the mel scale
Log = log of the powers at each of the mel frequencies
DCT = Discrete cosine transform of the list of mel log powers
The MFCCs are the amplitudes of the resulting spectrum

Step 4: Look Up Table

On the above feature extraction we defined a look up table which is used for method selection as shown in Table 2.

Step 5: Method Number and Arbitrary Characters Assignment

By referring method number we mean that a mathematical formula is used to convert a number from one value to another. So method number is used to convert data from

Table 2 Look up table

S. no.	MFCC	Cepstrum	RMS energy of signal	Method number
1	6.0007	0.2287	0.2400	1
2	6.1955	2.1219	0.2490	2
3	6.2703	0.5001	0.3500	3
4	7.5735	0.6323	0.4023	4
5	7.5320	1.5364	0.4810	5
6	8.1994	1.5816	0.3500	6
7	8.2072	2.5751	0.3400	7
8	9.0575	0.4014	0.8670	8

original form to arbitrary characters which are only known to users. These functions are only known to the sender and receiver. There are two parameters for each function one is X and other is Y.

1. X: ASCII order of the character
2. Y: Index of mathematical function

For example, let suppose "ZOO" is the text where $X = 26$ for Z, and $Y = 6$. If on look up table we have method number 6 and the mathematical Eq. (5) is:

$$f(x, y) = x^2 + 3x + y^2 \tag{5}$$

where "x" is the alphabetic order of the character, "y" is the method number, $f(x, y)$ is the resultant number to be converted in ASCII

Step 6: Chaotic Maps

After arbitrary conversion of input data it is further subjected for security and it is done by chaotic maps. To process this first the incoming data are converted in to string of binary numbers. This was then applied to chaotic maps which were chosen from method number selection. The listed maps are known to sender or receiver. Figure 8 shows chaotic maps, their auto correlation, and cross correlation factor.

After choosing the listed chaotic map, it is directly applied to binary string of data which was gathered from arbitrary character conversion process. The generated signal is shown in Fig. 9.

Using chaotic map will provide additional security to data, as chaotic maps further covert the data.

4 Protocol of the Proposed Audio Header

Header carries the key information about the stegotext, in the given method header is basically divided in two parts. The first part of the header carries the information about the second part of header by telling the receiver that at what place the second header is placed, like pointer, and it is placed at start of first audio activity zone. This is done due to the fact that parameter on which audio activity is computed is only known to

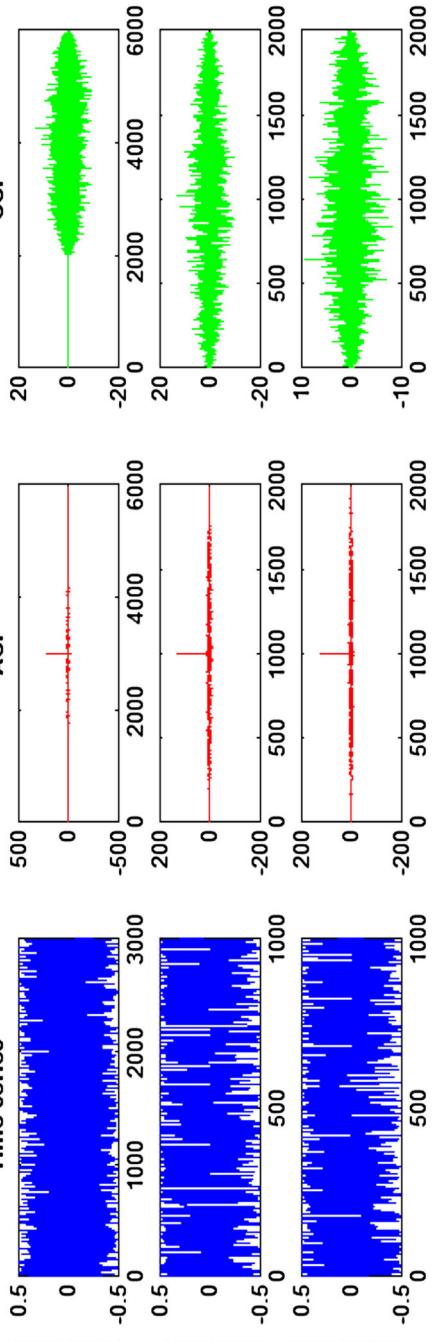

Fig. 8 Chaotic map with auto correlation factor and cross correlation factor

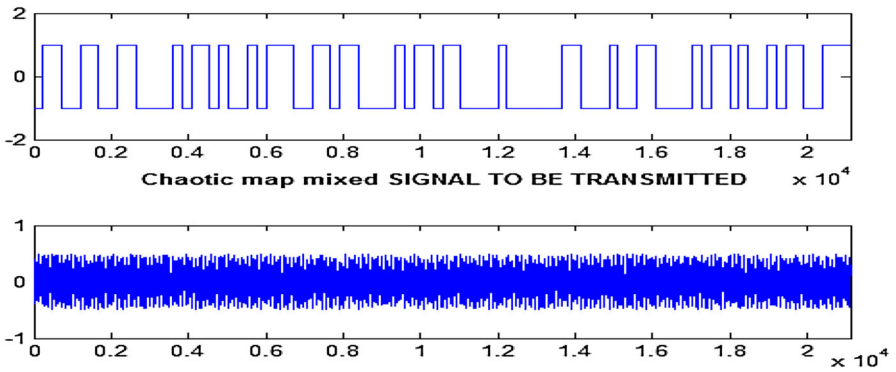

Fig. 9 Resultant signal generated by chaotic map

Table 3 First part of the header describe single parameter

Parameter (1)
Audio activity zone
Size: 2*double
Example: 0.0006

Table 4 Second part of the header describes these parameters

Parameter (1)	Parameter (2)	Parameter (3)	Parameter (4)
Silence zone number	Starting index of the stegotext	Method number	Arbitrary conversion count
Size: 1*double	Size: 1*double	Size: 1*double	Size: 1*double
Example: 0.00001	Example: 0.0000167	Example: 0.000025	Example: 0.00005

sender and receiver, so if an intruder tries to read stegotext, it will be difficult for him to compute the audio activity due to the variation of the parameters.

The second part of the header is placed anywhere in the audio activity zone, it contains the information about the stego message like silence zone index, starting index of the secret message, and method number. Each time when the stego message is transmitted, there are two parts of the header first is activity zone number which is converted to PCM data format [−1 1] before sending it. This is done to avoid the detection of header in traffic-pattern analysis. The second part of the header is also converted to standard PCM data format [−1 1] to avoid traffic pattern analysis. Tables 3 and 4 show the header formats and parameters.

4.1 Hiding Capacity

Hiding capacity of the given method depends upon the given relation as shown by the Eq. (6)

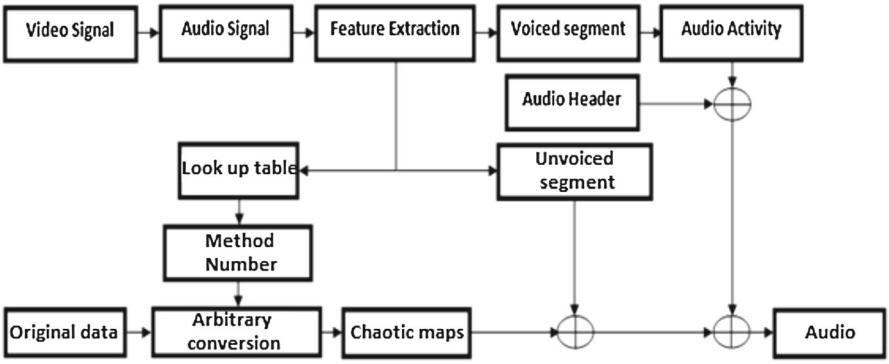

Fig. 10 Data hiding

$$\sum_{k=1}^{N} D(i) = \sum_{i=1}^{n} T(i) - \sum_{j=1}^{m} H(i) \tag{6}$$

$D(i)$ is the number of bytes available for stegotext
$T(i)$ total number of bytes in silence interval
$H(i)$ total header bytes

This shows the clear relation between the hidden message and the available space. If the given message is too large for the given silence zone, then the message is divided into multiple silence zones. At the start of each subdivided stegotext, we add a starting byte which gives the information about the next segment of the stegotext. Equation (7) shows the hiding capacity in a silence interval of audio signal which can be expressed as follows:

$$\mathrm{HCS} = \mathrm{TBS} - \mathrm{HBS} \tag{7}$$

HCS hiding capacity in a silence interval
TBS total bytes in silence interval
HBS header bytes in silence interval

5 Results and Discussion

For testing the proposed methodology, the application has been developed using MATLAB 7.0 tool. Video file and secret text are the major inputs. The result that is discussed in Table 5 is of "wmv" files, and the frame processing speed is 30 frames/s. As Table 5 shows that number of silence intervals may increase with video length, it totally depends upon the nature of input video so we can say that number of silence intervals is independent of video length e.g., test video 8 has smaller length as compared to video 10, but has larger silence interval than video 10. Similarly whose videos which have continues background music like video 9, 18, 21, 25, 24 generates less silence intervals and lowest hiding capacity and those videos which contain human speech without background voice like video 15, 19, 20, 17 generates large number of silence zones and high hiding capacity.

Table 5 Basic parameters and their respective values

Video test no.	Video duration (approximately)	Total no. of video frames	Total no. of silence intervals	Text hiding capacity in silence intervals (bytes)	Data set group
Video 1	5 (s)	167	3	728	Funny clip
Video 2	30 (s)	906	7	1816	Funny clip
Video 3	1 (min)	2016	11	2760	Funny clip
Video 4	1:30 (min)	3002	13	3364	Funny clip
Video 5	2 (min)	3811	17	4468	Funny clip
Video 6	2:30 (min)	4692	26	6166	Funny clip
Video 7	3 (min)	5462	24	6266	Funny clip
Video 8	20 (min)	40406	87	43891	Wildlife
Video 9	20 (min)	40296	63	42856	Wildlife
Video 10	27 (min)	59896	70	58654	Documentary film
Video 11	25 (min)	59652	61	57117	Documentary film
Video 12	1 (min)	20166	7	1914	National anthem
Video 13	1 (min)	20219	9	2064	National anthem
Video 14	1 (min)	20277	13	2294	Car racing clip
Video 15	1 (min)	20465	15	2423	Car racing clip
Video 16	10 (min)	203104	101	17130	Video lecture
Video 17	10 (min)	203003	87	16567	Video lecture
Video 18	10 (min)	202717	111	16902	Celebrity interview
Video 19	10 (min)	204116	94	18723	Celebrity interview
Video 20	3 (min)	6231	7	2760	Songs
Video 21	6 (min)	12462	18	6420	Songs
Video 22	10 (min)	204000	50	15900	Wrestling clip
Video 23	10 (min)	207023	54	16677	Wrestling clip
Video 24	3 (min)	5592	9	4136	Rock video clip
Video 25	3 (min)	5625	7	3954	Rock video clip

After separating the voice and video content of input file, we apply different filters in order to remove background or unnecessary sounds. In which Wiener gives us the best result for the input audio signal. Signal to noise ration which is found using signal to noise ratio was computed by using minimum mean square error (MMSE) algorithm. Table 6 below shows average noise spectrum.

Table 6 Average noise spectrum

S. no.	Filter applied	Average MMSE noise spectrum (dB)
1	Wiener [18–33,35–37]	1.7757
2	Spectral subtraction based on tracking a priori SNR using decision-directed method [8]	19.5980
3	Minimum mean square error method [15]	23.6685
4	Nonlinear spectral subtraction based on Berouti [4]	49.1053
5	Spectral subtraction based on Steven [35]	2.4103

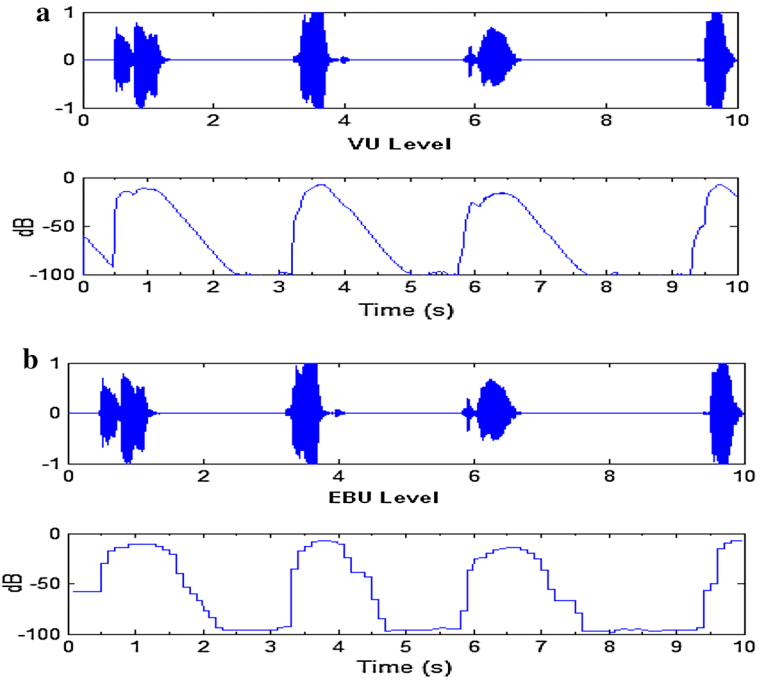

Fig. 11 **a** Volume unit (VU) of audio. **b** European broadcasting union (EBU) level

After filtration features were extracted for look up table, and then input signal is applied to be stenographic in audio signal. The data hiding is shown in Fig. 10 of section data hiding. There are three sections of proposed method:

Section 1: The first part is method number which is chosen on the extracted parameters from input audio signal. The method number is used to convert data from one form to other which are only known to sender and receiver. So this will help to keep message covert.

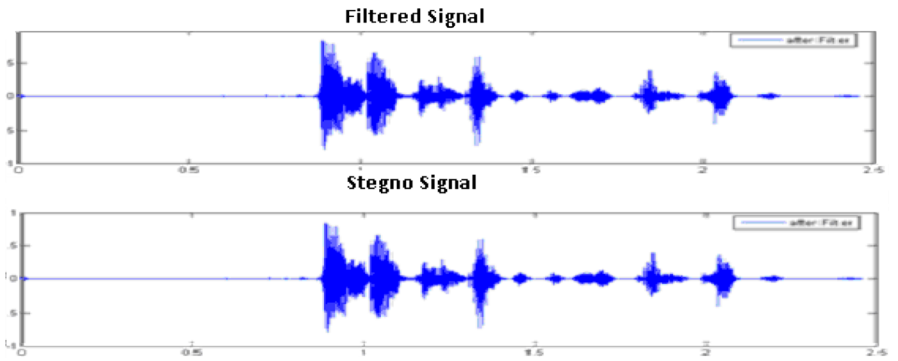

Fig. 12 Difference between filtered and stegano signal

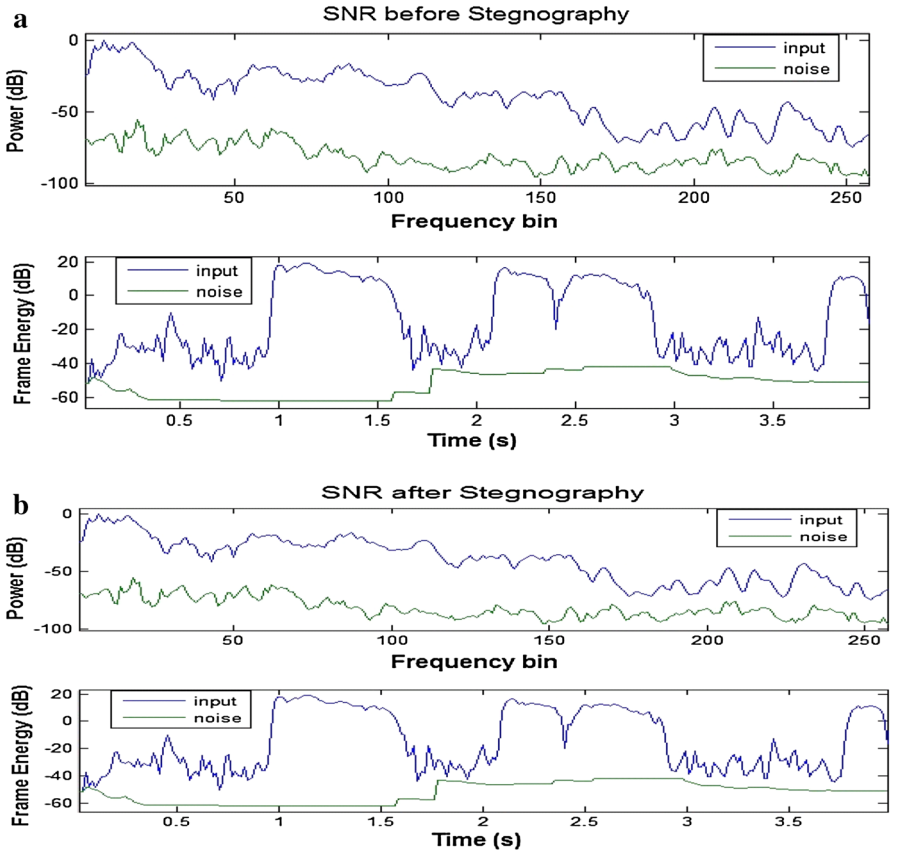

Fig. 13 **a** SNR before steganography. **b** SNR after steganography

Section 2: Second part in data security is header, which carries the key information about embedded stegno message. The header is hidden in two parts and hidden in audio activity zone. The audio activity for input audio is calculated

Table 7 Audio quality measures

Perceptual-domain measures	Time-domain measures	Result	Frequency-domain measures	Result
Bark spectral distortion (BSD)	Signal-to-noise ratio (SNR)	No change in signal to noise ratio it remains the same after stegano	Log-likelihood ratio (LLR)	Response null as no changes appeared after stegano as by formula $D = 2\sum_{i,j} k_{i,j} \log\left(\frac{n_{ij}}{m_{ij}}\right)$
Modified Bark spectral distortion (MBSD)	Segmental signal-to-noise ratio (SNRseg)	No change in signal to noise ratio of segmented audio	Log-area ratio (LAR)	Null response
Perceptual audio quality measure (PAQM)			Cepstral distance (CD)	Response null as we are not changing any cepstral distances
Measuring normalizing block 2 (MNB2)			Spectral phase distortion (SP)	No spectral phase distortion as we stego message on silence interval of audio
Weighted slope spectral distance (WSS)			Spectral phase-magnitude distortion (SPM)	Response null as no change in magnitude and spectral value

on user-defined energy thresholds and which are only known to them so it will be difficult for attacker to have an access to header of input data.

Section 3: During hiding data in unvoiced segment one thing which was done is that after all the mathematical formulation on the data, it is scaled at such a level that it cannot be heard or detected or the data are formulated in such a way that it shows no abnormality in listing. Generally, such problem occurred when parity coding is used [13]. So, for this propose we calculate loudness level of stego audio signal and scaled unvoiced segment much below −60 dB [14,41]. This lies in un-audible range. This is shown in Fig. 11a, b.

Lastly, the data have been steganographed and rewritten to the video file. Figure 12 shows the noise comparison of the original signal and the steganographed signal. After completing the steganography process, it was found that data size remains same and secret message data does not produce any additional noise in the signal (no increment found in data size). Furthermore, data are distributed and embedded in all over the intervals exist in input signal rather than phase coding in which secret message is encoded in the first signal segment only [13].

After this, signal to noise ratio was computed by using MMSE algorithm and the ratio between the original and stego signal was found same. Figure 13a and b shows the SNR comparison of two signals.

Table 8 Existing techniques

Technique	Methods	Embedding techniques	Advantages	Drawbacks	Reference
Temporal domain	Low bit encoding (least significant bit)	LSB of each audio sample is replaced with data sample	Easy and simple data hiding in target signal	Easy to extract and to destroy	[1,9,20,21]
	Echo hiding	Cover data by introducing echo signal in target signal	Resilient to lossy data compression algorithms	Low capacity and security	[23,32,39]
	Magnitude spectrum	Use frequency bands to hide data	Longer message to hide and less likely to be affected by errors during transmission	Low robustness to simple audio manipulations	[2,11]
Transform domain	Phase spectrum	Modulate the phase of the cover signal	Robust against signal processing manipulation and data retrieval needs the original signal	Low capacity	[27,28]
	Spread spectrum	Spread the data over all signal frequencies	Provide better robustness	Vulnerable to time scale modification	[3,17,30]
	Wavelet	Altering wavelet coefficients for embedding data	Provide high embedding capacity	Lossy data retrieval	[33,36]

6 Audio Quality Measures

In the end we perform some common audio quality measures on the proposed method. Here, we apply five different methods which are Bark spectral distortion (BSD), modified Bark spectral distortion (MBSD), perceptual audio quality measure (PAQM), measuring normalizing block 2 (MNB2), and weighted slope spectral distance (WSS). Table 7 shows the clear picture of audio quality measures.

After this we compare these techniques with some of existing techniques, this is shown in Table 8.

All these techniques have their drawbacks which are mentioned above; proposed technique data are added in such a way that it cannot be visually detected or by

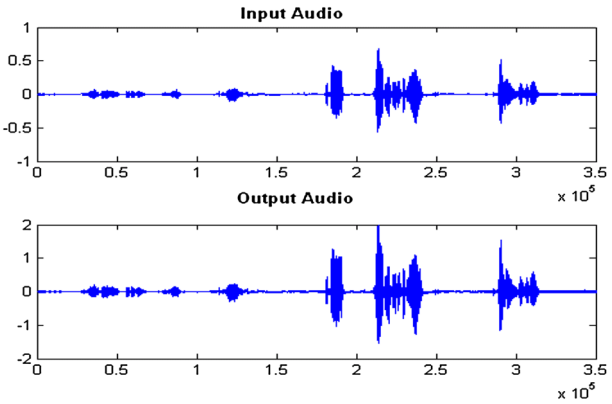

Fig. 14 Data hiding using discrete cosine transform (DCT)

some scaling in temporal or transform domain. A comparison of this is also shown with discrete cosine transform (DCT) technique in Fig. 14. As shown visually, there is some disturbance in signal level of audio signal so one can visually detect the presence of covert information, while in over proposed system which is shown in Fig. 12 there is no difference between original signal and stego signal.

7 Conclusion

In this paper a method of text steganography over silence zone interval of an audio signal embedded in video file is presented. In the proposed method, the audio signal is extracted from the video first. After doing speech enhancement using Wiener filter, the data have been steganographed on the audio signal and rewritten on video file again. In each silence interval, ASCII characters were arbitrarily assigned. Then apply chaotic maps over arbitrary characters to get better security. Cover text, Stegotext, and header are placed in the range of -1 to 1. As a result, the hidden message is undetectable under traditional traffic-pattern analysis. This technology has been implemented in MATLAB 7.0. Lastly, SNR values were also analyzed and it proved that proposed algorithm works well.

References

1. M.A. Ahmed, L.M. Kiah, B.B. Zaidan, A.A. Zaidan, A novel embedding method to increase capacity and robustness of low-bit encoding audio steganography technique using noise gate software logic algorithm. J. Appl. Sci. **10**, 59–64 (2010)
2. W. Andreas, J. Wurzer, C. Fabian, E. Piller, Pit stop for an audio steganography algorithm, in *Communications and Multimedia Security*, ed. by B. De Decker, J. Dittmann, C. Kraetzer, C. Vielhauer (Springer, Berlin, 2013), pp. 123–134
3. L. Ballesteros, M. Dora, J.M. Moreno, A real-time, speech-in-speech hiding scheme based on least significant bit substitution and adaptive key. Comput. Electr. Eng. **39**(4), 1192–1203 (2013)
4. M. Berouti, R. Schwartz, J. Makhoul, Enhancement of speech corrupted by acoustic noise. IEEE Int. Conf. Acoust. Speech Signal Process. **4**, 208–211 (1979)
5. D. Bhattacharyya, P. Dutta, M.O. Balitanas, T. Kim, P. Das, Hiding data in audio signal. Adv. Commun. Netw. Commun. Comput. Inf. Sci. **77**, 23–29 (2010)

6. J.J. Chae, B.S. Manjunath, in *Data hiding in video*. Proceedings of the 6th IEEE International Conference on Image Processing (1999), pp. 311–315
7. S.S. Christal Mary, Improved protection in video steganography used compressed video bit streams. Int. J. Comput. Sci. Eng. **2**(3), 764–766 (2010)
8. I. Cohen, Noise spectrum estimation in adverse environments: improved minima controlled recursive averaging. IEEE Trans. Speech Audio Process. **11**(5), 466–475 (2003)
9. R. Darsana, A. Vijayan, Audio steganography using modified LSB and PVD, in *Trends in Network and Communication*, ed. by D.C. Wyld, M. Wozniak, N. Chaki, N. Meghanathan, D. Nagamalai (Springer, Berlin, 2011), pp. 11–20
10. K. Dasgupta, J.K. Mandal, P. DuttaL, Hash based least significant bit technique for video steganography (HLSB). Int. J. Secur. Priv. Trust Manage. **1**(2), 1–11 (2012)
11. F. Djebbar, B. Ayad, K. Abed-Meraim, H. Hamam, Unified phase and magnitude speech spectra data hiding algorithm. Secur. Commun. Netw. **6**(8), 961–971 (2013)
12. S. Doclo, M. Moonen, On the output SNR of the speech-distortion weighted multi-channel Wiener filter. IEEE Signal Process. Lett. **12**(12), 809 (2005)
13. P. Dutta, D. Bhattacharyya, T. Kim, Data hiding in audio signal: a review. J. Database Theory Appl. **2**(2), 1–8 (2009)
14. EBU Tec. 3342, Loudness range, a descriptor to supplement loudness normalization according to EBU technical recommendation R128. Geneva (2010)
15. Y. Ephraim, D. Malah, Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. IEEE Trans. Speech Signal Process. **32**(6), 1109–1121 (1984)
16. K. Fahad, S. Beg, Transference and retrieval of voice message over low signal strength in satellite communication. Innov. Syst. Softw. Eng. **8**(4), 293–299 (2012)
17. R. Fahimeh, T. Ma, M. Hempel, D. Peng, H. Sharif, in *An anti-steganographic approach for removing secret information in digital audio data hidden by spread spectrum methods*. IEEE International Conference on Communications (ICC) (2013), pp. 2117–2122
18. D.-Y. Fang, L. Chang, in *Data hiding for digital video with phase of motion vector*. IEEE International Symposium on Circuits and Systems (ISCAS) (2006), p. 4
19. J. Fridrich, R. Du, M. Long, in *Stag-analysis of LSB encoding in color images*. IEEE International Conference on Multimedia and Expo (2000), pp. 1279–1282
20. T. Hui, J. Liu, S. Li, Improving security of quantization-index-modulation steganography in low bit-rate speech streams. Multimed. Syst. 1–12 (2013)
21. L. Jin, K. Zhou, H. Tian, in *Least-significant-digit steganography in low bit rate speech*. 47th IEEE International Conference on Communications (ICC) (2012), pp. 1133–1137
22. J.C. Judge, *Steganography: Past, Present and Future* (SANS Institute Info Sec Reading Room, version 1.2f. 2001)
23. V. Korzhik, G.M. Luna, I. Fedyanin, in *The use of wet paper codes with audio watermarking based on echo hiding*. IEEE Federated Conference on Computer Science and Information Systems (FedCSIS) (2012), pp. 727–732
24. J.R. Krenn, *Steganography and Steganalysis* (2004)
25. M.L. Mat Kiah, B.B. Zaidan, A.A. Zaidan, A. Mohammed Ahmed, S. Hasan Al-bakri, A review of audio based steganography and digital watermarking. Int. J. Phys. Sci. **6**(16), 3837–3850 (2011)
26. S.K. Mitra, *Digital Signal Processing: A Computer Based Approach* (McGraw-Hill, New York, 2006)
27. M. Nutzinger, J. Wurzer, in *A novel phase coding technique for steganography in auditive media*. IEEE Sixth International Conference on Availability, Reliability and Security (ARES) (2011), pp. 91–98
28. N. Parab, M. Nathan, K.T. Talele, Audio steganography using differential phase encoding, in *Technology Systems and Management*, ed. by K. Shah, V.R. Lakshmi Gorty, A. Phirke (Springer, Berlin, 2011), pp. 146–151
29. J. Proakis, D. Manolakis, *Digital Signal Processing, Principles, Algorithms and Applications*, 2nd edn. (Macmillan Publishing Company, New York, 1992)
30. A.R. Remya, M.H. Supriya, A. Sreekumar, A novel non-repudiate scheme with voice feature marking, in *Computational Intelligence, Cyber Security and Computational Models*, ed. by G. Sai Sundara Krishnan, R. Anitha, R.S. Lekshmi, M. Senthil Kumar, A. Bonato, M. Graña (Springer, India, 2014), pp. 183–194
31. V. Sampat, K. Dave, J. Madia, P. Toprani, in *A novel video steganography technique using dynamic cover generation*. National Conference on Advancement of Technologies—Information Systems &

Computer Networks: Proceedings published in International Journal of Computer Applications (2012), pp. 26–30
32. A.M. Shiddiqi, T. Priambadha, B.A. Pratomo, *Echo Data Hiding Steganography and RSA Cryptography on Audio, Media* (2012)
33. R. Siwar, D. Guerchi, S.A. Selouani, H. Hamam, Speech steganography using wavelet and Fourier transforms. EURASIP J. Audio Speech Music Process. **1**, 1–14 (2012)
34. D. Stanescu, M. Stratulat, B. Ciubotaru, D. Chiciudean, R. Cioarga, M. Micea, in *Embedding data in video stream using steganography*. IEEE 4th International Symposium on Applied Computational Intelligence and Informatics (2007), pp. 241–244
35. B. Steven, Suppression of acoustic noise in speech using spectral subtraction. IEEE Trans. Acoust. Speech Signal Process. **27**(2), 113–120 (1979)
36. D.T. Sudeep, S. Erandole, in *Extended performance comparison of tiling based image compression using wavelet transforms & hybrid wavelet transforms*. IEEE Conference on Information & Communication Technologies (ICT) (2013), pp. 1150–1155
37. N. Wiener, *Extrapolation, Interpolation and Smoothing of Stationary Time Series with Engineering Applications* (Wiley, New York, 1949)
38. N. Wiener, R.E.A.C. Paley, *Fourier Transforms in the Complex Domains* (American Mathematical Society, Providence, 1934)
39. Y. Xiang, D. Peng, I. Natgunanathan, W. Zhou, Effective pseudo noise sequence and decoding function for imperceptibility and robustness enhancement in time-spread echo-based audio watermarking. IEEE Trans. Multimed. **13**(1), 2–13 (2011)
40. C. Xu, X. Ping, T. Zhang, in *Steganography in compressed video stream*. Proceedings of the First International Conference on Innovative Computing, Information and Control (2006), pp. 269–272
41. W.A. Yost, M.C. Killion, Hearing thresholds. Encycl. Acoust. **3**, 1545–1554 (1997)