

An efficient multi-bit steganography algorithm in spatial domain with two-layer security

Nabanita Mukherjee (Ganguly)¹ · Goutam Paul²  · Sanjoy Kumar Saha¹

Received: 16 August 2017 / Revised: 23 December 2017 / Accepted: 24 January 2018 /
Published online: 7 March 2018
© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract Steganography is a very useful technique which aims at preventing loss of privacy during the process of data communication, especially over the internet. It can involve different forms of media like image, video (i.e., image sequence), audio etc. We propose a novel steganographic approach in spatial domain using pixel value differencing (PVD) or sample value differencing (SVD) technique and Galois field ($GF(2^8)$) operations in order to provide a two layered security for hiding message bits. Our method not only has a very high embedding capacity, but is also capable of withstanding statistical attacks. The proposed method embeds from 2 to 6 bits of the message per pixel in each image component, whereas it can embed a minimum of 6 bits and a maximum of 13 bits of message per sample in audio component at the expense of no perceivable distortion and loss of the cover media quality.

Keywords Image steganography · Audio steganography · Data hiding · Galois field · Symmetric bivariate polynomial · PVD

1 Introduction

The word “*steganography*” has been derived from the Greek word “*steganos*” which literally means “*covered*”. It is basically a technique of camouflaging information in a medium

✉ Goutam Paul
goutam.paul@isical.ac.in

Nabanita Mukherjee (Ganguly)
nabanitaganguly0@gmail.com

Sanjoy Kumar Saha
sks_ju@yahoo.co.in

¹ Department of Computer Science & Engineering, Jadavpur University, Kolkata 700 032, India

² Cryptology and Security Research Unit, Indian Statistical Institute, Kolkata 700 108, India

which prevents the detection of the hidden data [15]. The medium where the data is hidden is named as cover medium (e.g., image, video, audio file). LSB (Least Significant Bits) based steganographic techniques [16, 20, 36] are very old practice for hiding secret data where the embedding capacity as well as embedding efficiency are very low. But sometime security becomes a key issue for secret data transmission. Rather than directly flipping the LSBs of the pixel intensities some transformation functions may be used. For example, Lou et al. [23] have used a reversible histogram transformation in their LSB based steganographic method. It can resist the statistical steganalysis attacks. A steganography technique is presented in the work of Feng et al. [7] that works on binary image. It focuses on the minimization of texture distortion. As the number of bitplanes of a binary image is only one, hence selecting binary image as a cover media is not preferred. Admitting this fact, Lin et al. [22] have proposed a closed-loop iterative computing framework that optimizes picture quality.

A steganographic scheme based on $(2^{\eta+\theta} - 1)$ -ary notational system [18], where η denotes size of each pixel group and θ characterizes the inherent trade-off embedding rate and image quality, encodes a stream of bits with a cover pixel using codewords. But the major drawback of this algorithm is the capacity. Average embedding capacity here is less than 1 bpp, specially when the large number of pixels are considered in each non-overlapping set. Maya et al. [25] have presented an image steganography scheme based on Bitplane Complexity Segmentation and Integer Wavelet Transform.

Later, new techniques have been developed to alter multiple bits (LSB and higher weighted bits) of cover media. In recent times researchers have incorporated cryptographic concepts to enhance the security while passing secrete message bit stream. For example Hashim et al. [10] have proposed a data hiding technique which is combination of steganography and cryptography. Song et al. [39] have applied LSB matching method and used Boolean functions in stream ciphers. Lou et al. [24] have proposed a edge adaptive image steganography technique and it relies on LSB matching. Socek et al. [38] have proposed a digital video steganography technique in which one can disguise a particular video with the help of another video. Most of the works embeds the data sequentially. But it is susceptible to various statistical attacks [34]. To get rid of it, pixels for embedding are chosen randomly. Keys or seeds are required for such selection. Many steganographic algorithms are based on key sharing or block-code sharing. For example, Mstafa et al. [27] have proposed a secure video steganography method which is based on the concept of linear block code where for cover data, nine uncompressed video sequences are used and for secret message, a binary image logo is used. The positions of pixels of both cover videos and hidden message are randomly reordered with the help of a private key to improve the security of the system. Sharing of keys and/or the linear block-code is an essential criteria for such algorithms, but establishment of such priori agreement is an overhead. Paul et al. [30] have described a steganographic approach that does not require any key for random selection of pixels. Here the major concern again is the low capacity. The multi-bit steganographic technique of Petitcolas et al. [31] is capable of hiding at most 4 bits per pixel depending on the energy value. A multibit steganography has also been proposed by Mukherjee et al. [29]. It is applicable for audio steganography.

In modern era, Internet of Things, popularly known as IoT, has become very important. It can be used in several application areas like e-agriculture system [40], e-Health System [6]. In such applications, security of data is very important. Though Location-based Services (LBS) with IoT technology provides a lot of flexibility and conveniences, but there is a high probability to lose their privacy. Specially, the untrusted and malicious LBS servers with all clients' data may track the users in different ways or provide the personal information to the

third parties. One solution to preserve the user's location privacy is dummy location privacy-preserving (DLP) algorithm [41] or k-anonymity trajectory (KAT) algorithm [21], but personal data can be made more secure by imposing strong cryptographic/steganographic algorithms. Amin et al. [1] have proposed an architecture for distributed cloud environment that supports an authentication protocol using smartcard where the registered users are empowered to access all private data securely from each private cloud servers. Beside using AVISPA (Automated Validation of Internet Security Protocols and Applications) tool and BAN [3] (Burrows Abadi Needham) logic model, the authors have used cryptanalysis techniques in order to confirm that the protocol can withstand against all probable security threats. It is quite clear that for applications in IoT, steganography can play its role. Security and capacity, both the aspects are crucial for such cases.

It is observed that the capacity of LSB based steganography is very low. To overcome the same multibit steganography comes into picture. But most such schemes as suffer from statistical as well as steganalysis attacks when the embedding is done sequentially starting from the very first pixel of the image. This problem is addressed by selecting the pixels randomly from the image space. It introduces the overhead of sharing keys. In this paper, we propose a keyless multi-bit steganographic methodology applicable for both image and audio that can resist the first order statistical attacks and the state-of-the art blind steganalysis algorithm namely sample pair analysis [5]. Besides increasing the capacity, the security issues have been taken care of. In Section 2 the secret text bits to be embedded, have been encoded first based on the co-efficient of bivariate symmetric polynomial where operations are in $GF(2^8)$ (Section 2.2) and then this encoded text bits have been concealed in the image pixels using novel PVD (i.e., pixel value difference) technique and in audio samples as well by considering SVD (i.e., sample value difference) technique (Section 2.3). As the number of bits embedded in a pixel or audio sample is not fixed (it depends on the neighboring pixel intensities or sample values) it is difficult for the intruder to detect the hidden message. The rest of the paper is structured as follows. Section 2 details the methodology proposed. In Section 3, methodology is analyzed and theorems are framed. Experimental results are placed in Section 4 and concluded in Section 5.

We summarize our contributions of the work in the following subsection.

1.1 Contributions

Below we list our contributions one by one.

1. We propose novel steganographic algorithms for both image and audio based on PVD/SVD after encoding a message using bivariate symmetric polynomial based on $GF(2^8)$.
2. Our proposed algorithm is free from the key sharing overhead between receiver and sender.
3. Our method has high capacity - it hides 6 bits of message per pixel and 13 bits of message per audio sample.
4. We theoretically express embedding efficiency (Theorem 5). We have compared the outcome of our proposed method with the state-of-the art works (Section 4.5).
5. In order to ensure the security we have analyzed our algorithm through visual attacks, structural attacks and statistical attacks. For this we have included UIQI, PSNR, SNR, BER, etc.
6. We have shown that our algorithm withstands the popular blind steganalysis attacks (Section 4.8).
7. We have shown that our proposed algorithm is capable of withstanding the benchmark called StirMark.

Table 1 Text message arrangement

T_1	T_3	...	T_i	...	T_{S-1}
T_2	T_4	...	T_{i+1}	...	T_S

2 Proposed method

We propose a novel steganographic technique that enables two layered security in message hiding. At the first level the message to be hidden is encoded based on a bivariate symmetric polynomial operations are performed in $GF(2^8)$ (Section 2.2). At the next level the encoded data is then embedded by extending the idea of pixel value difference (PVD) or sample value difference (SVD) (Section 2.3). Note that we describe our embedding and extraction algorithm for one color plane only. For color image, the same process can be applied on each color plane separately. Before detailing the proposed methodology, in brief we revisit some preliminaries (Section 2.1).

2.1 Preliminaries

We discuss here the concept of Bivariate polynomial and Galois Field ($GF(2^8)$). Bivariate polynomial may be described as a polynomial which consists of two variables. It can be both symmetric as well as non-symmetric in nature. Let us consider a bivariate polynomial

$$G(x, y) = \sum_{i,j=0}^{k-1} C_{ij}x_iy_j. \tag{1}$$

The above polynomial is considered to be a bivariate symmetric polynomial, if it satisfies the criteria stated in (2).

$$G(x, y) = G(y, x),$$

i. e., $C_{ij} = C_{ji}$. (2)

The finite field $GF(2^8)$ [35] is generated by any irreducible polynomial of degree 8. For our purpose, we use the polynomial

$$g(x) = x^8 + x^4 + x^3 + x + 1. \tag{3}$$

The co-efficient of symmetric bivariate polynomial are utilized in encoding of text and all the operations are done in $GF(2^8)$.

2.2 Message encoding

In two layer security process, we first encode the text Consider a text message with S characters $T_1, T_2, T_3, \dots, T_S$ and then arrange it in column major order, shown in Table 1. Let, D_k denotes the corresponding ASCII values of the k -th character T_k . Hence, considering ASCII value of each character, the above matrix can be rewritten as in Table 2.

Divide this matrix into 2×3 non-overlapping cells. Here, we represent each cell as shown in Table 3. Generate a symmetric bivariate polynomial term (BPT) chart (Table 4).

Table 2 ASCII values of each character

D_1	D_3	...	D_i	...	D_{S-1}
D_2	D_4	...	D_{i+1}	...	D_S

Table 3 Smallest message block

D_i	D_{i+2}	D_{i+4}
D_{i+1}	D_{i+3}	D_{i+5}

Based on Table 4, construct the bivariate symmetric polynomial equation as shown in (1) and (2).

$$p(x, y) = D_i + D_{i+2}y + D_{i+4}y^2 + D_{i+2}x + D_{i+1}xy + D_{i+3}xy^2 + D_{i+4}x^2 + D_{i+3}x^2y + D_{i+5}x^2y^2. \tag{4}$$

Choose the values of $\{x_i : 0 < i \leq 3\}$ as follows.

$$x_i = (H \times W + i \times F) \pmod{2^8 - 1}, \text{ if } 0 < i \leq 3, \tag{5}$$

For an image, H and W are the height and the width respectively. F can be taken as any non-zero constant and it must be known to the recipient. For an audio clip, it is first decomposed into frames of equal size. W stands for the frame size and H is taken as 1. Depending on the size of the message to be embedded, a sequence of images or audio frames is required. In such cases, F stands for the image/audio index number in the sequence.

After substituting x by $\{x_i : 0 < i \leq 3\}$ in (4), we get three equations (6).

$$\begin{aligned} p(x_1, y) &= C_1 + C_2y + C_3y^2; \\ p(x_2, y) &= C_4 + C_5y + C_6y^2; \\ p(x_3, y) &= C_7 + C_8y + C_9y^2. \end{aligned} \tag{6}$$

Note that from the first equation, we take coefficients from all the powers of y . However, in the second equation we skip the term y^2 and in the third equation we skip the term y . This makes the three resultant equations (formed by the subset of selected coefficients) linearly independent. It enables unique the decoding of text at the time of retrieval. It has been detailed in Section 3 as D' and Theorem 3 in Section 4.

The co-efficients $[C_1, C_2, C_3, C_4, C_5, C_7]$ are stored in an array. Here, we perform all our calculations in $GF(2^8)$ using the irreducible polynomial in (3). On processing for every block of text change, 6-dimension co-efficient vector \vec{C}_j , where $j \in \{1, 2, 3, 4, 5, 7\}$ are taken to for 48-bit binary coefficient vector. It is the encoded version of the text block. The entire text message is divided into the blocks and each block is thus encoded. Instead of directly embedding the ASCII value of text data, the encoded bit stream is then embedded into the images (or in audio frames).

2.3 Embedding of encoded message

Once encoding is over, the next task is to embed the encoded stream into the cover (as next layer of security). Here we describe the embedding process for a single image. For a sequence of images, the same process is to be repeated for the next portion of the encoded stream. Similar approach may be followed in audio track also.

Table 4 BPT chart

D_i	$D_{i+2}y$	$D_{i+4}y^2$
$D_{i+2}x$	$D_{i+1}xy$	$D_{i+3}xy^2$
$D_{i+4}x^2$	$D_{i+3}x^2y$	$D_{i+5}x^2y^2$

Algorithm 1 Algorithm for embedding secret message bits into the cover image

- Input:** An image sequence with $F (>= 1)$ images, a text message T_1, T_2, \dots, T_S of known length S (in byte).
- Output:** The stego image sequence containing the secret data.
- 1 Create a group of $6N$ successive characters from text group where, N any positive number;
 - 2 Create a symmetric bivariate polynomial as shown in (4);
 - 3 choose $\{x_i : 0 < i \leq 3\}$ as shown in (5);
 - 4 substitute the value of $\{x_i : 0 < i \leq 3\}$ in (4) to determine $p(x_1, y), p(x_2, y), p(x_3, y)$ as shown in (6);
 - 5 Repeat steps 1 to 4 until the entire text message gets exhausted to get the coefficient vector \vec{C}_{j+9k} , where $j \in \{1, 2, 3, 4, 5, 7\}$ and $0 \leq k \leq \lfloor \frac{S}{6} \rfloor - 1$;
 - 6 Partition 3×3 non-overlapping blocks of pixels as shown in Table 5;
 - 7 $P^{(\beta)} \leftarrow$ pivot component as suggested in (7);
 - 8 $Q^{(\beta)} \leftarrow$ the difference of each element of 3×3 subcomponents of the image with $P^{(\beta)}$;
 - 9 Determine the number of bits to be embedded (n_m) based the (9);
 - 10 Calculate $b_{(m)}$, and $t^{(\beta)}$ from (12) and (11) respectively;
 - 11 **if** $t^{(\beta)}$ causes fall off **then**
 - if** $m = 0$ **then**
 - $t^{(\beta)} \leftarrow P^{(\beta)} - b_m$, if $\omega^{(\beta)} \geq P^{(\beta)}$;
 - $t^{(\beta)} \leftarrow P^{(\beta)} + b_m$, if $\omega^{(\beta)} < P^{(\beta)}$;
 - end**
 - else**
 - $n_m \leftarrow \log_2(R_m^{(l)} - R_{m-1}^{(l)})$;
 - $b_m \leftarrow d_{(m)} + R_{m-1}^{(l)}$;
 - $t^{(\beta)} \leftarrow P^{(\beta)} + b_m$, if $\omega \geq P^{(\beta)}$;
 - $t^{(\beta)} \leftarrow P^{(\beta)} - b_m$, if $\omega < P^{(\beta)}$;
 - end**
 - end**
 - 12 Replace the cover pixel under consideration with $t^{(\beta)}$;
 - 13 The cover audio break into 3×1 block of samples as per Table 6;
 - 14 Repeat steps 13 to 19 to calculate $P^{(\beta)}, Q^{(\beta)}, R_m^{(l)}, n_m, d_m, b_m$ and $t^{(\beta)}$;
 - 15 If $t^{(\beta)}$ causes fall off repeat step 15 using $P^{(\beta)}, R_m^{(l)}, n_m, d_m$;
 - 16 Write $t^{(\beta)}$ in the place of the sample under consideration;
 - 17 Output the transformed image sequence.

The image is divided into blocks of size 3×3 , indexed by β . The number of bits to be embedded in a pixel primarily depends on its intensity value and that of the pixel at the center of the corresponding block. Thus, it varies from pixel to pixel. Let $\omega_{i,j}^{(\beta)}$ be the intensity of the center pixel of the block β as shown in Table 5.

Pivot value for the block is calculated as in (7).

$$P^{(\beta)} = \omega_{i,j}^{(\beta)} - (\omega_{i,j}^{(\beta)} \bmod 4). \quad (7)$$

Table 5 3 × 3 cover image block

$\omega_{i-1,j-1}^{(\beta)}$	$\omega_{i-1,j}^{(\beta)}$	$\omega_{i-1,j+1}^{(\beta)}$
$\omega_{i,j-1}^{(\beta)}$	$\omega_{i,j}^{(\beta)}$	$\omega_{i,j+1}^{(\beta)}$
$\omega_{i+1,j-1}^{(\beta)}$	$\omega_{i+1,j}^{(\beta)}$	$\omega_{i+1,j+1}^{(\beta)}$

The embedding process will be such that $P^{(\beta)}$ remains unaltered (proof shown in section III). Each pixel of the block is compared with the pivot value, $P^{(\beta)}$ and their difference $Q_{i+k_1,j+k_2}^{(\beta)}$ is calculated as shown in (8).

$$Q_{i+k_1,j+k_2}^{(\beta)} = |\omega_{i+k_1,j+k_2}^{(\beta)} - P^{(\beta)}|; \tag{8}$$

where $k_1 \in \{-1, 0, 1\}$ and $k_2 \in \{-1, 0, 1\}$.

Based on $Q_{i+k_1,j+k_2}^{(\beta)}$, partition the size of bitstream for the pixel at $i + k_1, j + k_2$ is determined. The intensity range $[0, 255]$ is divided into number of non-uniform slots (in our case, it is 11) and $R_m^{(l)}$, the lower intensity value of m^{th} slot can be obtained as follows.

$$R_m^{(l)} = \begin{cases} 0, & \text{if } m = 0; \\ 4 \times \sum_{v=1}^m 2^{\lfloor \frac{m-v}{2} \rfloor}, & \text{if } 0 < m < 11. \end{cases} \tag{9}$$

Assume that $Q_{i+k_1,j+k_2}^{(\beta)}$ belongs to the m^{th} slot for some m .

Number of consecutive bits (n_m) are then taken from the binary coefficient vector, where

$$n_m = \log_2(R_{m+1}^{(l)} - R_m^{(l)}). \tag{10}$$

Let d_m be the decimal equivalent of the bit pattern of length n_m and it is the value to be embedded. The stego pixel value corresponding to the cover pixel $\omega^{(\beta)}$ is computed as follows.

$$t^{(\beta)} = \begin{cases} P^{(\beta)} + b_m, & \text{if } \omega^{(\beta)} \geq P^{(\beta)}; \\ P^{(\beta)} - b_m, & \text{otherwise.} \end{cases} \tag{11}$$

where b_m stands for

$$b_m = d_m + R_m^{(l)}. \tag{12}$$

Higher the intensity difference between the center pixel (hence the pivot) and pixel chosen for embedding, it is likely to be mapped onto a slot resulting into embedding of more number of bits from the binary coefficient vector. Embedding higher value to relatively high intensity will have less visual impact. Moreover, the embedding is contrast enhancing. It is worth to mention that the embedding process also maintains the uniformity of a smooth region as less bits are embedded.

It may be noted the b_m is obtained by adding d_m to the lower limit of the intensity slot where the difference between pixel intensity and the pivot value is mapped. As d_m is not added to actual intensity value of the pixels, possibility of artifacts is also reduced. n_m is chosen in a way that b_m will always lie within the range of the slot. b_m is obtained by incorporating the embedding effect on the difference between the pixel value and the pivot. To obtain the stego pixel value, finally b_m is added to or subtracted from the pivot value. It may result into fall off.

Fall off is said to take place if $t^{(\beta)}$ exceeds $2^8 - 1$ or is less than 0. $t^{(\beta)}$ is stored in the place of $\omega^{(\beta)}$ in the cover image if fall off does not occur. If Fall off takes place then we use the following approach to counter it. For $m = 0$, $t^{(\beta)}$ is adjusted as shown in (13).

$$t^{(\beta)} = \begin{cases} P^{(\beta)} - b_m, & \text{if } \omega^{(\beta)} \geq P^{(\beta)}; \\ P^{(\beta)} + b_m, & \text{otherwise.} \end{cases} \tag{13}$$

$t^{(\beta)}$ is then stored in the place of $\omega^{(\beta)}$ in the cover image. If $m > 0$, then n_m is recalculated for fall off pixel as shown in (14).

$$n_m = \log_2(R_m^{(l)} - R_{m-1}^{(l)}). \tag{14}$$

n_m number of consecutive bits are taken from the binary equivalent array of coefficient matrix and its corresponding decimal equivalence (d_m) is calculated. b_m is obtained by (15).

$$b_m = d_m + R_{m-1}^{(l)}. \tag{15}$$

$t^{(\beta)}$ is also recalculated by using (16).

$$t^{(\beta)} = \begin{cases} P^{(\beta)} + b^{(\beta)}, & \text{if } \omega^{(\beta)} \geq P^{(\beta)}; \\ P^{(\beta)} - b^{(\beta)}, & \text{otherwise.} \end{cases} \tag{16}$$

The above process to counter fall off is repeated until $t^{(\beta)}$ comes within accepted limits, i. e., 0 to $2^8 - 1$.

The process of embedding can also be applied on the audio track. The samples of cover audio is first broken into 3×1 blocks. A sample block is shown in Table 6.

The pivot value for the block is calculated using the following equation.

$$P^{(\beta)} = \Omega_i^{(\beta)} - (\Omega_i^{(\beta)} \bmod 4). \tag{17}$$

Generally, audio samples are of 16 bit. Hence total magnitude is divided into non-uniform slots as follows.

$$R_m^{(l)} = \begin{cases} 0, & \text{if } m = 0; \\ 64 \times \sum_{v=1}^m 2^{\lfloor \frac{m-v}{5} \rfloor}, & \text{if } 0 < m < 39; \\ 65472, & \text{if } m = 39; \end{cases} \tag{18}$$

Subsequently, similar approach of embedding as in case of cover image is deployed.

Table 6 3×1 audio cover block

$\Omega_{i-1}^{(\beta)}$
$\Omega_i^{(\beta)}$
$\Omega_{i+1}^{(\beta)}$

2.4 Retrieval

Algorithm 2 Algorithm for extracting secret message bits

Input: A steganised image sequence with $F (>= 1)$ images.

Output: The text message that is embedded in the image sequence.

- 1 Partition 3×3 blocks of pixels as per Table 7;
 - 2 $P^{(\beta)*} \leftarrow$ pivot component as suggested in (19);
 - 3 $Q^{(\beta)*} \leftarrow$ difference of each element of Table 7 with $P^{(\beta)*}$;
 - 4 $R_m^{(l)*} \leftarrow$ The greatest integer lesser than the difference obtained from (9);
 - 5 $n_m^* \leftarrow \log_2(R_{m+1}^{(l)*} - R_m^{(l)*})$;
 - 6 $d_m^* \leftarrow Q^{(\beta)*} - R_m^{(l)*}$;
 - 7 $b_m^* \leftarrow n_m^*$ bit binary equivalence of d_m^* ;
 - 8 Consider 3 consecutive sample values;
 - 9 Repeat steps 3 to 8 to calculate $P^{(\beta)*}, Q^{(\beta)*}, R_m^{(l)*}, n_m^*, d_m^*, b_m^*$;
 - 10 $\vec{B} \leftarrow b_m^* + b_m^*$ where + is concatenation;
 - 11 $\vec{C} \leftarrow$ 8 bit decimal equivalence of each element of \vec{B} ;
 - 12 $[x_1, x_2, x_3] \leftarrow$ The same values as during embedding;
 - 13 Compute the decimal equivalence of characters embedded using \vec{C}, x_1, x_2 and x_3 by using (23) to (28);
 - 14 Output the secret message bits.
-

The stego image is first broken into 3×3 blocks of pixel values.

The center pixel $\omega_{i,j}^{(\beta)*}$ is then considered and the pivot value for Table 7 is calculated using (19).

$$P^{(\beta)*} = \omega_{i,j}^{(\beta)*} - (\omega_{i,j}^{(\beta)*} \bmod 4). \tag{19}$$

Each pixel value of Table 7 is compared with $P^{(\beta)*}$ and their difference $Q^{(\beta)*}$ is calculated by using (20).

$$Q_{i+k_1, j+k_2}^{(\beta)*} = |\omega_{i+k_1, j+k_2}^{(\beta)*} - P^{(\beta)*}|, \tag{20}$$

where $k_1 \in \{-1, 0, 1\}$ and $k_2 \in \{-1, 0, 1\}$.

Now $Q_{i+k_1, j+k_2}^{(\beta)*}$ is checked in which range it falls by (9) and the greatest value of $R_m^{(l)*}$ lesser than $Q_{i+k_1, j+k_2}^{(\beta)*}$ is noted.

The number of bits that were embedded is obtained from (21).

$$n_m^* = \log_2(R_{m+1}^{(l)*} - R_m^{(l)*}). \tag{21}$$

The decimal equivalence d_m^* of the embedded bit-pattern is obtained by:

$$d_m^* = Q_{i+k_1, j+k_2}^{(\beta)*} - R_m^{(l)*}. \tag{22}$$

Table 7 3×3 stego image block

$\omega_{i-1, j-1}^{(\beta)*}$	$\omega_{i-1, j}^{(\beta)*}$	$\omega_{i-1, j+1}^{(\beta)*}$
$\omega_{i, j-1}^{(\beta)*}$	$\omega_{i, j}^{(\beta)*}$	$\omega_{i, j+1}^{(\beta)*}$
$\omega_{i+1, j-1}^{(\beta)*}$	$\omega_{i+1, j}^{(\beta)*}$	$\omega_{i+1, j+1}^{(\beta)*}$

Table 8 Retrieved binary sequence

b_1	b_2	...	b_i	...	b_n
-------	-------	-----	-------	-----	-------

Finally the binary sequence b_m^* is the binary equivalence (of length n_m^*) of d_m^* . Like this the entire image is processed to get the complete embedded binary sequence.

Then the audio component is similarly processed, albeit with 3×1 blocks of sample values as shown in Table 6.

The binary sequence obtained from image be concatenated with that from the audio as shown in Table 8. Eight consecutive and non-overlapping bits are then extracted from this array to convert them into their decimal values to generate the coefficient matrix as shown in Table 9.

Calculate x_1, x_2 and x_3 in the same manner as during embedding. Six consecutive and non-overlapping bits [$C'_1, C'_2 \dots C'_5, C'_7$] are then extracted from this array. These values are same as the values of [$C_1, C_2, C_3, C_4, C_5, C_7$] obtained from (6). The decimal values of each character of the original text message is obtained as follows. For $i = 1$ and $x = x_1$ in (4) we get (23) to (25). For $i = 1$ and $x = x_2$ in (4) we get (26) and (27). Finally for $i = 1$ and $x = x_3$ in (4) we get (28).

$$D'_1 + D'_3x_1 + D'_5x_1^2 = C'_1; \tag{23}$$

$$D'_3 + D'_2x_1 + D'_4x_1^2 = C'_2; \tag{24}$$

$$D'_5 + D'_4x_1 + D'_6x_1^2 = C'_3; \tag{25}$$

$$D'_1 + D'_3x_2 + D'_5x_2^2 = C'_4; \tag{26}$$

$$D'_3 + D'_2x_2 + D'_4x_2^2 = C'_5; \tag{27}$$

$$D'_1 + D'_3x_3 + D'_5x_3^2 = C'_7. \tag{28}$$

We call (23) to (28) as the *Retrieval System of Equations*. These equations are solved to obtain [$D'_1 \dots D'_6$]. The decimal values are converted to characters using ASCII to obtain the text messages.

3 Theoretical analysis of proposed method

Here we describe theoretical analysis of propose method by following theorems.

3.1 Pivot invariance theorem

Theorem 1 *The pivot value remains invariant before and after embedding data into the cover media.*

Proof From (7), we find that the pivot value is calculated by $P^{(\beta)} = \omega_{i,j}^{(\beta)} - (\omega_{i,j}^{(\beta)} \bmod 4)$. Thus, $P^{(\beta)}$ is divisible by 4 and $\omega_{i,j}^{(\beta)} = P^{(\beta)} + k$, where $k \in [0, 3]$. Hence, two bits of data are embedded into the center pixel.

At the time of retrieval the message, pivot value will be $P^{(\beta)*} = \omega_{i,j}^{(\beta)*} - (\omega_{i,j}^{(\beta)*} \bmod 4) = P^{(\beta)} + k - (P^{(\beta)} + k \bmod 4) = P^{(\beta)} + k - k = P^{(\beta)}$. □

Table 9 Coefficient matrix

C_1	C_2	...	C_i	...	$C_{n/8}$
-------	-------	-----	-------	-----	-----------

3.2 Range invariance theorem

Positive distortion is defined as the increase in pixel value after embedding while negative distortion is defined as the decrease in pixel value after embedding.

Lemma 1 *The maximum possible positive and negative distortion for embedding n bits of data in each of the pixels of the image is given in below.*

$$\max \delta_i^+ = 2^n - 1. \tag{29}$$

$$\max \delta_i^- = -2^n + 1. \tag{30}$$

Proof $\max \delta_i = 2^n - 1$ and $\max \delta_i^- = -2^n + 1$ because maximum possible decimal value that can be generated by n bits is $2^n - 1$. □

Theorem 2 (Range Invariance Theorem) *The range of $Q_{i+k,j+k}^{(\beta)}$ and $Q_{i+k,j+k}^{(\beta)*}$ remain invariant before and after embedding the secret data.*

Proof The maximum possible positive and negative distortion for embedding n bits of data in each of the pixels of the image is given in Lemma 1. $R_m^{(l)}$ is defined lower range of $Q_{i+k,j+k}^{(\beta)}$. Thus the maximum possible value of $Q_{i+k,j+k}^{(\beta)}$ will be $R_m^{(l)} + \max \delta_i + 1$. Now,

$$\begin{aligned} R_m^{(l)} + \max \delta_i + 1 &= R_m^{(l)} + 2^n - 1 + 1 \\ &\text{(from (29))} \\ &= R_m^{(l)} + 2^n \\ &= R_m^{(l)} + 2^{\log_2(R_{m+1}^{(l)} - R_m^{(l)})} \\ &= R_m^{(l)} + R_{m+1}^{(l)} - R_m^{(l)} \\ &= R_{m+1}^{(l)}. \end{aligned}$$

Hence, we see that that the lowest value of in any range $R_m^{(l)} + 1$ do not exceed the highest value $R_{m+1}^{(l)}$ even if maximum distortion occurs. □

3.3 Linear independence of retrieval system of equations

Next, we show that System of (23) to (28) yields a unique solution.

Theorem 3 *The System of (23) to (28) is consistent and has a unique solution in $D'_{i \in \{1,2,3,\dots,6\}}$.*

Proof The co-efficient matrix (D'_{coeff}) of the system shown in (23) to (28) can be written as:

$$D'_{coeff} = \begin{bmatrix} 1 & 0 & x_1 & 0 & x_1^2 & 0 \\ 0 & x_1 & 1 & x_1^2 & 0 & 0 \\ 0 & 0 & 0 & x_1 & 1 & x_1^2 \\ 1 & 0 & x_2 & 0 & x_2^2 & 0 \\ 0 & x_2 & 1 & x_2^2 & 0 & 0 \\ 1 & 0 & x_3 & 0 & x_3^2 & 0 \end{bmatrix}$$

We can write,

$$D'_{coeff} = \begin{bmatrix} D_{x_1} \\ D_{x_2} \\ D_{x_3} \end{bmatrix},$$

where,

$$D_{x_1} = \begin{bmatrix} 1 & 0 & x_1 & 0 & x_1^2 & 0 \\ 0 & x_1 & 1 & x_1^2 & 0 & 0 \\ 0 & 0 & 0 & x_1 & 1 & x_1^2 \end{bmatrix},$$

$$D_{x_2} = \begin{bmatrix} 1 & 0 & x_2 & 0 & x_2^2 & 0 \\ 0 & x_2 & 1 & x_2^2 & 0 & 0 \end{bmatrix},$$

$$D_{x_3} = [1 \ 0 \ x_3 \ 0 \ x_3^2 \ 0].$$

From (5), we find that $\{x_i | 0 < i \leq 3\}$ has been chosen in such a way that $x_i \in \{1, 2, \dots, 2^8 - 1\}$ are distinct. So any row of D_{x_i} is independent of any row of D_{x_j} , for $i \neq j \in \{1, 2, 3\}$. Thus, the rank of D'_{coeff} would be the sum of the ranks of D_{x_1} , D_{x_2} and D_{x_3} . Now D_{x_1} has 3 rows and hence its rank is at most 3. Since the submatrix formed by the first three columns of D_{x_1} has rank 3, D_{x_1} has rank exactly equal to 3. Similarly, the first two columns of D_{x_2} indicate that its rank is 2. The rank of D_{x_3} is trivially 1. Hence, the rank of D'_{coeff} is $(3 + 2 + 1) = 6$. Thus, the system is consistent and has a unique solution. \square

3.4 Proof of correctness

One important issue regarding any steganographic algorithm is to check whether the embedded secret message can be correctly extracted at the receiver end or not. We have proved it both theoretically and experimentally. Experimental result have been discussed in Section 4 and the theoretical proof is given below.

Theorem 4 *The embedded message bits can be extracted without any bit loss.*

Proof We have to prove that $d_{(m)}^* = d_{(m)}$. Here the two possible cases are:

Case 1: If $t^{(\beta)} > p^{(\beta)}$ then

$$d_{(m)}^* = Q^{((\beta)^*)} - R_{m^*}^{(l)} = \omega_{i+k, j+k}^{(\beta)^*} \sim P^{(\beta)^*} - R_{m^*}^{(l)} = \omega_{i+k, j+k}^{(\beta)^*} \sim P^{(\beta)^*} - R_m^{(l)}$$

(by Range Invariance Theorem) $= (t^{(\beta)} \sim P^{(\beta)^*}) - R_m^{(l)}$ (Since stego-pixel intensity is $t^{(\beta)} = \omega_{i+k, j+k}^{(\beta)^*} = (P^{(\beta)} + b_{(m)} - P_{(\beta)}^*) - R_m^{(l)}$ (Since $t^{(\beta)} > P^{(\beta)}$) $= b_{(m)} - R_m^{(l)}$ (Since $P^{(\beta)^*} = P^{(\beta)}$) $= d_{(m)} + R_m^{(l)} - R_m^{(l)} = d_{(m)}$)

Case 2: If $t^{(\beta)} < p^{(\beta)}$ then $d_{(m)}^* = Q^{((\beta)^*)} - R_{m^*}^{(l)} = \omega_{i+k, j+k}^{(\beta)^*} \sim P^{(\beta)^*} - R_{m^*}^{(l)} = \omega_{i+k, j+k}^{(\beta)^*} \sim P^{(\beta)^*} - R_m^{(l)}$ (by Range Invariance Theorem) $= (t^{(\beta)} \sim P^{(\beta)^*}) - R_m^{(l)}$ (Since stego-pixel intensity is $t^{(\beta)} = \omega_{i+k, j+k}^{(\beta)^*} = (P_{(\beta)}^* - P^{(\beta)} + b_{(m)}) - R_m^{(l)}$ (Since $t^{(\beta)} < P^{(\beta)}$) $= b_{(m)} - R_m^{(l)}$ (Since $P^{(\beta)^*} = P^{(\beta)}$) $= d_{(m)} + R_m^{(l)} - R_m^{(l)} = d_{(m)}$ the result follows. \square

3.5 Embedding efficiency

Embedding efficiency is the embedding strength of any steganographic algorithm against distortion that occurs due to concealing message bits into cover media. The embedding efficiency definition [8] appearing as follows:

Definition 1 Embedding efficiency of any steganographic algorithm can be defined as the expected number of message bits concealed per embedding change.

Theorem 5 *If n is the number of embedded bits per pixel of the cover media, then the maximum embedding efficiency of proposed method is $\frac{n \cdot 2^n}{2^n - 1}$*

Proof The possible values of distortions for concealing n bits message in 8-bits intensity value is given by $\{1, 2, \dots, 2^n\}$. Assuming uniform distribution, probability of change in pixel intensity is given by $\frac{2^n - 1}{2^n}$. Dividing n by this probability, we get embedding efficiency of our technique. \square

For any LSB based steganographic algorithm, the embedding efficiency is $\frac{1}{1/2} = 2$ [8]. The embedding efficiency for steganographic algorithms using random ternary symbols with uniform distribution in the media is 2.3774. The embedding efficiency of the works [8] and [28] are 4.4 and 5.33 respectively. Limiting n to 6 in our technique we get the embedding efficiency as 6.0952 (image) and 13.0015 (audio) which is higher than the said techniques.

4 Experimental results

Along with the analysis of the visual quality of the cover and stego (image or audio) we also analyze the strength of our method in terms of visual quality analysis, average embedding capacity, embedding efficiency etc. Our method is compared with several other existing methods. The superiority of this novel algorithm can be seen through the comparative study shown in Table 15.

Besides theoretical outcomes, we have thoroughly analyzed the results obtained by applying the proposed steganographic algorithm on the different types of images, audios and videos as well. In our experimental setup, we have considered standard images like lena, baboon, fruits. It is to be noted that a sequence of images may be required when the message size becomes very large. For such cases, we have considered the sequence of image frames extracted from the video clips like Architectures, Cartoon, Wildlife. Number of frames depends on size of the data to be embedded. So starting from a point consecutive frames are used to satisfy the need. However, in our experimental setup we have considered 100 of each categories of video clips, viz., Architectures, Cartoon, Wildlife mostly having approximately 100 frames of size 255×141 . In order to embed message bits in the audio clips we have divided the clip into frames consisting of 1024 samples. For audio, .wav files are considered. Image sequence is in uncompressed form. It is considered that the message is not so long to make the transmission of uncompressed image sequence prohibitive.

4.1 Visual perceptibility analysis

Our proposed method has been widely tested over several image and audio. In our experimental setup we find an insignificant change between the cover and its respective stego

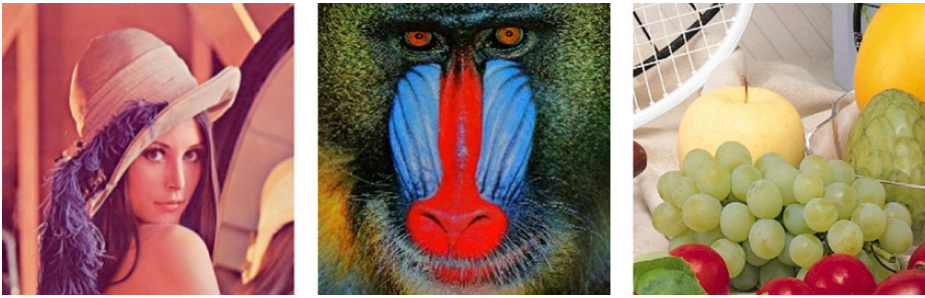


Fig. 1 Cover version of standard images, viz., lena, baboon and fruits (left to right)

version. Figures 1 and 2 shows original and corresponding stego version of standard images. Figures 3 and 4 shows one sample image from every image sequence and also the corresponding stego version. For audio, few samples of every audio sequence and also the corresponding stego version is given in Figs. 5 and 6.

4.2 Analysis of MSE, PSNR and SNR

Mean Square Error (MSE) as shown in (31) (where I represents the cover and I^* represents the stego image of size $W \times H$) shown in Tables 10 and 11 for images of four categories and standard images respectively.

$$MSE = \frac{1}{W \times H} \sum_{i=0}^{W-1} \sum_{j=0}^{H-1} [I(i, j) - I^*(i, j)]^2. \quad (31)$$

The values of Peak-Signal-to-Noise-Ratio (PSNR) as shown in (32)

$$PSNR = 10 \cdot \log_{10} \left(\frac{255^2}{MSE} \right). \quad (32)$$

Signal-to-noise ratio (SNR) is defined as the power ratio between a signal (meaningful information) and the background noise (unwanted signal), i.e.,

$$SNR = \frac{\text{signal power}}{\text{noise power}}.$$

The Signal-to-Noise-Ratio (SNR) for the audio frame is shown in Table 12.

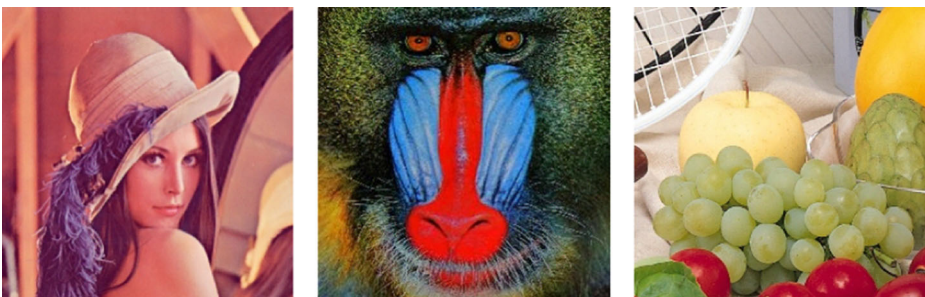


Fig. 2 Stego version of standard images, viz., lena, baboon and fruits (left to right)



Fig. 3 One of cover frame from each of architectural, cartoon and wildlife video (left to right)

4.3 Histogram analysis

A histogram is the graphical representation of distributed numerical image intensity values. Figure 7 shows the histograms of both cover and stego version of a sample image. We find negligible differences between cover and stego. There is no difference of histogram for sample cover lena and stego lena image.

4.4 Bitplane analysis

A bitplane of an image can be defined as a set of bits which corresponds to any particular bit position for the respective binary numbers that represents the image Fig. 8.

4.5 Average embedding capacity

Definition 2 Average embedding capacity is defined by the number of embedded bits per pixel. i.e.,

$$AEC = \frac{\text{Number of Embedded Bits}}{\text{Total Number of Pixels}}$$

Average embedding capacity for an image with x pixels can be formulated as,

$$\gamma = \frac{1}{x} \sum_{i=1}^x y_i,$$

where $y_i \Rightarrow$ Number of message bits embedded in the i -th pixel.

The average embedding capacity is one of the most important criteria to determine the efficiency of an algorithm. Embedding capacity (on an average) can be of several forms.

Bits per pixel (bpp) for the image component is calculated by the ratio of number of bits embedded to the number of pixels in the image while bits per sample (bps) for the audio frame is calculated by the ratio of number of bits embedded to the number of samples in the audio.

Our method has a very high embedding efficiency which is not achieved at the cost of visual impairing. The average embedding capacity of our method on different video categories is shown in Tables 13 and 14. The fact that our method proves its superiority over other methods in terms of average embedding capacity can be understood from Table 15.

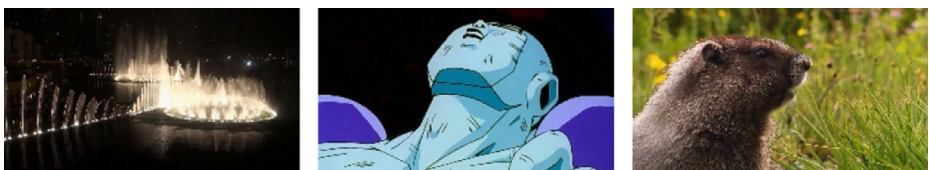


Fig. 4 One of stego frame from each of architectural, cartoon and wildlife video (left to right)

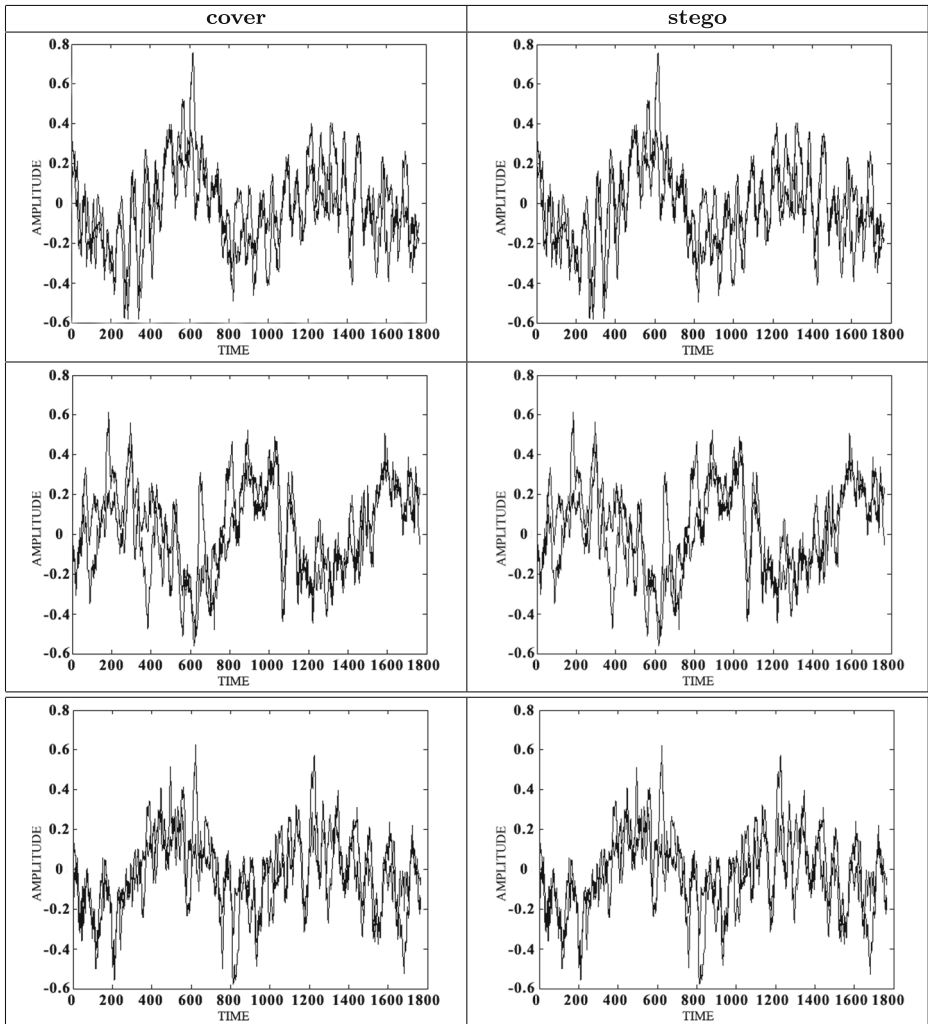


Fig. 5 The waveforms of sample cover audio frame (left) and stego audio frame (right) from each of Artificial video

Although the capacity of method [13] is better than our method yet our method yields a comparatively very high PSNR value claiming higher embedding capacity.

4.6 Color frequency test

Chi-square test is used to show whether the distribution of color frequency in an image matches with the distribution which demonstrates distortion from hidden data that is embedded. We determine the probability of embedding by using (33).

$$p = \int_0^{\chi^2} \frac{t^{\frac{v}{2}-1} e^{-t/2}}{2^{\frac{v}{2}} \Gamma(\frac{v}{2})} dt. \tag{33}$$

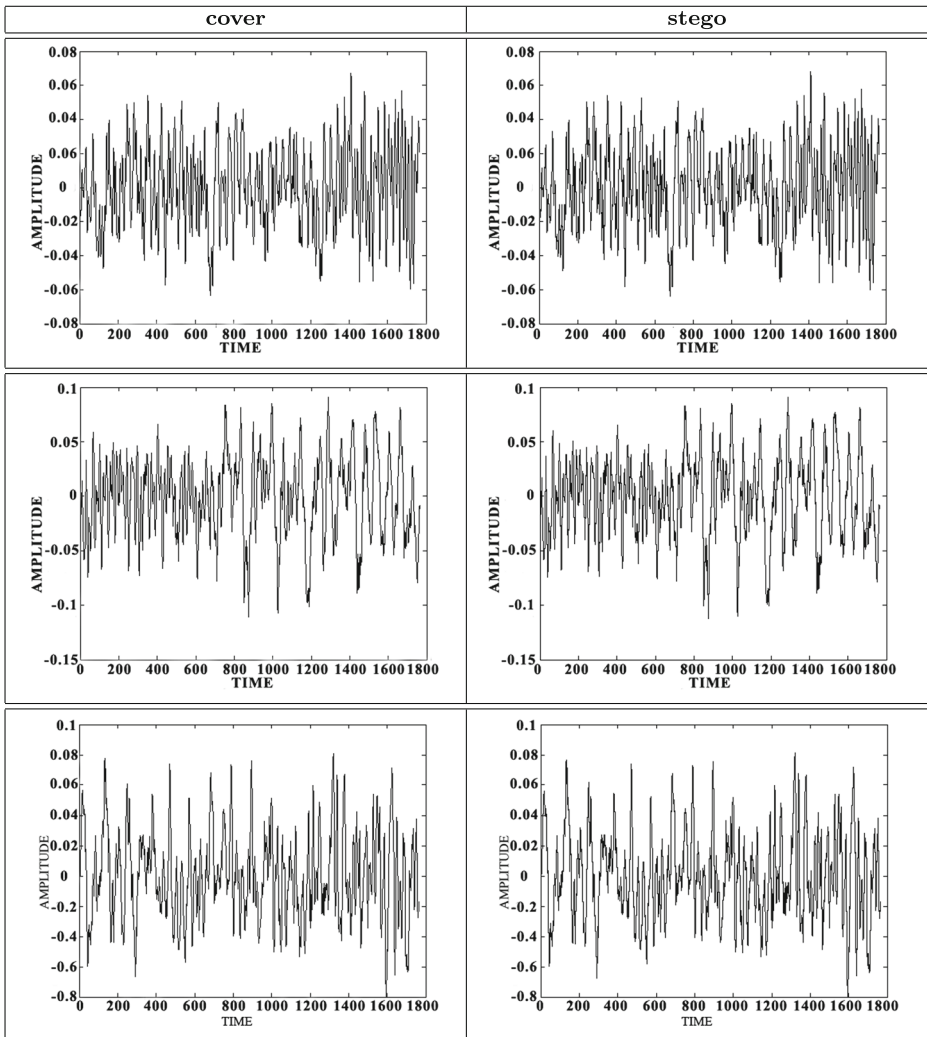


Fig. 6 The waveforms of sample cover audio frame (left) and stego audio frame (right) from each of Cartoon video

where, v is the degrees of freedom in Chi-square test and $v + 1$ is the number of distinct color categories. This probability is calculated by considering pixel values all over the image component of the video. Places of video where no secret message is embedded, p , obtained from

Table 10 MSE and PSNR for images of different sequence

Category	Avg. MSE	Avg. PSNR (dB)
Cartoon	10.77	37.83
Wildlife	14.20	36.60
Architectures	8.045	39.07
Standard	10.10	38.35

Table 11 MSE and PSNR for standard images

Category	Avg. MSE	Avg. PSNR (dB)
lena	6.95	39.70
baboon	17.94	35.59
fruits	8.20	38.99

(33), should be almost equal to 0. However in the proposed technique data is not directly embedded into pixels but adjustment is made to the difference in values of two neighboring pixels. Thus we can safely conclude that this method is incapable of analyzing hidden messages in our novel method. Moreover the testing of our algorithm in different categories of videos have yielded the value $p = 0.0191$ which further proves the effectiveness of this method.

4.7 Sample pair method

Dumitrescu et al. [5] show that this scheme is based on the assumptions given below: *Assumption:* Let, P be the set of all pixel intensity pairs (r, s) such that either $r - s = 2u + 1$ where $0 \leq u \leq 2^7 - 1$ and the even component is lesser than the odd component of the pair. Let, Q be the set of all pixel intensity pairs (r, s) such that either $r - s = 2u + 1$ where $0 \leq u \leq 2^7 - 1$ and the even component is greater than the odd component of the pair. The assumption is that statistically we have $|P| = |Q|$.

In Table 13, we find the AEC of 3 types of images cartoon, Wildlife, Architectural are 2.21 bpp, 2.22 bpp, 2.25bpp. But in Table 16, we find that the sample pair analysis can detect only 0.98 bpp, 0.89 bpp, 1 bpp for the aforesaid image types, i.e., almost 60% message bits remains undetectable by sample pair analysis. We conclude that our proposed algorithm withstands this test.

4.8 ROC analysis

Among various performance measurement techniques against steganalysis tools, measuring the false positive rate and false negative rate in order to draw the receiver operating characteristic (ROC) curve is very popular. As shown in Table 17, the confusion matrix is determined in terms of True positive, True Negative, False Negative and False Positive as follows:

- TP : Stego that is correctly classified as Stego
- TN : Cover that is correctly classified as Cover
- FP : Cover that is wrongly classified as Stego
- FN : Stego that is wrongly classified as Cover

Table 12 SNR for Audio of different sequences

Category	Avg. SNR (dB)
Cartoon	60.91
Wildlife	61.75
Architectural	54.61

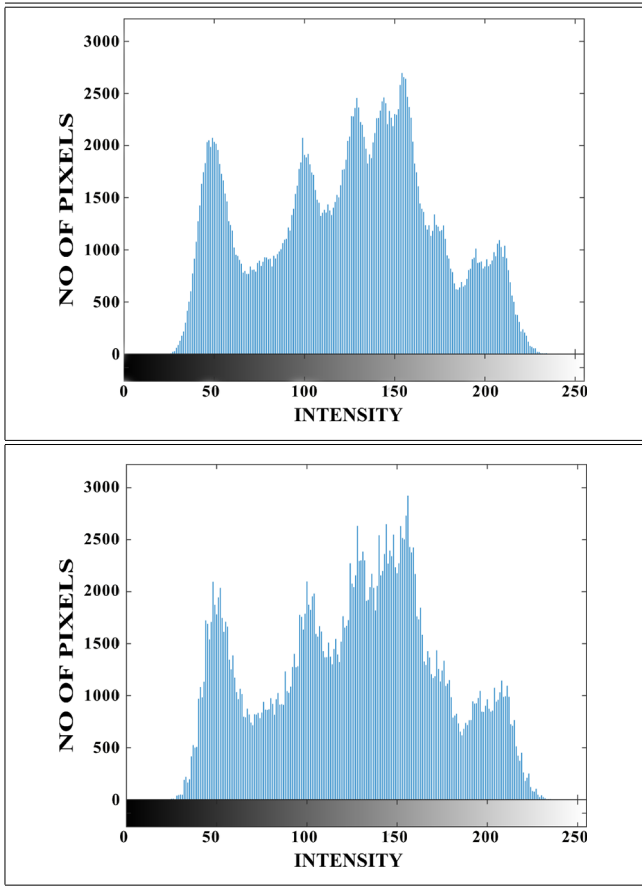


Fig. 7 Histogram of cover (at top) and stego (at bottom)










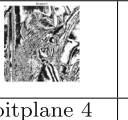
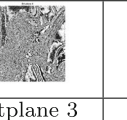
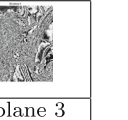



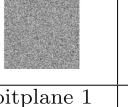
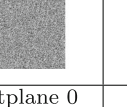
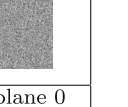
cover	stego	cover	stego	cover	stego
					
All bitplanes	All bitplanes	bitplane 7	bitplane 7	bitplane 6	bitplane 6
					
bitplane 5	bitplane 5	bitplane 4	bitplane 4	bitplane 3	bitplane 3
					
bitplane 2	bitplane 2	bitplane 1	bitplane 1	bitplane 0	bitplane 0

Fig. 8 Bitplane analysis of cover (lena.bmp) and stego (lenaStego.bmp)

Table 13 Image for different sequences

Category	AEC (bpp)
Cartoon	2.21
Wildlife	2.22
Architectures	2.25

Table 14 Capacity for audio

Category	Avg. embedding capacity (bps)
Cartoon	6.47
Wildlife	6.14
Architectural	6.71

Table 15 Comparison of capacity and PSNR with other methods

Methodology	Avg. embedding capacity (bpp)	Avg. PSNR (dB)
Wu D.C et al. [44]	1.57	38.65
Wu H.C et al. [45]	2.91	35.75
Hashim et al. [10](using LSB)	1.00	51.96
Hashim et al. [10](using HWT)	< 1	51.18
Yang et al. [46]	0.67	52.91
Wang et al. [42]	1.99	45.16
Kim et al. [19]	1.95	46.88
Sajjad et al. [36]	1	47.93
Soumendu et al. [4]	<= 1	30.0
Proposed Method	2.20	38.55

Table 16 Sample pair results for image sequences

Category	Avg. BPP (sample pair)	Avg. BPP (actual)
Cartoon	0.98	2.21
Wildlife	0.89	2.22
Architectural	1.00	2.25

Table 17 Confusion matrix

		True Class	
		<i>p</i>	<i>n</i>
Hypothesized Class	<i>p</i>	true positives (<i>TP</i>)	false positives (<i>FP</i>)
	<i>n</i>	false negatives (<i>FN</i>)	true negatives (<i>TN</i>)
Column total		<i>P</i>	<i>N</i>

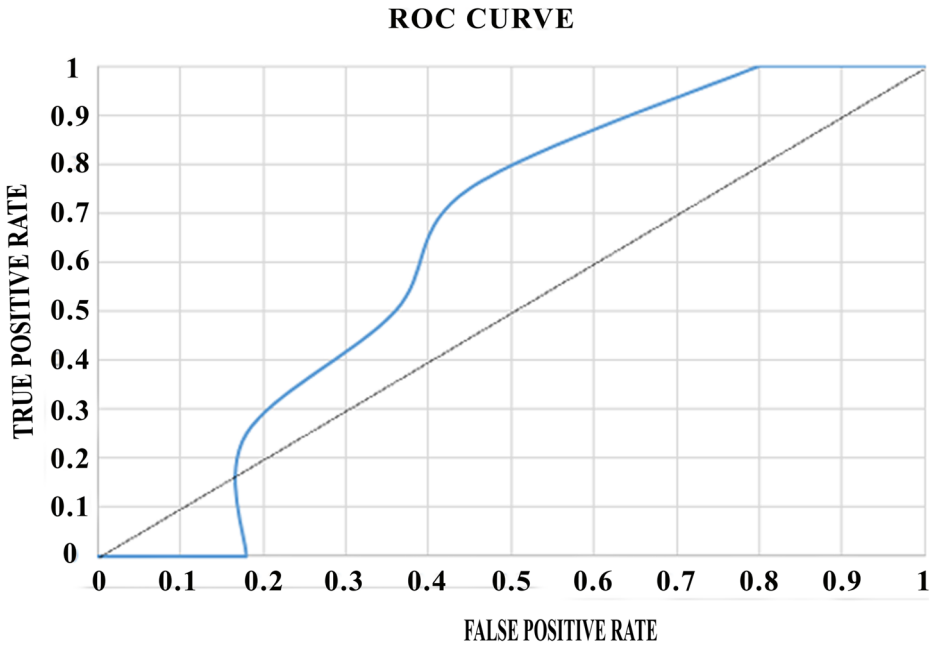


Fig. 9 ROC curve for proposed algorithm

Based on the TP , TN , FP and FN values, True Positive Rate (TPR) and False Positive Rate (FPR) are determined as shown in (34) and (35)

$$TPR = \frac{\#TP}{\#TP + \#FN} \tag{34}$$

$$FPR = \frac{\#FP}{\#TN + \#FP} \tag{35}$$

Figure 9 shows the Receiver Operating Curve (ROC) for proposed algorithm using the TPR and FPR values obtained from (34) for different thresholds using StegExpose tool [2] that combines RS analysis, Sample Pair, etc. In Appendix A we show a table of sample images containing filename, classification (stego or cover), quantitative steganalysis (payload size in bytes), Primary Sets, χ^2 , Sample Pairs, RS analysis, Fusion (mean) in CSV format. In our experimental setup, we test 300 set of images and find the area under the curve (AUC) of the ROC i.e., 0.5503 which is low that indicates the strength of our proposed method.

In our experimental setup, we perform Bit Error Rate (BER), Normalized Cross-Correlation (NCC), Universal Image Quality Index (UIQI) test Structural Similarity Index Metric (SSIM) test as describe below.

Table 18 Avg. BER for image and audio sequences

Category	Avg. BER (for image)	Avg. BER (for audio)
Cartoon	0.157	0.19
Wildlife	0.21	0.19
Architectural	0.16	0.22

4.9 Bit error rate analysis

The value of Bit Error Rate (BER) (shown in Tables 18 and 19) represents the ratio of total number of distorted bits to the total number of bits in the image or audio frame [37].

4.10 Normalized cross-correlation (NCC)

The Normalized Cross-Correlation (NCC) is defined by the amount of deflection in the stego image or audio with respect to its cover version [37]. Always the value of *NCC* is 1 for a pair of same images (or audio frame). The value of *NCC* is obtained from the (36)

$$NCC = \frac{\sum_{i=1}^P \sum_{j=1}^Q [\omega(i, j) \cdot \omega^*(i, j)]}{\sum_{i=1}^P \sum_{j=1}^Q \omega(i, j)^2} \tag{36}$$

where $\omega(i, j)$ and $\omega^*(i, j)$ represent the pixel intensity values of the cover and stego image (or audio) respectively. The average *NCC* value of our method is shown in Tables 20 and 21, which is very close to 1 denoting marginal distortion caused by embedding.

4.11 Universal image quality index (UIQI)

Universal Image Quality Index is an index which calculates any kind of distortion as a combination of three factors: loss of correlation (Q_1), luminance distortion (Q_2) and contrast distortion (Q_3) [11].

Let $x = \{x_i | i \in \mathbb{N}^+\}$ and $y = \{y_i | i \in \mathbb{N}^+\}$ be the pixels of the original and the stego images respectively. N is number of pixels in the images. Here Q_1 , Q_2 and Q_3 are given in following equation:

$$Q_1 = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \tag{37}$$

$$Q_2 = \frac{2\bar{x}\bar{y}}{(x^2 + y^2)} \tag{38}$$

$$Q_3 = \frac{2\sigma_x \sigma_y}{(\sigma_x^2 + \sigma_y^2)} \tag{39}$$

where,

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$$

$$\bar{y} = \frac{1}{N} \sum_{i=1}^N y_i$$

$$\sigma_x^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2$$

$$\sigma_y^2 = \frac{1}{N-1} \sum_{i=1}^N (y_i - \bar{y})^2$$

Table 19 BER for standard images

Category	BER
lena	0.17
baboon	0.22
fruits	0.18

Table 20 Avg. NCC for image and audio sequences

Category	Avg. NCC (for image)	Avg. NCC (for audio)
Cartoon	1.0009	0.9999
Wildlife	1.0010	1.0000
Architectural	0.9998	0.9999

Table 21 Avg. NCC for standard images

Category	NCC
lena	1.0010
baboon	1.0013
fruits	1.0005

Table 22 Avg. UIQI for image and audio sequences

Category	Avg. UIQI
Cartoon	0.9987
Wildlife	0.9962
Architectural	0.9989
Standard	0.9978

Table 23 Avg. UIQI for standard images

Category	UIQI
lena	0.9981
baboon	0.9969
fruits	0.9985

Table 24 Avg. SSIM for image and audio sequences

Category	Avg. SSIM (for image)	Avg. SSIM (for audio)
Cartoon	0.9987	0.9994
Wildlife	0.9963	0.9996
Architectural	0.9989	0.9999

Table 25 Avg. SSIM for standard images

Category	SSIM
lena	0.9981
baboon	0.9969
fruits	0.9985

$$\sigma_{xy} = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})$$

Now, UIQI (say, Q) is determined by (40)

$$Q = \frac{Q_1 * Q_2 * Q_3}{4\sigma_{xy}\bar{x}\bar{y}} = \frac{4\sigma_{xy}\bar{x}\bar{y}}{(\sigma_x^2 + \sigma_y^2)[(\bar{x})^2 + (\bar{y})^2]} \tag{40}$$

\bar{x} is the average intensity of original image. \bar{y} is the same for corresponding stego image. σ_x and σ_y denote the standard deviation of the same. σ_{xy} is the covariance. In our case UIQI value is given in following table Tables 22 and 23. The values are very close to 1 which signifies the distortion is minimal.

4.12 Structural similarity index metric (SSIM)

Structural Similarity Index Metric (SSIM) can be defined as follows:

Definition 3 Structural Similarity Index Metric (SSIM) [11] is a image degradation measurement estimated using (41).

$$SSIM \triangleq \frac{(2 \times \bar{x} \times \bar{y} + A_1)(2 \times \sigma_{xy} + A_2)}{(\sigma_x^2 + \sigma_y^2 + A_2) \times (\bar{x}^2 + \bar{y}^2 + A_1)} \tag{41}$$

where $A_1 = (m_1 \cdot F)^2$ and $A_2 = (m_2 \cdot F)^2$ are two constants. F is $2^{\text{number of bits per pixel}} - 1$. $m_1 = 0.01$ and $m_2 = 0.03$ by default.

The value of SSIM is between -1 and 1 . For identical image value is 1 . In our case SSIM value is given in following Tables 24 and 25. The values are very close to 1 which signifies the distortion is minimal.

4.13 StirMark analysis

The StirMark 4.0 [33] tool is considered as a benchmark tool to justify the strength of any steganographic algorithm requires. It occurs a small geometrical alteration that causes a loss of synchronization between the analyzed images. The geometrical distortion may be “sheared”, “stretches”, “bent”, “shifted” and “rotated” by a small amount. The proposed

Table 26 StirMark analysis of proposed technique on cover and stego version of image wildlife (size: 255×141)

	Factor	Cover	Stego
Self Similarities	1	37.0341	35.6238
Self Similarities	2	53.2843	51.8944
Self Similarities	3	36.5027	35.4051
PSNR	10	38.5543	38.5539
AddNoise	20	8.92531	8.91026
AddNoise	40	7.66654	7.65682
AddNoise	100	7.00953	6.99652
SmallRandom Distortions	0.95	18.143	10.8504
SmallRandom Distortions	1.00	17.9191	10.7232
SmallRandom Distortions	1.05	17.71	10.6064
ConvFilter	1.00	9.18838	9.194
ConvFilter	2.00	-8.74974	-8.42135
MedianCut	3.00	35.0096	34.1653

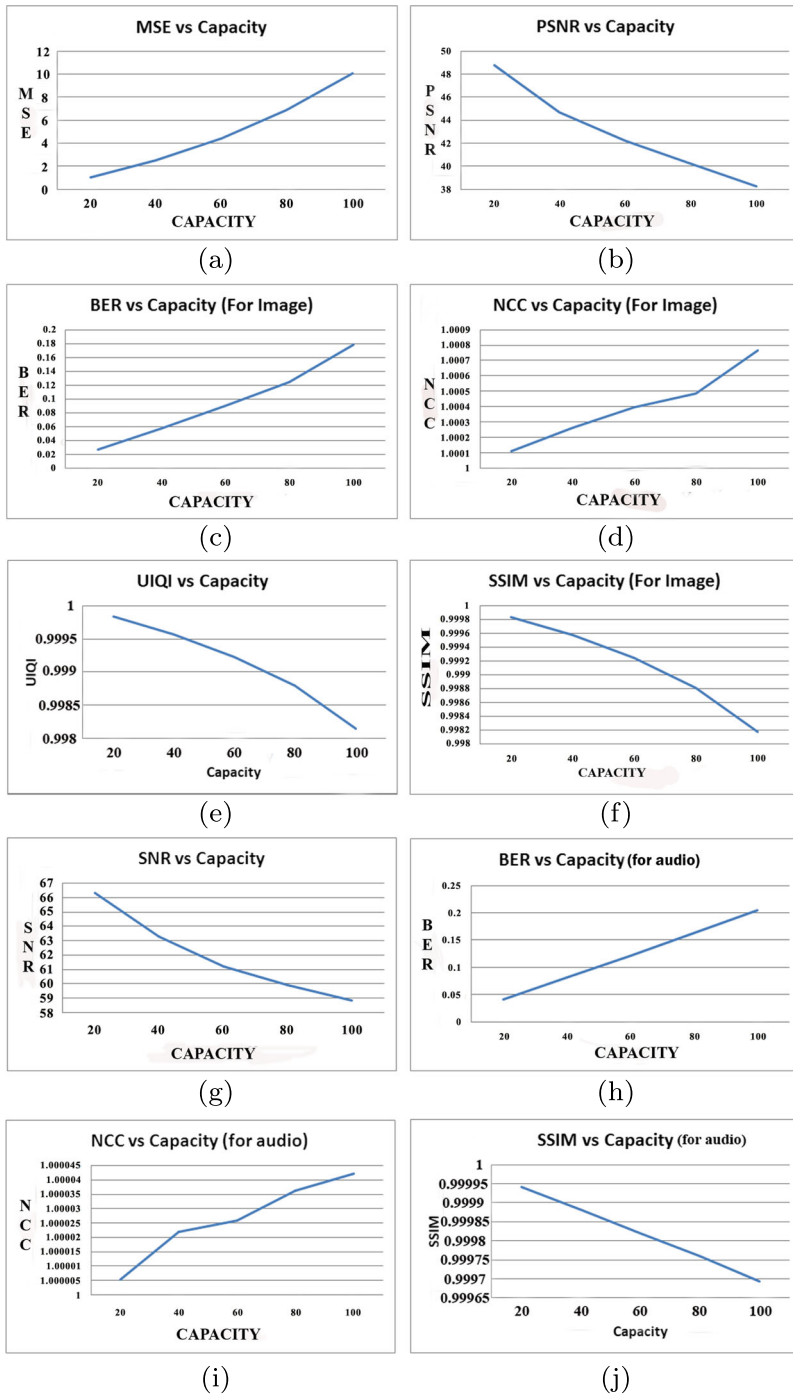


Fig. 10 Distortion parameter vs. embedding capacity

Table 27 Abbreviations used

<i>GF</i> :	Galois Field	<i>PVD</i> :	Pixel value differencing
<i>SVD</i> :	Sample value differencing	<i>PSNR</i> :	Peak-signal-to-noise ratio
<i>SNR</i> :	Signal-to-noise ratio	<i>AEC</i> :	Average Embedding Capacity
<i>BPP</i> :	Bits per pixel	<i>BPS</i> :	Bits per sample
<i>ROC</i> :	Receiver operating characteristic	<i>AUC</i> :	Area under the curve
<i>TP</i> :	True positives	<i>TN</i> :	True negative
<i>FP</i> :	False positives	<i>FN</i> :	False negative
<i>TPR</i> :	True positives rate	<i>TPR</i> :	True positive rate
<i>BER</i> :	Bit Error Rate	<i>SSIM</i> :	Structural Similarity Index Metric
<i>NCC</i> :	Normalized Cross-Correlation	<i>UIQI</i> :	Universal Image Quality Index

algorithm exhibits satisfactory outcomes for the image. The very minute differences between the values of the cover and its stego version, as shown in Table 26, is a clear indication of the fact that our technique withstand the benchmark.

4.14 Payload curve

Figure 10 shows the different payload curves of distortion parameter Vs. embedding capacity. Here distortion parameter includes MSE, PSNR, BER, SNR, NCC, UIQI, SSIM etc. We vary the capacity from 20% to 100% in order to analyzed the effect on different quality metrics and distortion parameter as well. Figure 10a denotes MSE vs. capacity where MSE increases with capacity. Figure 10b denotes PSNR vs. capacity where PSNR decreases with capacity. Figure 10c denotes BER vs. capacity (for image) where BER increases with capacity. Figure 10.d denote NCC vs. capacity (for image) where NCC increases with capacity. Figure 10e and f denote UIQI vs. capacity (for image) and SSIM vs. capacity (for image) respectively where UIQI and SSIM both decreases with capacity. Figure 10g denotes SNR vs. capacity where SNR decreases with capacity. Figure 10h denotes BER vs. capacity (for audio) where BER increases with capacity. Figure 10i denote NCC vs. capacity (for audio) where NCC increases with capacity. Figure 10j denote SSIM vs. capacity (for audio) where SSIM decreases with capacity.

The abbreviations, used in this paper, are provided in Table 27.

5 Conclusion

In this paper, we have proposed a novel multibit steganographic technique that involves two phases. In the first phase, we encode the information to be embedded using Galois field ($GF(2^8)$) arithmetic. In the second phase, we embed the encoded information in the cover multimedia in spatial domain. For images (an uncompressed video can be considered as a set of images as well), our algorithm is capable of hiding maximum of 6 bits per pixel and for audio, it can hide a maximum of 13 bits per sample without keeping any perceptible signature in the stego media. We have presented both theoretical arguments and experimental results to establish high embedding capacity and security provided by our method. In this work, we have worked with lossless image and audio only. Applicability of our method in lossy domain remains an interesting open problem and we plan to take this up as part of our future work.

Acknowledgements The authors would like to thank Mr. Uttiya Ghosh for his partial help in implementation of this project when he was the student of St. Thomas' College of Engineering & Technology, Kolkata.

Appendix A: StegExpose outputs on sample images

File name	Above stego threshold?	Secret message size	Primary Sets	Chi Square	Sample Pairs	RS analysis	Fusion (mean)
Cbaaboon.bmp	FALSE	2317	0.053377671	0.001257046	0.031670587	0.019738014	0.026510829
Ccameraman.bmp	FALSE	2790	0.05251683	0.004382724	0.037678798	0.033091533	0.031917471
chouse.bmp	TRUE	5395	0.016316266	0.216400282	0.010244265	0.003008525	0.061492334
Cjet.bmp	FALSE	420	0.004478719	0.006636901	0.004517818	0.003575426	0.004802216
Clena.bmp	FALSE	2089	0.040869985	0.001299321	0.037579107	0.015864729	0.023903286
clivingroom.bmp	FALSE	4278	7.91E-04	0.189903464	0.001774558	0.002548165	0.048754337
cpeppers.bmp	TRUE	21974	0.001250652	0.99999995	5.38E-04	0	0.25044714
cpirate.bmp	TRUE	12865	0.001023816	0.584048071	0.001088174	3.41E-04	0.146625333
cwalkbridge.bmp	FALSE	1369	0.024905956	9.87E-04	0.021398234	0.015133158	0.01560599
cwomanblonde.bmp	TRUE	22087	0.005740325	0.999999933	0.001164532	0	0.251726197
Sbaaboon.bmp	TRUE	6334	0.072145326	0.001222529	0.097431552	0.11909553	0.072473734
Scameraman.bmp	FALSE	4179	0.074550204	0.002528217	0.057094954	0.057073417	0.047811698
Sjet.bmp	TRUE	5374	0.069027774	0.004615046	0.095479557	0.076813388	0.061483941
Slena.bmp	FALSE	3124	0.063350777	0.001311896	0.046775722	0.031541216	0.035744903

References

1. Amin R, Kumar N, Biswas GP, Iqbal R, Chang V (2017) A light weight authentication protocol for IoT-enabled devices in distributed cloud computing environment. *Futur Gener Comput Syst* 78, Part 3:1005–1019. <https://doi.org/10.1016/j.future.2016.12.028>. ISSN. 0167-739X
2. Boehmm B (2014) Stegexpose a tool for detecting LSB steganography. <https://github.com/b3dk7/StegExpose>
3. Burrows M, Abadi M, Needham R (1990) A logic of authentication. DEC SRC Research Report 39
4. Chakraborty S, Singh JA, Bhatnagar C (2017) LSB Based non blind predictive edge adaptive image steganography. *Multimedia Tools Appl* 76:7973–7987. <https://doi.org/10.1007/s11042-016-3449-4>. ISSN. 1573-7721
5. Dumitrescu S, Wu X, Memon N (2002) On steganalysis of random LSB embedding in continuous-tone images. In: *International conference on IEEE image processing*, vol 3, pp 641–644. ISBN: 0-7803-7622-6. <https://doi.org/10.1109/ICIP.2002.1039052>
6. Farahani B, Firouzi F, Chang V, Badaroglu M, Constant N, Mankodiya K (2017) Towards fog-driven IoT ehealth: promises and challenges of IoT in medicine and healthcare. *Futur Gener Comput Syst* 78 Part 2:659–676. <https://doi.org/10.1016/j.future.2017.04.036>
7. Feng B, Lu W, Sun W (2015) Secure binary image steganography based on minimizing the distortion on the texture. *IEEE Trans Inf Forensics Secur* 10(2):243–255. <https://doi.org/10.1109/TIFS.2014.2368364>
8. Fridrich J, Lisonek P (2007) Grid colorings in steganography. *IEEE Trans Inf Theory* 53(4):1547–1549. <https://doi.org/10.1109/TIT.2007.892768>
9. Fridrich J, Lisonek P, Soukal D (2007) On steganographic embedding efficiency. In: Camenisch JL, Collberg CS, Johnson NF, Sallee P (eds) *IH 2006*. LNCS, vol 4437. Springer, Heidelberg, pp 282–296. ISBN. 978-3-540-74124-4, https://doi.org/10.1007/978-3-540-74124-4_19
10. Hashim AT, Ali YH, Ghazoul SS (2011) Developed method of information hiding in video AVI file based on hybrid encryption and steganography. *Eng Tech J* 29(2):359–373. ISSN. 16816900 24120758
11. Hemlatha S, Dinesh A, Renuka A (2015) Wavelet transform based steganography technique to hide audio signals in image. In: *Elsevier B.V. graph algorithms, high performance implementations and its applications*, vol 47, pp 272–281. <https://doi.org/10.1016/j.procs.2015.03.207>
12. Hrytskiv Z, Voloshynovskiy S, Rytsar Y (1998) Cryptography and steganography of video information in modern communications. In: *Third international conference on telecommunications in modern satellite, cable and broadcasting services TELSIKS'97*, vol 11, No. 1, pp 164–167
13. Hu SD, Kin Tak U (2011) A novel video steganography based on non-uniform rectangular partition. In: *IEEE international conference on computational science and engineering*, pp 57–61. ISBN. 978-1-4577-0974-6. <https://doi.org/10.1109/CSE.2011.24>
14. Huang X, Kawashima R, Segawa N, Abe Y (2008) Design and implementation of synchronized audio to audio steganography scheme. In: *International conference on intelligent information hiding and multimedia signal processing (IHMSP)*. IEEE Xplore, pp 331–334. ISBN: 978-0-7695-3278-3, <https://doi.org/10.1109/IIH-MSP.2008.98>
15. Huayong G, Huang M, Wang Q (2011) Steganography and steganalysis based on digital image. In: *4th international congress on image and signal processing (CISP)*, vol 1. IEEE, pp 252–255. <https://doi.org/10.1109/CISP.2011.6099953>
16. Juneja M, Sandhu PS (2009) Designing of robust image steganography technique based on LSB insertion and encryption. In: *Advances in recent technologies in communication and computing, ARTCom '09*. IEEE, pp 302–305. <https://doi.org/10.1109/ARTCom.2009.228>
17. Kak A (2015) *Lecture notes on computer and network security*. Purdue University
18. Kim C (2014) Data hiding by an improved exploiting modification direction. *Multimedia Tools and Applications* 69(3):569–584. <https://doi.org/10.1007/s11042-012-1114-0>
19. Kim C, Shin DK, Shin DI, Zhang X (2011) Improved steganographic embedding exploiting modification direction in multimedia communications. *Commun Comput Inf Sci* 186:130–138. https://doi.org/10.1007/978-3-642-22339-6_16
20. Kumar H, Anuradha (2012) Enhanced LSB technique for audio steganography. In: *Computing communication networking technologies (ICCCNT) third international conference*. IEEE, pp 1–4. <https://doi.org/10.1109/ICCCNT.2012.6395978>
21. Liao D, Sun G, Li H, Yu H, Chang V (2017) The framework and algorithm for preserving user trajectory while using location-based services in IoT-cloud systems. *Clust Comput*, Springer 20(3):2283–2297. <https://doi.org/10.1007/s10586-017-0986-1>
22. Lin GS, Chang YT, Lie WN (2010) A framework of enhancing image steganography with picture quality optimization and anti-steganalysis based on simulated annealing algorithm. *IEEE Trans Multimedia* 12(5):345–357. <https://doi.org/10.1109/TMM.2010.2051243>

23. Lou D, Hub C (2012) LSB Steganographic method based on reversible histogram transformation function for resisting statistical steganalysis. In: Information sciences. Elsevier, pp 346–358. <https://doi.org/10.1016/j.ins.2011.06.003>
24. Luo W, Huang F, Huang J (2010) Edge adaptive image steganography based on LSB matching revisited. *IEEE Trans Inf Forensics Secur* 5(2):201–214. <https://doi.org/10.1109/TIFS.2010.2041812>
25. Maya ST, Merico N, Meana HP (2006) An image steganography systems based on BPCS and IWT. In: 16th IEEE international conference on electronics, communications and computers (CONIELECOMP), p 51. <https://doi.org/10.1109/CONIELECOMP.2006.14>
26. Miller E, Sturmfels B (2006) Combinatorial commutative algebra. Springer, New York
27. Mstafa RJ, Elleithy KM (2014) A highly secure video steganography using hamming code (7, 4), systems. In: Applications and technology conference (LISAT), IEEE Long Island, pp 1–6. <https://doi.org/10.1109/LISAT.2014.6845191>
28. Mukherjee I, Paul G (2013) Efficient multi-bit image steganography in spatial domain. In: Information systems security, chapter 21, lecture notes, vol 8303. Springer, pp 270–284. https://doi.org/10.1007/978-3-642-45204-8_21
29. Mukherjee N, Bhattacharya A, Bose S (2013) Evolutionary multibit grouping steganographic algorithm. In: Information systems security, LNCS, vol 8303. Springer, pp 285–296. https://doi.org/10.1007/978-3-642-45204-8_22
30. Paul G, Davidson I, Mukherjee I, Ravi S (2012) Keyless steganography in spatial domain using energetic pixels. In: Venkatakrishnan V, Goswami D (eds) ICISS 2012, LNCS, vol 7671. Springer, pp 134–148. https://doi.org/10.1007/978-3-642-35130-3_10
31. Paul G, Davidson I, Mukherjee I, Ravi S (2016) Keyless dynamic optimal multi-bit image steganography using energetic pixels. *Multimedia Tools and Applications* 75:1–27. ISBN: 978-3-642-35129-7
32. Petitcolas FAP (2000) Watermarking schemes evaluation. *IEEE Signal Processing* 17(5):58–64. <https://doi.org/10.1109/79.879339>
33. Petitcolas FAP, Anderson RJ, Kuhn MG (1998) Attacks on copyright marking systems. In: Second international workshop on information hiding, IH98, Portland, Oregon, U.S.A., Proceedings, LNCS 1525. Springer, pp 219–239. https://doi.org/10.1007/3-540-49380-8_16
34. Provos N (2001) Defending against statistical steganalysis. In: Tenth USENIX security symposium, pp 325–335
35. Rudolf L, Harald N *Encyclopedia of mathematics and its applications: finite fields*, vol 20, part 1. Cambridge University Press, ISBN-13: 978-0521065672
36. Sajjad M, Khan M, Irfan M, Irfan R, Wook BS (2016) A novel magic LSB substitution method (M-LSB-SM) using multi-level encryption and achromatic component of an image. *Multimedia Tools and Applications* 75:14867–14893. <https://doi.org/10.1007/s11042-015-2671-9>
37. Singh S, Siddiqui TJ (2012) A security enhanced robust steganography algorithm for data hiding. *Int J Comput Sci Issues* 9(3, no. 1):131–139. ISSN (Online). 1694-0814
38. Socek D, Kalva H, Magliveras SS, Marques O, Culibrk D, Furht B (2007) New approaches to encryption and steganography for digital videos. *Multimedia Systems* 13(3):191–204. <https://doi.org/10.1007/s00530-007-0083-z>
39. Song S, Zhangb J, Liaoa X, Dua J, Wena Q (2011) A novel secure communication protocol combining steganography and cryptography. In: *Advanced in control engineering and information science*. Elsevier, pp 2767–2772. <https://doi.org/10.1016/j.proeng.2011.08.521>
40. Sreekantha DK, Kavya AM (2017) Agricultural crop monitoring using IOT - a study. In: 11th international conference on intelligent systems and control (ISCO), 2017, pp 134–139. <https://doi.org/10.1109/ISCO.2017.7855968>
41. Sun G, Chang V, Ramachandran M, Sun Z, Li G, Yu H, Liao D (2017) Efficient location privacy algorithm for Internet of Things (IoT) services and applications. *J Netw Comput Appl* 89:3–13. <https://doi.org/10.1016/j.jnca.2016.10.011>
42. Wang ZH, Kieu TD, Chang CC, Li MC (2010) A novel information concealing method based on exploiting modification direction. *J Inf Hiding Multimedia Signal Process* 1(1):1–9. https://doi.org/10.1007/978-3-642-35473-1_14
43. Wang K, Zhao H, Wang H (2014) Video steganalysis against motion vector-based steganography by adding or subtracting one motion vector value. *IEEE Trans Inf Forensics Secur* 9(5). <https://doi.org/10.1109/TIFS.2014.2308633>
44. Wu DC, Tsai WH (2003) A steganographic method for images by pixel-value differencing. *Pattern Recogn Lett* 24:1613–1626. [https://doi.org/10.1016/S0167-8655\(02\)00402-6](https://doi.org/10.1016/S0167-8655(02)00402-6)
45. Wu HC, Wu NI, Tsai CS, Hwang MS (2005) Image steganographic scheme based on pixel-value differencing and LSB replacement methods. *IEEE Proceedings, Vision, Image and Signal Processing* 152(5):611–615. <https://doi.org/10.1049/ip-vis:20059022>

46. Yang CN, Ouyang JF, Harn L (2012) Steganography and authentication in image sharing without parity bits. *Opt Commun* 285:1725–1735. <https://doi.org/10.1016/j.optcom.2011.12.003>
47. Zhou J Introduction to symmetric polynomials and symmetric functions, <http://faculty.math.tsinghua.edu.cn/jzhou/symmetricf.pdf>



Nabanita Mukherjee (Ganguly) passed B.Tech in Information Technology from West Bengal University of Technology in 2006. Received M.Tech in Information Technology from Jadavpur University in 2009. Presently doing Ph.D. in Computer Science and Engineering Department of Jadavpur University. My research area is High Capacity Steganography Algorithm for Multimedia.



Goutam Paul Goutam Paul did his Ph.D. in 2009 from Indian Statistical Institute, Kolkata (degree awarded from Jadavpur University, Kolkata, India). From 2006 to 2013, he was an Assistant Professor in the Department of Computer Science and Engineering of Jadavpur University and during 2012–2013, he visited RWTH Aachen, Germany as a Humboldt Fellow. From August 2013, Goutam Paul has been serving the R. C. Bose Centre for Cryptology and Security of Indian Statistical Institute, Kolkata, as an Assistant Professor. He also received the Young Scientist Award from the National Academy of Sciences, India (NASI) in 2013. His research interest includes stream cipher cryptanalysis, efficient hardware design of cryptographic primitives, analysis of BB84-like quantum key distribution protocols, steganography and data hiding.



Sanjoy Kumar Saha received B.E. and M.E. degrees in Electronics and Tele-communication engineering from Jadavpur University in 1990 and 1992 respectively. Obtained the Ph.D. degree from Bengal Engineering and Science University in 2006. Currently working as Professor in Computer Science and Engineering Department of Jadavpur University. Research interests are in the area of image, video and audio processing, multimedia data retrieval and sensor signal processing.