Diminish additive arc in Steganography Using Clause-Outline Policy

[1]SK. Karimulla, [2]G. Sasibhusana Rao

[1,2]Dept. of CSE, B.V.C College of Engineering and Technology, Odalarevu, AP, India

Abstract

Most realistic steganographic algorithms for experimental covers embed messages by minimizing a sum of per-pixel distortions. Current near-optimal policies for this minimization problem are limited to a binary embedding operation. In this paper, we extend this work to embedding operations of larger cardinality. The need for embedding changes of larger amplitude and the merit of this construction are confirmed experimentally by implementing an adaptive embedding algorithm for digital images and comparing its security to other schemes. This paper proposes a complete practical method for minimizing additive distortion in steganography with general embedding operation. Let every possible value of every stage element be assigned a scalar expressing the distortion of an embedding change done by replacing the cover element by this value. The total distortion is assumed to be a sum of per-element distortions. Both the payload-limited sender (minimizing the total distortion while embedding a fixed payload) and the distortion-limited sender (maximizing the payload while introducing a fixed total distortion) are considered. Without any loss of performance, the nonbinary case is decomposed into several binary cases by replacing individual bits in cover elements. The binary case is approached using a novel Clause-coding scheme based on dual convolution policy's equipped with the Viterbi algorithm. Most current coding schemes used in steganography (matrix embedding, wet paper policy's, etc.) and many new ones can be implemented using this outline. We report extensive experimental results for a large set of relative payloads and for different distortion profiles, including the wet paper channel. Practical merit of this approach is validated by constructing and testing adaptive embedding schemes for digital images in raster and transform domains.

Keywords

Outline-Policy Quantization, Convolution Policy's, Coding Loss, Steganography, Embedding Impact, Matrix Embedding, Wet Paper Policies

I. Introduction

This paper provides a general methodology for embedding while minimizing an arbitrary additive distortion function with a performance near the theoretical bound. We present a complete methodology for solving both the payload-limited and the distortion-limited sender. The implementation described in this paper uses standard signal processing tools – convolution policy's with a outline quantize – and adapts them to our problem by working with their dual representation. These policy's, which we call the Clause–Outline Policy's (STCs), can directly improve the security of many existing steganographic schemes, allowing them to communicate larger payloads at the same embedding distortion or to decrease the distortion for a given payload. Additionally, this work allows an iterative design of new embedding algorithms by making successive adjustments to the distortion function to minimize detectability measured using blind steganalyzers on real cover sources

In Steganography, a secret message is embedded in a cover Object $x = (x_1 \dots x_n) \in X = \{I\}n$ by slightly modifying its individual elements to produce the stage object $y = (y_1, \dots\dots, y_n) \in Y = I1 \times I_2 \times \dots\dots\dots \times In, I_i \subset I$, where

Ii is the range of the embedding operation at element i and $x_i \in I_i$. For example, for the Least Significant Bit (LSB) replacement method, $I_i = \{x_i, \dots..x^i\}$, where $\bar{}x_i$ is x_i after flipping its LSB. The embedding operation is binary if |Ii| = 2

or ternary if |Ii| = 3 for all i. For concreteness, we will call x image and xi its ith pixel but other interpretations are certainly Possible. For example, xi may represent an RGB triple in a color image, a DCT coefficient, etc. Steganographic schemes for complex cover sources, such as digital images, are usually constructed to minimize some distortion measure D between x and y [9] that is assumed to be related to statistical detectability of embedding changes. In this paper, we will consider the following distortion function

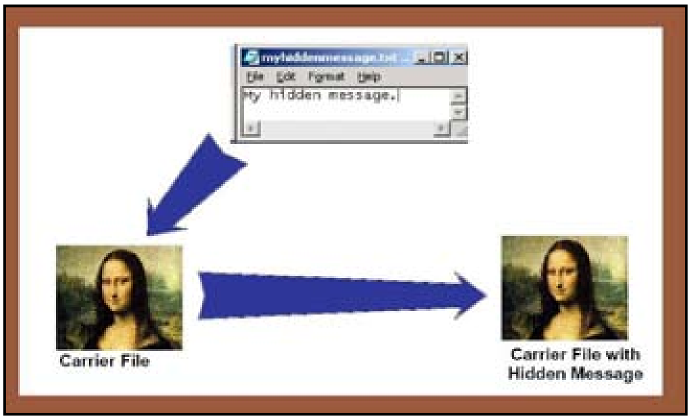


Fig. 1:

The most common choice of ρi for binary embedding operations with scalar costs i, where [S] is the Iverson bracket defined as 1 when the logical statement S is true and 0 otherwise. Note that when i = 1, D is the number of embedding changes. In the MMX algorithm is dependent on the quantization error of the ith DCT coefficient.

The cost functions can reflect higher-order dependencies of cover image pixels (because the dependence on x is not constrained) but the additively of (1) cannot capture dependencies among Embedding changes. Although it is straightforward to extend STCs to nonbinary Alphabets and thus apply them to q-ary embedding operations, their complexity rapidly increases (the number of states in the outline increases from 2h to qh for constraint Height h), limiting thus their performance in practice. The main contribution of this paper is extending the STCs to arbitrary q-ary embedding operations using a simple layered construction without any significant increase in complexity. By moving away from binary embedding operations by increasing the size of Ii, the embedding modifications may become larger and a larger payload can be embedded. We demonstrate experimentally that by restricting the larger amplitude embedding changes adaptively to the content, the multi-layered construction embeds larger payloads with lower statistical detectability, countering thus the established belief that the increase in payload does not outweigh the increase in statistical detectability

This paper provides a coding scheme how to embed with minimal distortion once the steganographer agreed on the distortion function and the embedding operation. The important question of how to choose both of these elements to minimize detectability is not addressed and is left as a future direction. Section II, restates some known relative payload–relative distortion bounds. Section III, reviews the binary embedding operation and Clause-outline policy's [7] for implementation. Even though the relationship between distortion and Steganography security is far from clear, it makes sense to embed messages by minimizing a heuristically chosen distortion function. At least, this is how today's least detectable image Steganography schemes work [14, 20, 16]. The main part, the multi-layered construction, is described and analyzed in Section IV. Application to spatial domain steganography is described in Section V.

Fig. 2:

The paper is concluded in Section VI. All vectors are typed in bold. Random variables and their realizations are denoted using capital and lower case letters, respectively. Furthermore, log(x) = $\log^2(x)$ and ln(x) denotes the natural logarithm. We use h(x) = log x-(1-x) log (1-x) for the binary entropy function. Two mainstream approaches to steganography in empirical covers, such as digital media objects: Steganography designed to preserve a chosen cover model and steganography minimizing a heuristically-defined embedding distortion. The strong argument for the former strategy is that provable undetectability can be achieved w.r.t. a specific model. The disadvantage is that an adversary can usually rather easily identify statistical quantities that go beyond the chosen model that allow reliable detection of embedding changes.

We admit that the relationship between distortion and steganographic security is far from clear, embedding while minimizing a distortion function is an easier problem than embedding with a steganographic constraint (preserving the distribution of covers). It is also more flexible, allowing the results obtained from experiments with blind steganalyzers to drive the design of the distortion function. In fact, today's least detectable steganographic schemes for digital images [2], were designed using this principle. Moreover, when the distortion is defined as a norm between feature vectors extracted from cover and stage objects, minimizing distortion becomes tightly connected with model preservation insofar the features can be considered as a low-dimensional model of covers.

II. Existing System

1. In special domain, the hiding process such as Least Significant Bit (LSB) replacement, is done in special domain, while transform domain methods; hide data in another domain such as wavelet domain.
2. Least Significant Bit (LSB) is the simplest form of Steganography. LSB is based on inserting data in the least significant bit of pixels, which lead to a slight change on the cover image that is not noticeable to human eye. Since this method can be easily cracked, it is more vulnerable to attacks.
3. LSB method has intense affects on the statistical information of image like histogram. Attackers could be aware of a hidden communication by just checking the Histogram of an image. A good solution to eliminate this defect was LSB matching. LSB-Matching was a great step forward in Steganography methods and many others get ideas from it

III. Proposed System

1. In this paper it is planned to introduce a method that embed 2 bits information in a pixel and alter one bit from one bit plane but the message does not necessarily place in the least significant bit of pixel and second less significant bit plane and fourth less significant bit plane can also host the massage.
2. Since in our method for embedding two bits message we alter just one bit plane, fewer pixels would be manipulated during embedding message in an image and it is expected for the steganalysis algorithm to have more difficulty detecting the covert communication. It is clear that in return complexity of the system would increase.
3. In our method there are only three ways that a pixel is allowed to be changed:

- Its least significant Bit would alter (So the gray level of the pixel would increased or decreased by one level)
- The second less significant bit plane would alter (So the gray level of the pixel would increase or decrease by two levels)
- The fourth less significant bit plane would alter (So the gray level of the pixel would increase or decrease by eight levels)

Benefits

- User cannot find the original data.
- It is not easily cracked.
- To increase the Security.
- To increase the size of stored data.
- We can hide more than one bit.

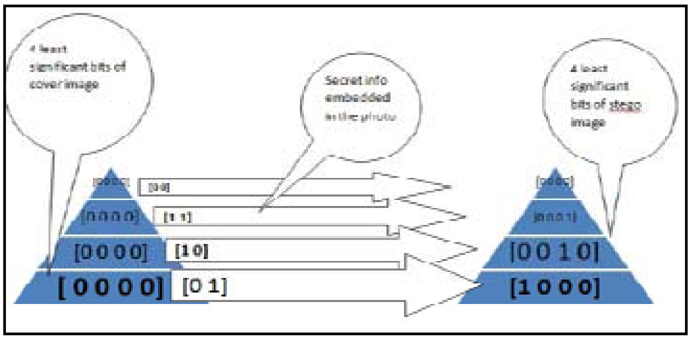

4 least significant bits of cover image
Secret info embedded in the photo
4 least significant bits of stego image
[0000] [00]
[0000] [11]
[0000] [10]
[0000] [01]
[0000]
[0001]
[0010]
[1000]

Fig. 3: How Cover Pixel with Four Less Significant Bits of [0000] Change According to Different Messages

IV. Element

- Key Module
- Watermark embedding
- Authenticator Watermark
- Spread Spectrum
- Watermarked content

A. Element Description

1. Key Module

The Key Module is designed as such a way that the proposed system must be capable of handling any type of data formats, such as if the user wishes to hide any image format then it must be compatible with all usual image formats such as jpg, gif, bmp, it must be also compatible with video formats such as avi, flv, wmf etc.. and also it must be compatible with various document formats, so that the user can be able to user any formats to hide the secret data.

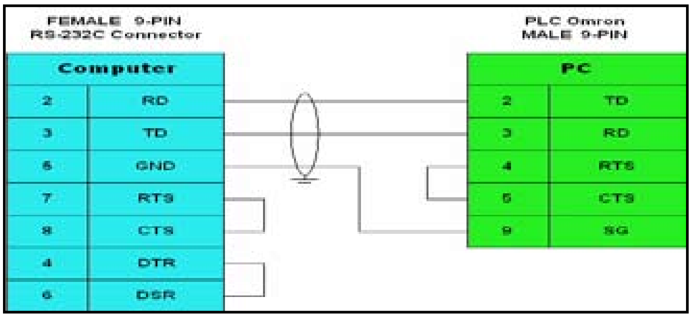


Fig. 4:

2. Watermark Embedding

Watermarking is a technology for embedding various types of information in digital content. In general, information for protecting copyrights and proving the validity of data is embedded as a watermark. Watermarked content can prove its origin, thereby protecting the data.

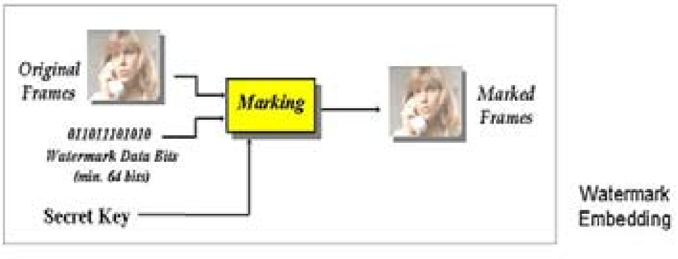


Fig. 5:

3. Authenticator Watermark

In this module we encrypt the data embedded image. The purpose of authenticator watermark of a block is invariant in the watermark embedding process; hence the watermark can be extracted without referring to the original content .The encryption and decryption techniques used in this module.

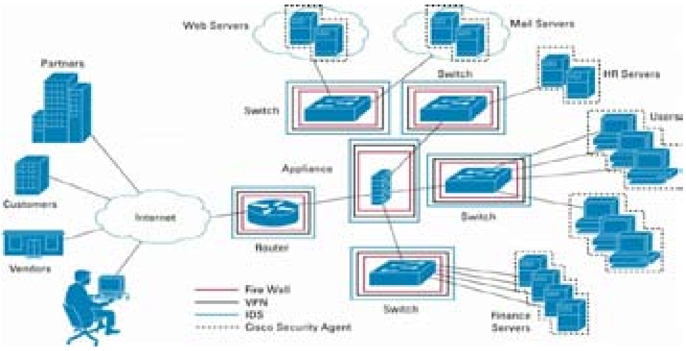


Fig. 6:

4. Spread Spectrum

We flip an edge pixel in binary images is equivalent to shifting the edge location horizontally one pixel and vertically one pixel. A horizontal edge exists if there is a transition between two neighboring pixels vertically and a vertical edge exists if there is a transition between two neighboring pixels horizontally. We use spread spectrum watermark morphological content.

E1 Spread Spectrum Radios
RF ODU
RF Modem
2E1 - 16E1
• 2 ~ 40 km
• 2E1 / 4E1 / 8E1 / 16E1
• Multiple Sequence Spread Spectrum
• 2.4GHz / 5.8GHz / 7GHz / 8GHz / 13GHz / 15GHz / 18GHz / 23GHz / 26GHz
RF ODU
RF Modem
2E1 - 16E1
K-Best technology

Fig. 7:

5. Watermarked Content

The watermarked content is obtained by computing the inverse for the main processing block to reconstruct its candidate pixels. Use this module we going to see the original and watermarked content.

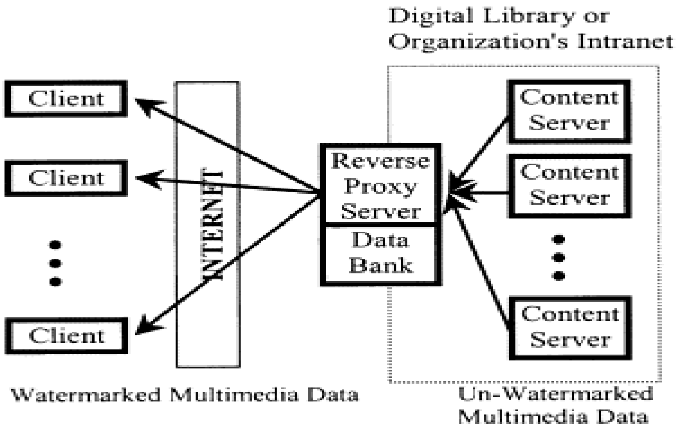

Digital Library or Organization's Intranet
Client
Client
Client
INTERNET
Reverse Proxy Server
Data Bank
Content Server
Content Server
Content Server
Watermarked Multimedia Data
Un-Watermarked Multimedia Data

Fig. 8:

V. Opening

A. Crisis Formulation

We assume the cover image x to be fixed and known only to the sender. The results of this paper will be general and independent of any particular choice of x. For this reason, we simply write D(y) , D(x, y) = Pn i=1 ρi(yi), while all quantities derived from Y or D(y) should be seen as being classed by x. The distortion function and its parameters are known only to the sender and not to the receiver. We assume that the embedding algorithm replaces the original cover x with y ∈ Y obtained as a realization of a random variable Y defined over Y and distributed according to π, π(y) P(Y = y). If the receiver knew x, the sender could send up to H(π) bits on average while introducing the average distortion Eπ[D], where

$$H(\pi) = -\sum_{y\in\mathcal{Y}} \pi(y) \log \pi(y), \qquad E_\pi[D] = \sum_{y\in\mathcal{Y}} \pi(y) D(y).$$

By the construction of the method, the knowledge of x to the receiver has no effect on the bounds between the above quantities as long as x is known to the sender. Having defined the distortion function, the sender is interested in the following optimization problems:

The task of embedding while minimizing distortion can assume two forms:

- freight-limited sender (FLS): embed a fixed average payload of m bits while minimizing the average

$$\underset{\pi}{\text{minimize}}\ E_\pi[D] \qquad \text{subject to } H(\pi) = m.$$

- Distortion-limited sender (DLS): maximize the average payload while introducing a fixed average distortion

$$\underset{\pi}{\text{maximize}}\ H(\pi) \qquad \text{subject to } E_\pi[D] = D_\epsilon.$$

The problem of embedding a fixed-size message while minimizing the total distortion D (the PLS) is more commonly used in steganography when compared to the DLS. When the distortion function is content-driven, the sender may choose to maximize the payload with a constraint on the overall distortion. This DLS corresponds to a more intuitive use of steganography since images with different level of noise and texture can carry different amount of hidden payload and thus the distortion should be fixed instead of the payload (as long as the distortion corresponds to statistical detectability).

B. Act Bounds and Comparison Metrics

The problems described above bear relationship to the problem of source coding with a fidelity criterion as described by Shannon [18]. Problems (2) and (3) are dual to each other, meaning that the optimal distribution for the first problem is, for some value of D, also optimal for the second one. Following the maximum entropy principle [2, Th. 12.1.1], the optimal solution has the form of a Gibbs distribution

$$\pi(y) = \frac{\exp(-\lambda D(y))}{Z(\lambda)} \overset{(a)}{=} \prod_{i=1}^{n} \frac{\exp(-\lambda \rho_i(y_i))}{Z_i(\lambda)} \triangleq \prod_{i=1}^{n} \pi_i(y_i),$$

Where the parameter λ ∈ [0,∞) has to be obtained from the corresponding constraints (2) and (3) by solving an algebraic equation, $Z(\lambda) = \sum_{y\in\mathcal{Y}} \exp(-\lambda D(y)), Z_i(\lambda) = \sum_{y_i\in\mathcal{I}_i} \exp(-\lambda\rho_i(y_i))$ are the corresponding partition functions. Step (a) follows from the additively of D, which also leads to mutual independence of individual stage pixels yi given x. Finally, the impact of embedding can be simulated by changing each pixel i with probability An established way of evaluating practical coding algorithms in steganography is to compare the embedding efficiency e(α) = αn/Eπ[D] for a fixed expected relative payload α = m/n with the upper bound derived from (4). When the number of changes is minimized, e is the average number of bits hidden per one change. For general ρi, the interpretation of this metric becomes less clear. A different and more easily interpretable metric is to compare the payload, m, of an embedding algorithm w.r.t. the payload, m MAX, of the optimal DLS for a fixed Do,

$$l(D_\epsilon) = \frac{m_{\mathrm{MAX}} - m}{m_{\mathrm{MAX}}},$$

Which we call the coding loss.

C. Twofold Embedding Operation

We start by describing the special case of a binary embedding operation and review a practical coding construction for this problem. In Section 4, we generalize this approach to operations with a larger cardinality. Since the operation is binary, we assume that $\mathcal{I}_i = \{x_i, y_i\}$.

In what follows, the only value the receiver needs to know is the number of message bits m he wants to receive. This information can be communicated in the same stage image using a different embedding scheme. According to (4), the coding algorithm for (2) or (3) is optimal if and only if it outputs pixel yi with probability

$$\pi_i(y_i) = \frac{\exp(-\lambda\rho_i(y_i))}{\exp(-\lambda\rho_i(x_i)) + \exp(-\lambda\rho_i(y_i))} = \frac{\exp(-\lambda\varrho_i)}{1+\exp(-\lambda\varrho_i)},$$

Where i = ρi(yi) -ρi(xi).2 For a fixed value of λ, the values i, i = 1, . . . , n, form sufficient statistic for π.

$$D'(y) = \sum_{i=1}^{n} \varrho_i \cdot [x_i \neq y_i].$$

A solution to the PLS with binary embedding operation can be used to derive the following "flipping lemma" that we will heavily used.

1. Useful Coding Algorithms

For a binary embedding operation, both types of senders can be realized in practice using Clause coding when the message miss communicated as a Clause of a linear policy d = HP(y) with parity-check matrix H ∈ {0, 1}m×n, where P : X → {0, 1} is a parity function shared between the sender and the receiver, e.g., P(x) = x mod 2. The PLS problem (2) then becomes

$$y = \arg \min_{\mathbb{H}\mathcal{P}(y)=m} D(y).$$

Ith m/n = const., this construction is asymptotically (w.r.t. n) optimal with high probability when the elements of H are chosen randomly. Such policy's are however highly impractical due to the exponential complexity of solving.

A practical algorithm for solving (2) based on Clause outline policy's (STCs) was proposed in [8]. It uses a pseudo randomly constructed, banded matrix H with a band of height h. For such matrices, the Vitoria algorithm finds the optimal solution to (8) with complexity exponential w.r.t. the constraint height h. For small values of h ∈ {7… 13}, STCs achieve a coding loss between 5% to 10% for various values of {i}n i=1 and arbitrary relative payloads α ∈ [0, 1] [7]. Since the whole cover image can be processed at once, wet pixels can be handled as well. Although the algorithm described in [8] uses the costs ρi(xi) = 0, ρi(yi) ≥ 0, it in fact works without a modification with arbitrary costs i > −∞. STCs can also be used for solving the distortion-limited sender (3) for given costs {i|i = 1, . . . , n} and bound Do. An optimal algorithm for this problem would send m MAX = H(π) bits on average.3 Although the number of communicated bits is a random variable in this problem(recall that now Do is fixed), we fix the number of bits to m = m MAX(1−l′) and let the Viterbi algorithm find the optimal solution as in problem (2). The parameter l′ is the coding loss we expect the algorithm will achieve and is determined experimentally for a given constraint height h.

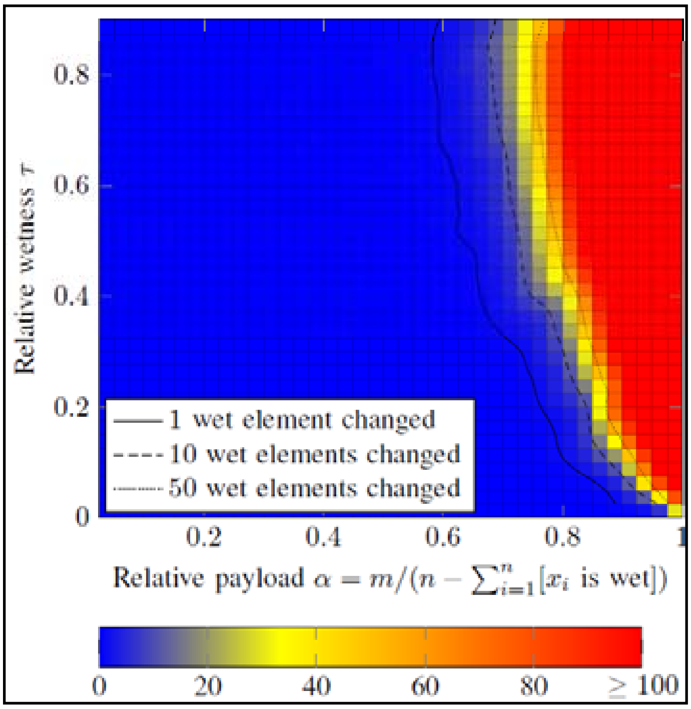


Fig. 9: Average number of wet elements out of n = 106 that need to be changed to find a solution to (8) using STCs with h = 11 versus relative payload α

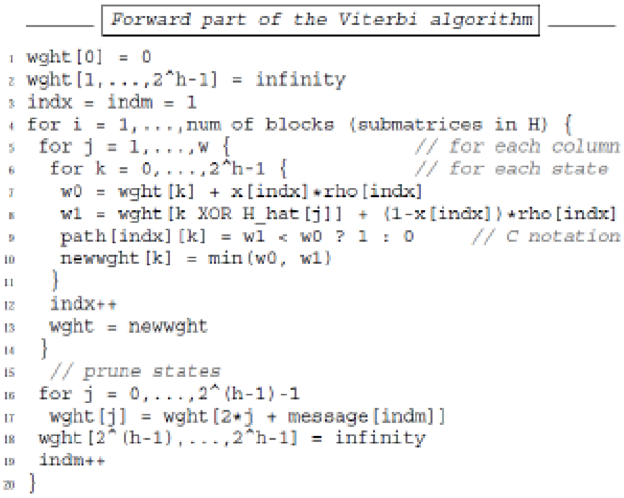
STCs were originally designed to handle payloads α ≤ 1/2. This assumption can be relaxed depending on the amount of wet pixels. Since wet pixels are not allowed to be changed, the maximum number of bits we can communicate is n –Pn i=1[xi is wet] (we define τ = Pn i=1[xi is wet]/n as the relative wetness). For this reason, the relative payload is commonly defined as α = m/(n−Pn i=1[xi is wet]) ∊ [0, 1]. When both τ and α are large, the Viterbi algorithm may need to change some wet elements due to the banded structure of H. This may be acceptable if this number is small, say 5 out of 106. Figure 1 shows the average number of wet elements out of n = 106 required to be changed in order to solve (8) for STCs with h = 11. The exact value of͇j is irrelevant in this experiment as long as it is finite. This experiment suggests that STCs can be used with arbitrary τ as long as α ≤ 0.7.

VI. Phrase-Outline Policy's

In this section, we focus on solving the binary PLS problem with distortion function (10) and modify a standard outline coding strategy for steganography. The resulting policy's are called the Clause-outline policy's. These policies will serve as a building block for non-binary PLS and DLS problems the construction behind STCs is not new from an Information-theoretic perspective, since the STCs are convolution policy's represented in a dual domain. However, STCs are very interesting for practical steganography since they allow solving both embedding problems with a very small coding loss over a wide range of distortion profiles even with wet pixels. The same policy can be used with all profiles making the embedding algorithm practically universal. STCs offer general and state-of-the-art solution for both embedding problems in steganography. Here, we give the description of the policy's along with their graphical representation, the Clause outline. Such construction is prepared for the Viterbi algorithm, which is optimal for solving (12). Important practical guidelines for optimizing the policy's and using them for the wet paper channel are also covered. Finally, we study the performance of these policy's by extensive numerical simulations using different distortion profiles including the wet paper channel. Clause-outline policy's targeted to applications in steganography were described in [17], which was written for practitioners. In this paper, we expect the reader to have a working knowledge of convolution policy's which are often used in data-hiding applications such as digital watermarking. Convolution policies are otherwise described For a complete example of the Viterbi algorithm used in the context of STCs

Our main goal is to develop efficient Clause-coding schemes for an arbitrary relative payload α with the main focus on small relative payloads (think of α ≤ 1/2 for

Example). In Steganography, the relative payload must decrease with increasing size of the cover object in order to maintain the same level of security, which is a consequence of the square root law [38]. Moreover, recent results from steganalysis in both spatial [39] and DCT domains [40] suggest that the secure payload for digital image Steganography is always far below 1/2. Another reason for targeting smaller payloads is the fact that as α → 1, all binary embedding



Forward part of the Viterbi algorithm

```
wght[0] = 0
wght[1,...,2^h-1] = infinity
indx = indm = 1
for i = 1,...,num of blocks (submatrices in H) {
 for j = 1,...,w {                          // for each column
  for k = 0,...,2^h-1 {                     // for each state
   w0 = wght[k] + x[indx]*rho[indx]
   w1 = wght[k XOR H_hat[j]] + (1-x[indx])*rho[indx]
   path[indx][k] = w1 < w0 ? 1 : 0          // C notation
   newwght[k] = min(w0, w1)
  }
  indx++
  wght = newwght
 }
  // prune states
 for j = 0,...,2^(h-1)-1
  wght[j] = wght[2*j + message[indm]]
 wght[2^(h-1),...,2^h-1] = infinity
 indm++
}
```

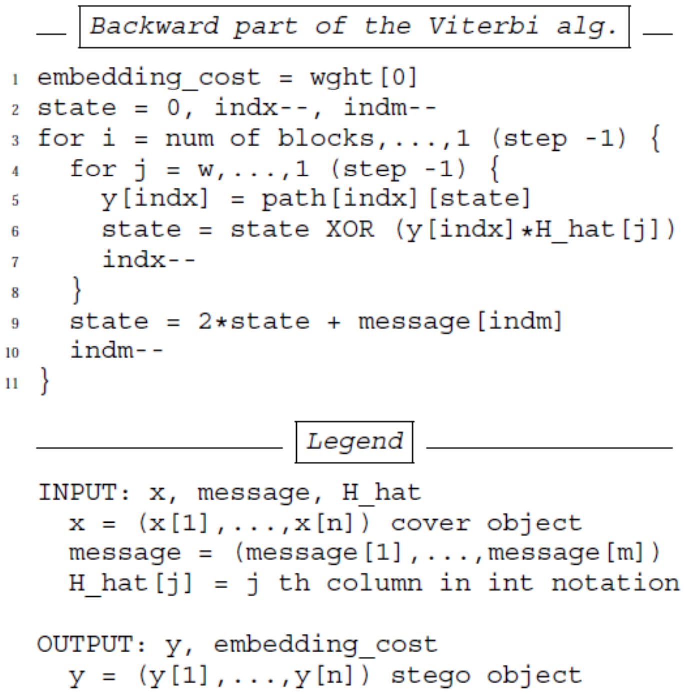
Algorithms tend to introduce changes with probability 1/2, no matter how optimal they are. Denoting with R = (n − m)/n the rate of the linear policy C, then α → 0 translates to R = 1 − α → 1, which is characteristic for applications of Clause coding in Steganography.

A. From Convolution Policy's to Clause-Outline Policy's

Since Shannon [21] introduced the problem of source coding with a fidelity criterion in 1959, convolution policy's were probably the first "practical" policy's used for this problem [41]. This is because the gap between the bound on the expected per-pixel distortion and the distortion obtained using the optimal encoding algorithm (the Viterbi algorithm) decreases exponentially with the constraint length of the policy [41-42]. The complexity of the Viterbi algorithm is linear in the block length of the policy, but exponential in its constraint length (the number of outline states grows exponentially in the constraint length). When adapted to the PLS problem, convolution policy's can be used for Clause coding since the best stage image in (12) can be found using the Viterbi algorithm. This makes convolution policy's (of small constraint length) suitable for our application because the entire cover object can be used and the speed can be traded for performance by adjusting the constraint length. Note that the receiver does not need to know D since only the Viterbi algorithm requires this knowledge. By increasing the constraint length, we can achieve the average per-pixel distortion that is arbitrarily close to the bounds and thus make the coding loss (7) approach zero. Convolution policy's are often represented with shift-registers (see Chapter 48 in [37]) that generate the policy word from a set of information bits. In channel coding, policy's of rates R = 1/k for k = 2, 3 . . . are usually considered for their simple implementation. Convolution policy's in standard outline representation are commonly used in problems that are dual to the PLS problem, such as the distributed source coding [43]. The main drawback of convolution policy's, when implemented using shift-registers, comes from our requirement of small relative payloads (policy rates close to one) which is specific to steganography. A convolution policy of rate R = (k − 1)/k requires k − 1 shift registers in order to implement a scheme for α = 1/k. Here, unfortunately, the complexity of the Viterbi algorithm in this construction grows exponentially with k. Instead of using puncturing (see Chapter 48 in [37]), which is often used to construct high-rate convolution policy's, we prefer to represent the convolution policy in the dual domain using its parity-check matrix. In fact, Sidorenko and Zyablov [44] showed that optimal decoding of convolution policy's (our binary quantizer) with rates R = (k −1)/k can be carried out in the dual domain on the Clause outline with a much lower complexity and without any loss of performance. This approach is more efficient as α → 0 and thus we choose it for the construction of the policy's presented in this paper. In the dual domain, a policy of length n is represented by a parity-check matrix instead of a generator matrix as is more common for convolution policy's. Working directly in the dual domain allows the Viterbi algorithm to exactly implement the cosset quantizer required for the embedding function (12). The message can be extracted in a straightforward manner by the recipient using the shared parity-check matrix.

B. Description of Clause-Outline Policy's

Although Clause-outline policy's form a class of convolution policy's and thus can be described using a classical approach with shift-registers, it is advantageous to stay in the dual domain and describe the policy directly by its parity check matrix. The parity-check matrix H ∈ {0, 1}m×n of a binary Clause-outline policy of length n and co dimension m is obtained by placing a small sub matrix ˆH of size h × w along the main diagonal as in Figure 2. The sub matrices ˆH are placed next to each other and shifted down by one row leading to a sparse and banded H. The height h of the sub matrix (called the constraint height) is a design parameter that affects the algorithm speed and efficiency (typically, 6 ≤ h ≤ 15). The width of ˆH is dictated by the desired ratio of m/n, which coincides with the relative payload α = m/n when no wet pixels are present. If m/n equals to 1/k for some k ∈ N, select w = k. For general ratios, find k such that 1/(k+1) < m/n < 1/k. The matrix H will contain a mix of sub matrices of width k and k+1 so that the final matrix H is of size m×n. In this way, we can create a parity-check matrix for an arbitrary message and policy size. The sub matrix ˆH acts as an key parameter shared between the sender and the receiver and its choice is discussed in more detail in Section V-D. For the sake of simplicity, in the following description we assume m/n = 1/w and thus the matrix H is of the size b × (b ∈ w), where b is the number of copies of ˆH in H. Similar to convolution policy's and their outline representation, every policyword of an STC C = {z ∈ {0, 1} n|Hz = 0} can be represented as a unique path through a graph called the Clause outline. Moreover, the Clause outline is parameterized by m and thus can represent members of arbitrary co set C(m) = {z ∈ {0, 1} n|Hz = m}. An example of the Clause outline is shown in Figure 2. More formally, the Clause outline is a graph consisting of b blocks, each containing 2h(w + 1) nodes organized in a grid of w + 1 columns and 2h rows. The nodes between two adjacent Columns form a bipartite graph,



```
Backward part of the Viterbi alg.

embedding_cost = wght[0]
state = 0, indx--, indm--
for i = num of blocks,...,1 (step -1) {
  for j = w,...,1 (step -1) {
    y[indx] = path[indx][state]
    state = state XOR (y[indx]*H_hat[j])
    indx--
  }
  state = 2*state + message[indm]
  indm--
}

Legend

INPUT: x, message, H_hat
  x = (x[1],...,x[n]) cover object
  message = (message[1],...,message[m])
  H_hat[j] = j th column in int notation

OUTPUT: y, embedding_cost
  y = (y[1],...,y[n]) stego object
```

i.e., all edges only connect nodes from two adjacent columns. Each block of the outline represents one sub matrix ˆH used to obtain the parity-check matrix H. The nodes in every column are called states. Each z ∈ {0, 1}n satisfying Hz = m is represented as a path through the Clause outline which represents the process of calculating the Clause as a linear combination of the columns of H with weights given by z. Each path starts in the leftmost all-zero state in the outline and extends to the right. The path shows the step-by-step calculation of the (partial) Clause using more and more bits of z. For example, the first two edges in Figure 2, that connect the state 00 from column p0 with states 11 and 00 in the next column, correspond to adding (P(y1) = 1) or not adding (P(y1) = 0) the first column of H to the Clause, respectively.4 At the end of the first block, we terminate all paths for which the first bit of the partial Clause does not match m1. This way, we obtain a new column of the outline, which will serve as the starting column of the next block. This column merely illustrates the transition of the outline from representing the partial Clause (s1. . . sh) to (s2. . . sh+1). This operation is repeated at each block transition in the matrix H and guarantees that 2h states are sufficient to represent the calculation of the partial Clause throughout the whole Clause outline. If P(xl) = 0, then the horizontal edges (corresponding to not adding the lth column of H) have a weight of 0 and the edges corresponding to adding the lth column of H have a weight of If P(xl) = 1, the roles of the edges are reversed. Finally, all edges connecting the individual blocks of the outline have zero weight. The embedding problem (12) for binary embedding can now be optimally solved by the Viterbi algorithm with time and space complexity O(2hn). This algorithm consists of two parts, the forward and the backward part. The forward part of the algorithm consists of n + b steps. Upon finishing the ith step, we know the shortest path between the leftmost all-zero state and every state in the ith column of the outline. Thus in the final, n + bth step, we discover the shortest path through the entire outline. During the backward part, the shortest path is traced back and the parities of the closest stego object P(y) are recovered from the edge labels.

VII. Multi-Layered Construction

In this section, we introduce a multi-layered construction which has been largely motivated by [22] and can be considered as a generalization of this work. The main idea is to decompose the problems (2) and (3) with a non-binary embedding operation into a sequence of similar problems for a binary embedding operation and then use the results of Section 3. Let |Ii| = 2L for some integer L ≥ 0 and let P1, . . . ,PL be parity functions uniquely describing all 2L elements in Ii, i.e., (xi 6= yi) ⇒ ∃j,Pj(xi) 6= Pj(yi) for all xi, yi ∊ Ii and all i ∊ {1, . . . , n}. For example, Pj(x) can be defined as the jth LSB of x. The individual sets Ii can be enlarged to satisfy the size constraint by setting the costs of added elements to ∞. The optimal algorithm for (2) and (3) sends the stego symbols by sampling from the optimal distribution (4) with some λ. LetYi be the random variable defined over Ii representing the ith stego symbol. Due to the assigned parities, Yi can be represented as Yi = (Y 1 i , . . . , Y L i) with Y j i corresponding to the jth parity function. We construct the embedding algorithm by induction over L, the number of layers. By the chain rule, for each i the entropy H(Yi) can be decomposed into

$$H(\mathrm{Y}_i) = H(Y_i^1) + H(Y_i^2, \ldots, Y_i^L | Y_i^1).$$

This tells us that H(Y 1 i) bits should be embedded by changing the first parity of the ith pixel. In fact, the parities should be distributed according to the marginal distribution P(Y 1 i). Using the flipping lemma, this task is equivalent to a PLS, which can be realized in practice using STCs as reviewed in Section 3.1. To summarize, in the first step we embed m1 = Pn i=1 H(Y 1 i) bits on average. After the first layer is embedded, we obtain the parities P1 (yi) for all stego pixels. This allows us to calculate the clausal probability P(Y 2i. . . Y Li |Y 1 i = P1 (yi)) and use the chain rule again, for example w.r.t Y 2 i. In the second layer, we embedm2 = Pn i=1 H(Y 2 i |Y 1 i = P1 (yi)) bits on average. In total, we have L such steps fixing one parity value at a time knowing the result of the previous parities. Finally, If all individual layers are implemented optimally, being fixed can be arbitrary.

Example (1 embedding): For simplicity, let xi = 2, Ii = {1, 2, 3}, ρi(1) = ρi(3) = 1, and ρi(2) = 0 for i ∊ {1, . . . , n} and large n. For such ternary embedding, we use two LSBs as their parities. Suppose we want to solve the problem (2) with α = 0.9217, which leads to λ = 2.08, P(Yi = 1) = P(Yi = 3) = 0.1, and P(Yi = 2) = 0.8. To make |Ii| a power of two, we also include the symbol 0 and define ρi(0) = ∞ which implies P(Yi = 0) = 0. Let yi = (y2i, y1i) be a binary representation of yi ∊ {0 . . . 3}, where y1 i is the LSB of yi. Starting from the LSBs as in [22], we obtain P(Y 1i = 0) = 0.8. If the LSB needs to be changed, then P(Y 2i = 0|Y 1i = 1) = 0.5 whereas P(Y 2i = 0|Y 1
i = 0) = 0. In practice, the first layer can be realized by any Clause coding scheme minimizing the number of changes and embedding. The second layer can be implemented with wet paper policy's [12], since we need to either embed one bit or leave the pixel unchanged (relative payload is 1). If the weights of symbols 1 and 3 were slightly changed, however, we would have to use STCs in the second layer, which causes a problem due to the large relative payload (α = 1) combined with large wetness (τ = 0.8) (see Figure 1). The opposite decomposition starting with the MSB y2i will reveal that P(Y 2i = 0) = 0.1, P(Y 1i = 0|Y 2i = 0) = 0, and P(Y 1i = 0|Y 2i = 1) = 0.8/0.9. Both layers can now be easily implemented by STCs since here the wetness is not as severe (τ = 0.1).

A. Practical Embedding Construction

We have implemented the above multi-layered construction based on STCs and present here a practical embedding scheme that was largely motivated by [6, 16], which contain the justification and motivation of the design elements that appear below. Let x ∊ {0 . . . 255}$_{\mathrm{n1×n2}}$ be an n1 × n2 grayscale cover image, n = n1n2, represented in the spatial domain. Define the co-occurrence matrix computed from horizontal pixel differences

$$D^{\rightarrow}_{i,j}(\mathrm{x}) = x_{i,j+1} - x_{i,j},\ i = 1, \ldots, n_1,$$

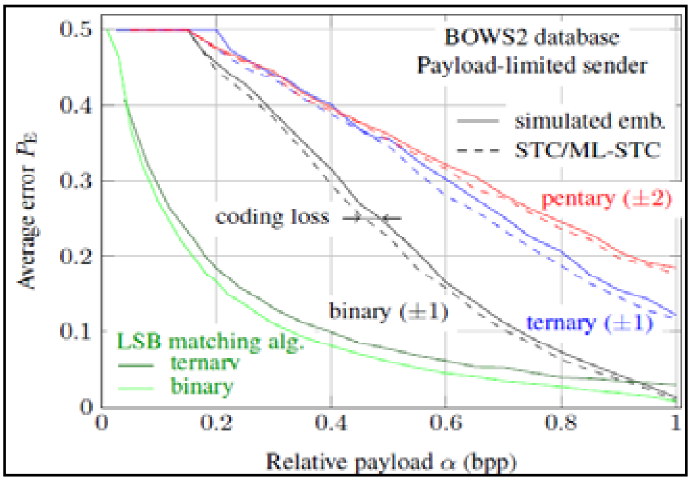


Fig. 10: Comparison of LSB matching with optimal binary and ternary coding with embedding algorithms based on the additive distortion measure (10) using embedding operations of three different cardinalities.

All tests were carried out on the BOWS2 database [1] containing approximately 10800 grayscale images with a fixed size of 512 × 512 pixels coming from rescaled and cropped natural images of various sizes. Steganalysis was implemented using the second-order SPAM feature set with T = 3 [15]. The image database was evenly divided into training and a testing set of cover and stego images, respectively.

$j = 1, \ldots, n_2 - 1$:

$$A^{\rightarrow}_{p,q,r}(\mathrm{x}) = \sum_{i=1}^{n_1}\sum_{j=1}^{n_2-3} \frac{[(D^{\rightarrow}_{i,j}, D^{\rightarrow}_{i,j+1}, D^{\rightarrow}_{i,j+2})(\mathrm{x}) = (p,q,r)]}{n_1(n_2-3)}$$

where $[(D^{\rightarrow}_{i,j}, D^{\rightarrow}_{i,j+1}, D^{\rightarrow}_{i,j+2})(\mathrm{x}) = (p,q,r)] = [(D^{\rightarrow}_{i,j}(\mathrm{x}) = p)\&(D^{\rightarrow}_{i,j+1}(\mathrm{x}) = q)\&(D^{\rightarrow}_{i,j+2}(\mathrm{x}) = r)]$. Clearly, $A^{\rightarrow}_{p,q,r}(\mathrm{x}) \in [0,1]$ is the normalized count of neighboring quadruples of pixels $\{x_{i,j}, x_{i,j+1}, x_{i,j+2}, x_{i,j+3}\}$ with differences $x_{i,j+1} - x_{i,j} = p$, $x_{i,j+2} - x_{i,j+1} = q$, and $x_{i,j+3} - x_{i,j+2} = r$ in the entire image. The superscript arrow "→" denotes the fact that the differences are computed by subtracting the left pixel from the right one. Similarly, we define matrices $A^{\swarrow}_{p,q,r}(\mathrm{x})$, $A^{\uparrow}_{p,q,r}(\mathrm{x})$, and $A^{\searrow}_{p,q,r}(\mathrm{x})$. Let $y_{i,j}\mathrm{x}_{\sim i,j}$ be an image obtained from x by replacing the (i,j)th pixel with value $y_{i,j}$. Finally, we define the distortion measure $D(\mathrm{y}) = \sum_{i=1}^{n_1}\sum_{j=1}^{n_2} \rho_{i,j}(y_{i,j})$ as

$$\rho_{i,j}(y_{i,j}) = \sum_{\substack{p,q,r\in\{-255,\ldots,255\}\\ s\in\{\rightarrow,\nearrow,\uparrow,\nwarrow\}}} w_{p,q,r}|A^{s}_{p,q,r}(\mathrm{x}) - A^{s}_{p,q,r}(y_{i,j}\mathrm{x}_{\sim i,j})|, \quad (10)$$

where $w_{p,q,r} = 1/(1+\sqrt{p^2+q^2+r^2})$ are heuristically chosen weights.

A soft-margin support-vector machine was trained using the Gaussian kernel. The kernel width and the penalty parameter were determined using five-fold cross validation $(C, \gamma) \in \{(10^k, 2^{j-d}) | k \in \{-3, \ldots, 4\}, j \in \{-3, \ldots, 3\}\}$ the grid where d is the binary logarithm of the number of features. We report the results using a measure frequently used in steganalysis – the minimum average classification error PE = (PFA + PMD)/2, where PFA and PMD are the false alarm and missed-detection probabilities. Figure 2 contains the comparison of embedding algorithms implementing the PLS and DLS with the costs (10). All algorithms are contrasted with LSB matching simulated on the binary and ternary bounds. To compare the effect of practical policy's, we first simulated the embedding algorithm as if the best policy's were available and then compared these results with algorithms implemented using STCs with h = 10. Both types of senders are implemented with binary, ternary $(\mathcal{I}_i = \{x_i - 1, \ldots, x_i + 1\})$, and pentary $(\mathcal{I}_i = \{x_i - 2, \ldots, x_i + 2\})$ Embedding operations. Before embedding, the binary embedding operation was initialized to Ii = {xi, yi} with yi randomly chosen from {xi−1, xi+1}.

VIII. Conclusion

This paper describes near-optimal policy's for minimal-distortion Steganography implemented in the following manner. Before embedding in a given cover, the sender first specifies for each cover element xi the set of values yi ∊ Ii to which xi can change and the associated cost of making this modification, ρi(yi). The problem is to communicate a payload of a certain size with minimal expected embedding distortion obtained as a sum of individual pixel costs ρi (alternatively, embed the largest possible payload for a given bound on the expected distortion). The proposed approach works for an arbitrary cost assignment and arbitrary sets Ii, which can even be different for every i. The method is a generalization of previously proposed Clause-outline policies and other special cases, including matrix embedding and wet paper policies. The merit of the proposed method is demonstrated experimentally by implementing it for binary, ternary, and pentary embedding operations in spatial domain and showing an improvement in statistical detectability measured by a blind steganalyzer. This construction is not limited to embedding with larger amplitudes but can be used, e.g., for embedding in color images, where the LSBs of all three colors can be seen as 3- bit symbols on which the cost functions are defined. Applications outside the scope of digital images are possible as long as the costs can be meaningfully defined. Matlab and C++ implementation of multi-layered STCs is available at http://dde.binghamton.edu/download.

References

[1] "Generalization of the ZZWembedding construction for steganography", [5], 4(3), pp. 564–569, Sep. 2009.

[2] X. Zhang, W. Zhang, S. Wang,"Efficient double layered steganographic embedding", Electronics Letters,43:, pp. 482–483, Apr. 2007.

[3] R. Crandall,"Some notes on steganography", Steganography Mailing List, [Online] Available: http://os.inf.tudresden.de/westfeld/crandall.pdf, 1998.

[4] T. Filler, J. Fridrich,"Wet ZZW construction for steganography", In First IEEE Intern. Workshop on Information Forensics and Security, London, UK, Dec. 2009.

[5] IEEE Trans. on Information Forensics and Security.

[6] T. Filler, J. Fridrich.,"Gibbs construction in steganography", 2010. To appear in December issue.

[7] T. Filler, J. Judas, J. Fridrich,"Minimizing additive distortion in steganography using Clause-outline policy's", [5], 2010. Submitted.

[8] T. Filler, J. Judas, J. Fridrich,"Minimizing embedding impact in Steganography using outline-policyd quantization", In Proceedings SPIE, Electronic Imaging, Vol. 7541, pp. 05–01–05–14, Jan. 17–21, 2010.

[9] J. Fridrich,"Steganography in Digital Media: Principles, Algorithms, and Applications", Cambridge University Press, 2009.

[10] J. Fridrich, T. Filler.,"Practical methods for minimizing embedding impact in steganography. ProceedingsSPIE, Electronic Imaging, Vol. 6505, pp. 02–03,San Jose, CA, Jan. 29–Feb. 1, 2007.

[11] J. Fridrich, M. Goljan, D. Soukal,"Wet paper policy'swith improved embedding efficiency", [5], 1(1), pp. 102–110, 2006.

[12] J. Fridrich, M. Goljan, D. Soukal, P. Lisonˇek,"Writingon wet paper", In IEEE Transactions on Signal Processing,Special Issue on Media Security, Vol. 53, pp. 3923–3935, Oct. 2005. (journal version).

[13] J. Fridrich, D. Soukal,"Matrix embedding for largepayloads", [5], 1(3), pp. 390–394, 2006.

[14] Y. Kim, Z. Duric, D. Richards,"Modified matrix encodingtechnique for minimal distortion steganography", In Information Hiding, 8th Intern. Workshop, volume 4437 of Lecture Notes in Computer Science (LNCS), pages 314–327, Alexandria, VA, Jul. 10–12, 2006.

[15] T. Pevný, P. Bas, J. Fridrich. Steganalysis by subtractivepixel adjacency matrix", In Proceedings of the11th ACM Multimedia & Security Workshop, pp. 75–84, Princeton, NJ, Sep. 7–8, 2009.

[16] T. Pevný, T. Filler, P. Bas,"Using high dimensional image models to perform highly undetectable steganography", In Inform. Hiding, 12th Intern. Conference,LNCS, Calgary, Alberta, Canada, Jun. 28–30 2010.

[17] D. Schönfeld, A. Winkler,"Embedding with Clause encoding based on BCH policy's", In Proceedings of the 8th ACM Multimedia & Security Workshop, pp. 214–223, Geneva, Switzerland, Sep. 26–27, 2006.

[18] C. E. Shannon,"Coding theorems for a discrete source with a fidelity criterion", IRE Nat. Conv. Rec., 4, pp. 142–163,1959.

[19] A. West feld,"High capacity despite better steganalysis(F5 – a steganographic algorithm)", In Information Hiding, 4th International Workshop, Vol. 2137 of LNCS, pp. 289–302, Pittsburgh, PA, Apr. 25–27, 2001.

[20] R. Zhang, V. Sachnev, H. J. Kim,"Fast BCH Clause coding for steganography", In InformationHiding, 11th International Workshop, Vol. 5806 of LNCS, pp. 31–47, Darmstadt, Germany, Jun. 7–10, 2009.

[21] P. Bas, T. Furon, BOWS-2, [Online] Available: http://bows2.gipsalab.inpg.fr, July 2007.W. Zhang and X.Wang.

[22] Y. Kim, Z. Duric, D. Richards,"Modified matrix encoding techniquefor minimal distortion steganography", in Information Hiding, 8th International Workshop (J. L. Camenisch, C. S. Collberg, N. F. Johnson,and P. Sallee, eds.), vol. 4437 of Lecture Notes in Computer Science,(Alexandria, VA), pp. 314–327, Springer-Verlag, New York, July 10–12,2006.

[23] R. Zhang, V. Sachnev, H. J. Kim,"Fast BCH Clause codingfor steganography", in Information Hiding, 11th International

Workshop (S. Katzenbeisser and A.-R. Sadeghi, eds.), Vol. 5806 of Lecture Notes in Computer Science, (Darmstadt, Germany), pp. 31–47, Springer-Verlag,New York, June 7–10, 2009.

[24] V. Sachnev, H. J. Kim, R. Zhang,"Less detectable JPEG steganography method based on heuristic optimization and BCH Clause coding", in Proceedings of the 11th ACM Multimedia & Security Workshop (J. Dittmann, S. Craver, J. Fridrich, eds.), (Princeton, NJ), pp. 131– 140, September 7–8, 2009.

[25] T. Pevný, T. Filler, P. Bas,"Using high-dimensional image models to perform highly undetectable steganography", in Information Hiding, 12th International Workshop (P. W. L. Fong, R. Böhme, and R. Safavi-Naini, eds.), vol. 6387 of Lecture Notes

Mr. Sk.Karimulla is a student of B.V.C college of Engineering & Technology, Odalarevu. Presently he is pursuing his M.TECH (CSE) from this college. He received his M.C.A from Adikavi Nannaya University In the year 2009 and B.Sc from Andhra University In the year 2006.

Mr. G.SASIBHUSANA RAO is an Associate professor in CSE Department at B.V.C college of Engineering & Technology, Odalarevu. He is an excellent teacher publishing papers in national and international conferences. He is a life member of ISTE and CSI. His areas of interest are Software Engineering, Software Project Management, Software Testing Methodologies, Network Security, Computer Networks, Design and Analysis of Algorithms, Theory of Computation, Pattern Recognition, Database Management Systems, principles of programming languages, IT workshop, software quality assurance and testing. He received his Bachelors and Masters degree in the field of Computer science and Information technology and he designed and guided many Projects.