STAT 5703 Data Mining I - Winter 2018

STEGANALYSIS

Enrique Reveron

Michael Armanious

Alex El-Hajj

Muneer Khan

March 28, 2018

Table of Contents

1. What is Steganography and Steganalysis?

Steganography is all about hiding information in different mediums (e.g., image, audio, video) innocuously.

Steganalysis is all about detecting this hidden information. It detects changes (normally due to embedding hidden information) in statistical properties of the original or cover media. [1] It can be passive (i.e. only the presence of the hidden message or steganographic algorithm is detected) or active (characteristics of hidden data, such as embedding location or length of the message is estimated).

2. Why is it important?

“Steganalysis has gained prominence in national security and forensic sciences since detection of hidden (ciphertext or plaintext) messages can lead to the prevention of disastrous security incidents.”

3. News articles

In 2001, US officials stated that they have suspicions that terrorists communicate using steganography in the internet.

“Hidden in the X-rated pictures on several pornographic Web sites and the posted comments on sports chat rooms may lie the encrypted blueprints of the next terrorist attack against the United States or its allies. It sounds far-fetched, but U.S. officials and experts say it's the latest method of communication being used by Osama bin Laden and his associates to outfox law enforcement. Bin Laden, indicted in the bombing in 1998 of two U.S. embassies in East Africa, and others are hiding maps and photographs of terrorist targets and posting instructions for terrorist activities on sports chat rooms, pornographic bulletin boards and other Web sites, U.S. and foreign officials say.”

A lot of articles, like the above, were published at that time and brought the world’s attention on the use of steganography. Government officials, especially in the US, have shown a large interest on steganography and steganalysis research over the last years.

4. Challenges

Steganalysis is a very challenging field because of the scarcity of knowledge about the specific characteristics of the cover media (an image, an audio or video file) that can be exploited to hide information and detect the same. The approaches adopted for steganalysis also sometimes depend on the underlying steganography algorithm(s) used.

5. Review of Literature

General Theories/Principles:

- The Kerckhoffs Principle: The system should remain secure under the assumption that the attacker knows the scheme. The interpretation differs when referring to stenography; the additional knowledge of the cover source is debatable.

- The Usability Principle (due to Kerckhoffs): The system should be straightforward for the average person to utilize precisely. For example, instead of expecting the user to apply a square root law, the stenographic software should implement the calculations automatically.

Specific vs. Universal (General) Steganalysis

A **Specific** approach represents a class of image steganalysis techniques that very much depend on the underlying steganographic algorithm used and have a high success rate for detecting the presence of the secret message if the message is hidden with the algorithm for which the techniques are meant for. If the data embedding manner of a steganographic scheme is *known*, the steganalyst can fully utilize the knowledge to devise a steganalytic detector targeted to such a steganographic scheme. Many successful specific steganalytic methods have been proposed in the past years. We discuss the main idea of some specific attacking methods.

Designing a detector which does not depend on the specific steganographic algorithm is called **universal/generic** steganography.

Although universal steganalysis is more general and less efficient that the specific

Two main interests for universal steganalysis are:

1. they are independent of the steganography algorithm
2. it is the only possible way to detect the use of steganography algorithm for which no specific steganalysis is known.

In short, the specific steganalysis answers the question: “Is this the medium which was embedded with the steganalysis algorithm?” and the universal steganalysis answers the question : “Is the medium a stego-medium?”

6. Steganalysis Algorithms

Steganalysis algorithms are techniques used to try to detect the hidden message in the media. The three most commonly used medias for Steganalysis algorithms are image, audio, and video.

6.1 Image

Algorithms for image steganalysis are primarily of two types: Specific and Generic/Universal. The image steganalysis techniques under both the specific and generic categories are often designed to detect the presence of a secret message and the decoding of the same is considered complementary not mandatory.

6.1.1 Least Significant Bit (LSB) embedding

There are several different techniques for concealing data inside of normal files. One of the most widely used and perhaps simplest to understand is the least significant bit technique, known commonly as LSB.

This technique changes the last few bits in a byte to encode a message, which is especially useful in something like an image, where the red, green, and blue values of each pixel are represented by eight bits (one byte) ranging from 0 to 255 in decimal or 00000000 to 11111111 in binary. But grayscaled images are also effective.

The LSB steganography method is divided into two parts: LSB Substitution and LSB Matching techniques. In LSB substitution technique the LSB of cover image is substituted by hidden secret message. In this technique the hidden message bit match with cover image LSB bit. If it is different than substitute the bit, otherwise leave as it is.

A few ideas for steganalysis in LSB images have been proposed. One of them, proposed by Fridrich et al.[3], used for colour images, is as follows: the relation of close colour pairs between the natural image and the stego-image is used. We then compute two ratios one for the natural image and one for the stego-image. These are the ratios of the close colour pairs over the number of unique colour divided by two. If the two ratios are equal then we conclude that the input image can be considered as the stego-image.

6.1.2 Palette Image Steganalysis

Palette image steganalysis is primarily used for GIF images. A GIF format contains 8 bits per pixel and a pixel has up to 256 distinct colours which are mapped to the 24-bit RGB colour space. Using LSB embedding on a GIF image would change the 24-bit RGB value of a pixel which could change the palette colour of the pixel. The steganalysis of a GIF stego image is conducted by performing a statistical analysis of the palette table in relation to the image and the detection is made when there is an appreciable increase in entropy (a measure of the variation in the palette colors).

6.1.3 Raw Image Steganalysis

Raw image steganalysis is mainly used for BMP format images. LSB embedding on BMP images causes the flipping of the two grayscale values. The embedding of the hidden message is more likely to result in averaging the frequency of occurrence of the pixels with the two gray-scale values. There are a few steganalysis techniques that can be used for raw image steganalysis. One of them which was proposed by Fridrich et. al. [3] was a steganalysis technique that studies color bitmap images for LSB embedding and it provides high detection rates for shorter hidden messages. This technique makes use of the property that the number of unique colors for a high quality bitmap image is half the number of pixels in the image. The new color palette that is obtained after LSB embedding is characterized by a higher number of close color pairs (i.e., pixel pairs that have a maximum difference of one count in either of the color planes).

6.1.4 JPEG Image Steganalysis

The JPEG is a popular cover image format used in steganography. Two well-known Steganography algorithms for hiding secret messages in JPEG images are: the F5 algorithm [4] and Outguess algorithm [5].

6.1.5 Generic Image Steganalysis Algorithms

The generic steganalysis algorithms, usually referred to as Universal or Blind Steganalysis algorithms, work well on all known and unknown steganography algorithms. These steganalysis techniques make use of the changes in certain natural features of the cover images when a message is embedded. The generic steganalysis algorithms are developed to precisely and maximally distinguish these changes

6.2 Audio

6.2.1 Steganography Algorithms

In audio steganography, a secret message is embedded into digitized audio signal which result slight alteration of binary sequence of the corresponding audio file. Moreover, audio signals have a characteristic redundancy and unpredictable nature that make them ideal to be used as a cover for covert communications to hide secret messages. Methods of hiding data in audio files are mainly divided into Low-Bit Encoding, Phase Encoding, Spread Spectrum, and Echo Data Hiding.

6.2.1.1 Low-bit Encoding

In Low-bit encoding the binary version of the secret data message is substituted with the least significant bit (LSB) of each sample of the audio cover file. This method cannot protect the hidden message from small modifications that can arise as a result of format conversion or lossy compression. There are other two disadvantages, the first one is that the human ear is very sensitive and can often detect the presence of single bit of noise into an audio file. Second disadvantage is that LSB coding is not very robust. Embedded information will be lost through a little modification of the stego-audio.

6.2.1.2 Phase Coding

The phase coding method works by substituting the phase of an initial audio segment with a reference phase that represents the data. Phase coding is based on the fact that the phase components of sound are not as perceptible to the human ear as noise is. Message bits are encoded as phase shifts in the phase spectrum of a digital signal. This leads to inaudible encoding in terms of the Signal-to-Perceived Noise Ratio (SPNR) and the secret message gets camouflaged in the audio signal, not detectable by the steganalysis methods based on SPNR.

The sequence of steps involved in phase coding is as follows:

1. The original audio signal is decomposed into smaller segments such that their length equals the size of the message that needs to be encoded.
2. A Discrete Fourier Transform (DCT) is then applied to each segment in order to create a phase matrix.
3. Phase differences between every pair of consecutive segments are computed.
4. Phase shifts between adjacent segments are identified. Although, the absolute phases of the segments can be altered, the relative phase differences between consecutive segments must be unchanged.
5. The new phase matrix is created using the new phase of the signals first segment and the set of original phase differences.
6. Based on the new phase matrix and the original magnitude matrix, the sound signal is regenerated by using inverse DFT and then by joining the sound segments together. The receiver is mandated to know the message length in order to use DFT and extract the embedded message from the cover signal.

To extract the secret message from the audio file, the receiver needs to know the segment length. The receiver can extract the secret message through different reverse process.

The disadvantage associated with phase coding is that it has a low data embedding rate due to the fact that the secret message is encoded in the first signal segment only. This situation can be overcome by increasing the length of the signals segment which in turn increases the change in the phase relations between each frequency component of the segment more drastically, making the encoding easier to detect. Thus, the phase coding method is useful only when a small amount of data, such as a watermark, needs to be embedded.

6.2.1.3 Spread Spectrum Coding

The basic Spread Spectrum (SS) coding method randomly spreads the bits of the secret data message across the frequency spectrum of the audio signal. This is equivalent to a system using the LSB coding method which randomly spreads the message bits over the entire audio file. However, unlike LSB coding, the SS coding method spreads the secret message using a code that is independent of the actual cover signal. Like the LSB coding method, the SS method can introduce noise to the audio file. This vulnerability can be tapped for steganalysis.

Two versions of SS can be used for audio steganography one is the direct sequence where the secret message is spread out by a constant called the chip rate and then modulated with a pseudo random signal whereas in the second method frequency-hopping SS, the audio file's frequency then interleaved with the cover-signal spectrum is altered so that it hops rapidly between frequencies.

6.2.1.4 Echo Hiding

With echo hiding, information is embedded by introducing an echo into the discrete audio signal. Like SS coding, echo hiding allows for a higher data transmission rate and provides superior robustness when compared to the noise-inducing methods. To successfully hide the data, three parameters of the echo need to be altered: amplitude, decay rate and offset (delay time) from the original signal. The echo is not easily resolved as all the three parameters are set below the human audible threshold limit. Also, the offset is altered to represent the binary message to be hidden. The first offset value represents a one (binary), and the second offset value represents a zero (binary). Once the encoding process is completed, the blocks are concatenated back together to form the final signal. To extract the secret message from the final stego-audio signal, the receiver must be able to break up the signal into the same block sequence used during the encoding process. Then the autocorrelation function of the signal's cepstrum which is the Forward Fourier Transform of the signal's frequency spectrum can be used to decode the message because it reveals a spike at each echo time offset, allowing the message to be reconstructed.

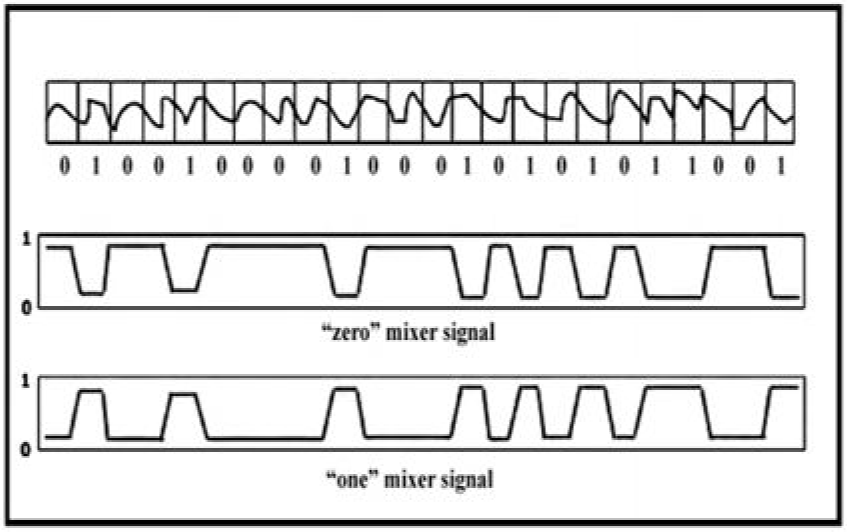


Figure 1: Echo Hiding Steganography Method

6.2.2 Steganalysis Algorithms

Audio steganalysis is very difficult due to the existence of advanced audio steganography schemes and the very nature of audio signals to be high-capacity data streams necessitates the need for scientifically challenging statistical analysis. To detect the presence/absence of secret message in audio by human ear is impossible. The embedding process generally alters the statistics of the carrier. Therefore, many steganalytic techniques are based on statistical model.

6.2.2.1 Phase and Echo Steganalysis

Zeng et. al [6] proposed steganalysis algorithms to detect phase coding steganography based on the analysis of phase discontinuities and to detect echo steganography based on the statistical moments of peak frequency [7]. The phase steganalysis algorithm explores the fact that phase coding corrupts the extrinsic continuities of unwrapped phase in each audio segment, causing changes in the phase difference [8]. A statistical analysis of the phase difference in each audio segment can be used to monitor the change and train the classifiers to differentiate an embedded audio signal from a clean audio signal.

6.2.2.2 Universal Steganalysis based on Recorded Speech

Johnson et. al [9] proposed a generic universal steganalysis algorithm that bases it study on the statistical regularities of recorded speech. Their statistical model decomposes an audio signal (i.e., recorded speech) using basis functions localized in both time and frequency domains in the form of Short Time Fourier Transform (STFT).

Time domain is the analysis of mathematical functions, physical signals or time series of economic or environmental data, with respect to time. In the time domain, the signal or function's value is known for all real numbers, for the case of continuous time, or at various separate instants in the case of discrete time. In the case of audio signal, the values are discrete.

The time domain and the frequency domain are both discrete and finite. Although finite, the time and frequency domains are both implicitly periodic.

The spectrograms collected from this decomposition are analyzed using non-linear support vector machines to differentiate between cover and stego-audio signals. This approach is likely to work only for high-bit rate audio steganography and will not be effective for detecting low bit-rate embeddings.

6.2.2.3 Use of Statistical Distance Measures for Audio Steganalysis

A group of statisticians calculated the distribution of various statistical distance measures on cover audio signals and stego audio signals in relation to their versions without noise and observed them to be statistically different [10]. The authors employed audio quality metrics to capture the deviations in the signal introduced by the embedded data. They designed an audio steganalyzer that relied on the choice of audio quality measures, which were tested depending on their perceptual or non-perceptual nature. The selection of the proper features and quality measures was conducted using the (i) ANOVA test [11] to determine whether there are any statistically significant differences between available conditions and the (ii) SFS (Sequential Floating Search) algorithm that considers the inter-correlation between the test features in ensemble [12]. Subsequently, two classifiers, one based on linear regression and another based on support vector machines were used and simultaneously evaluated for their capability to detect stego messages embedded in the audio signals.

6.2.2.4 Audio Steganalysis based on Hausdorff Distance

The audio steganalysis algorithm proposed by Liu et. al [24] uses the Hausdorff distance measure [13] to measure the distortion between a cover audio signal and a stego audio signal. The algorithm takes as input a potentially stego audio signal x and its de-noised version x' as an estimate of the cover signal. Both x and x' are then subjected to appropriate segmentation and wavelet decomposition to generate wavelet coefficients [14] at different levels of resolution. The Hausdorff distance values between the wavelet coefficients of the audio signals and their de-noised versions are measured. The statistical moments of the Hausdorff distance measures are used to train a classifier on the difference between cover audio signals and stego audio signals with different content loadings.

6.2.2.5 Audio Steganalysis for High Complexity Audio Signals

More recently, Liu et. al [15] proposed the use of stream data mining for steganalysis of audio signals of high complexity. Their approach extracts the second order derivative based Markov transition probabilities and high frequency spectrum statistics as the features of the audio streams. The variations in the second order derivative based features are explored to distinguish between the cover and stego audio signals. This approach also uses the Mel-frequency cepstral coefficients (the mel-frequency cepstrum (MFC) is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency) [16]-[17], widely used in speech recognition, for audio steganalysis.

6.3 Video

There is very little literature on video steganalysis algorithms. Some researchers have attempted to apply some image steganalysis algorithms to video on a frame-by-frame basis with low success. The reason for applying image algorithms on video is because videos are essentially a series of still images (frames).

One issue with this approach is related to redundant information present from frame-to-frame of a video. That is, images normally don't change much from one frame to the next (e.g. a standard 60 frame-per-second 4K video).

Hence, image steganalysis algorithms don't perform well as they are designed for single image watermark (secret information) detection vs. multi-image (video). That said, the algorithms described below have been relatively successful in detecting hidden information.

6.3.1 Video Steganalysis Exploring the Temporal Correlation between Frames

Budia et. al took advantage of the redundant information (e.g. pixels duplicated from frame-to-frame) and correlation between frames (using redundant information) in video as a deterrent against secret messages embedded by spread spectrum steganography, which is a system that hides and recovers a message in images while maintaining the original image size and dynamic range (i.e., the difference between the smallest and largest signal values – e.g., the difference in brightness of a video).

https://en.wikipedia.org/wiki/Dynamic_range#Video

Figure 1 below illustrates this system including embedding a secret binary message into the cover video (original video) to producing a stego video (video with hidden message) with the same original size. The secret message bits are embedded into the cover video in a signal called a watermark.

A steganalyst will detect the watermark by taking the average of correlated subsets (groupings of frames) of the video and then examining different statistics (e.g., kurtosis, variance and skewness) – linear collusion attack. The greater the correlation between frames, the greater the collusion performance.

Essentially, the algorithm groups correlated and similar parts or frames of the video together (treating it as one unit) to better examine differences and detect statistical discrepancies.

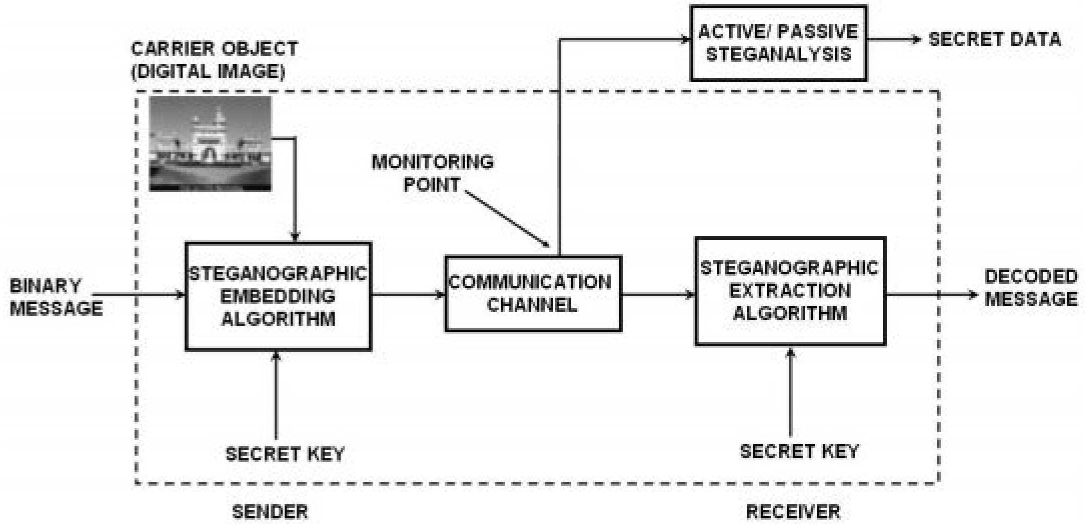
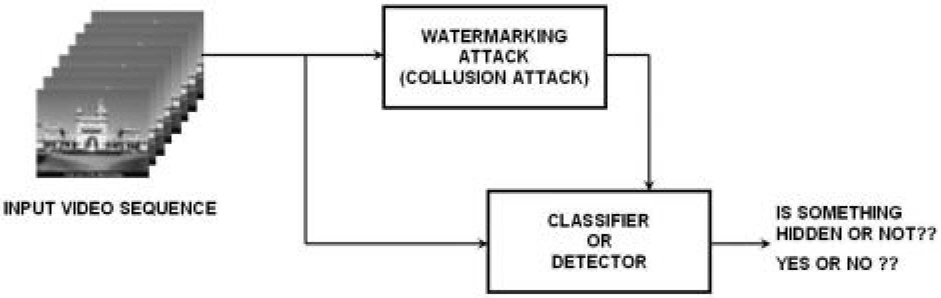


Figure 1: [18]



They created a framework (Figure 2 above) based on two stages [19]:

(i) A Watermarking attack stage to estimate the cover media from the potentially watermarked stego media and

(ii) A Pattern recognition stage for the detection of the steganographic activity.

6.3.2 Video Steganalysis based on Asymptotic Relative Efficiency (ARE)

The Motion-based Video Steganalysis algorithm uses asymptotic relative efficiency (a measure of the quality of sample estimators) to detect secret messages. This algorithm is for applications in which a subset (vs. all frames watermarked in the algorithm in 5.3.1) of the video frames are watermarked with the secret message and not all of them. The image frames are correlated. Steganalysis includes two phases:

(i) Signal processing phase

(ii) Detection phase

The signal processing phases emphasizes the presence of hidden information in the sequence of frames using a motion estimation scheme. Motion estimation is normally used to restore video by using surrounding video frames because of high correlation between frames. Similarly, in steganalysis:

(1) Assume one-by-one each frame is "missing"
(2) Reconstruct this "missing" frame by using the adjacent frames, but not the original "missing" frame.
(3) Compare this original "missing" frame with the reconstructed frame to check for contamination since the reconstructed frame was not created using the original "missing" frame. The higher the difference between the reconstructed frame and the "missing" frame, the higher the likelihood of data tampering. [20]

6.3.3 Video Steganalysis based on Mode Detection

Moscow State University (MSU) developed steganographic video software that can embed secret information in any file of AVI (Audio Video Interleave) format. The proposed steganalysis algorithm targets this software and can successfully extract the hidden message. The algorithm uses the correlation between neighbouring frames and finds a **distribution mode** across frames for 32X32 and 16X16 video pixel blocks. The algorithm compares the actual distribution mode found with what it should be and decides if there is contamination. [21]

6.3.4 Video Steganalysis based on Spatial and Temporal Prediction

This video steganalysis algorithm, using neighbouring frames and motion compensation, is designed for MPEG video coding. [22]

6.3.5 Other Video Steganalysis Algorithms

"Kancherla and Mukkamala's [23] video steganalysis method uses neural networks and support vector machines (redundancies) to find secret messages. Zhang et. al's [24] algorithm counter-attacks spread spectrum techniques. The algorithm uses the probability mass function of the inter-frame difference signal to show distortion caused by adding hidden information. Liu et. al [26] employ an algorithm using inter-frame correlation, collusion and feed forward neural networks.

7. Discussion of the Technical Material

Our application is divided into two parts:

(1) Steganography using the Stegasaur package in R

(2) Steganalysis using various data mining techniques

Stegasaur supports least-significant-bit (LSB) embedding on a diverse range of image types. In our example, we use a png image to implement LSB steganography. In the second part, we conduct image steganalysis using data mining techniques such as:

- clustering techniques and data reduction techniques(e.g., k-means, principal component analysis (PCA));
- classification techniques;
- support vector machine (SVM); and,
- neural networks (NN)

7.1 Stegasaur Package
The statistical model for LSB hiding considers the case of independent and identically distributed (i.i.d.) data samples. We assume the data to be one dimensional since the host samples are assumed to be i.i.d., without loss of generality . Suppose the i.i.d. host is $\{h_k\}_{k=1}^{N}$, where the intensity values h_k are the represented by 8 bits, that is, $h_k \in \{0,1,...,255\}$. We assume the hidden data $\{d_k\}_{k=1}^{N}$ is i.i.d. and,

$$\mathrm{P}\,(d_k = 0) = \mathrm{R}/2,\ \mathrm{P}\,(d_k = 1) = \mathrm{R}/2,$$

$$\mathrm{P}\,(d_k = NULL) = (1\text{-}\mathrm{R}),\ 0 < \mathrm{R} \leq 1.$$

If $d_k = NULL$, the hider does not hide in host sample h_k, if otherwise, the hider replaces the LSB of h_k with d_k.

7.2 Data Mining Techniques

7.2.1 K-means Clustering
K-means clustering is an unsupervised learning algorithm that classifies or divides n items into k clusters, where k is the number of clusters required. Each cluster has a unique centroid and are obtained by minimizing the sum of squared distances (Euclidean distances) between items and the corresponding centroid.

For k-means clustering, we start by dividing the image into clusters using pattern matching based on predefined colour palette range. Subsequently, a secret message is embedded in a selected cluster using steganographic techniques. When clusters are placed in their proper positions, this is when the image is created. Finally, the stego-image is then sent over the channel. The objective function

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 ,$$

where $\left\| x_i^{(j)} - c_j \right\|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster centre c_j, is an indicator of the distance of the *n* data points from their respective cluster centres.

7.2.2 Support Vector Machines
The objective of employing a support vector machine (SVM) algorithm is to learn a model which forecasts class tag of cases in the testing set. In terms of two-class classification, the SVM algorithm is one of the most robust classifiers. SVM can manage both linear and nonlinear classification problems. For linear discrete problems, SVM classifiers purely explore for a hyper-plane that distinguishes negative and positive instances (Cortes & Vapnik, 1995), (Vapnik, 1998), (Boser et al, 1992).

The SVMs present supervised ML (machine learning) methods based on binary classification. Binary classification divides the data into two classes, where each will have class value +1 or -1. For each data $(\overline{x}_i, \overline{y}_i)$ with i=1...N, $\overline{x}_i \in R^d$, and $\overline{y}_i$ ε {-1,+1}, binary classification $f(\overline{x}_i)$ results as follows.

$y_i = \{+1, f(\overline{x}_i) \geq 0$ (1)

$\{-1, f(\overline{x}_i) < 0$

$\overline{x}_i$ stands for dataset, which is a collection of real numbers a as attribute, and k is the number of attributes in the data.

$$\overline{x}_i = (a_1\ a_2 \ldots a_k)$$

SVM method will form a support vector for each class and based on data that is closest to the separating hyperplane. When determining confidence, the support vector will assist in classifying. If data is located in between the support vector, then the data is classified with lower confidence than one in below or above the support vector. Figure 2 shows the representation of support vector illustrated by the dotted line.

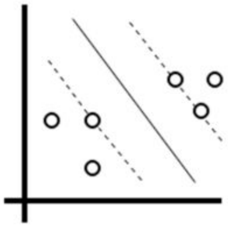


Figure 2. Support vector representation

The model testing process is divided into four steps; testing for message detection in grayscale images, testing for message length estimation in grayscale images, testing for message detection in color images, and testing for message length estimation in color images. The accuracy of message detection and message length estimation is calculated with following equation.

Accuracy = number of correct classified images/number of images in testset (2)

Empirical transition matrices of Markov chain are calculated along horizontal, vertical and diagonal directions and serve as features for steganalysis. For feature classification, the SVM with both linear and non-linear kernels are used as classifier. The non-linear SVM performs much better than linear SVM for proposed higher-dimensional features.

SVM method is strongly correlated to neural networks. In fact, Support Vector Machine (SVM) methods have a close relation to traditional N-layer perceptron neural networks (Hernandez et al, 2008).

7.2.3 Neural Network Classification

The most important problem in a neural network is that convergence is not fast. Practically, this is the most important restriction of neural network applications, because data hiding method is not a linear method, if we only employ linear classification technique to categorize images. The neural network has an admirable facility to simulate any nonlinear correlation. Therefore, it has been used to categorize images. Neural network draws on three levels: input level, hidden level and output level (Liu et al, 2003). Figure 1 shows a simple neural net, more specifically, a Threshold Logic Unit (TLU):

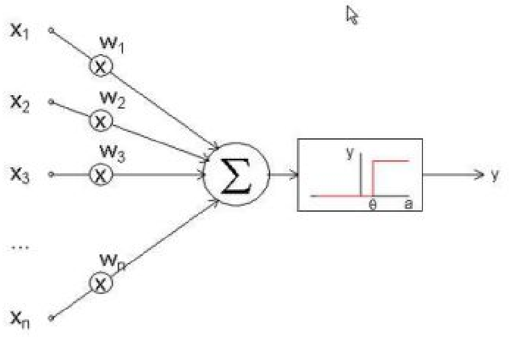


Figure 1. Threshold Logic Unit

8. Applications

As part of the applications, we implemented an example of **Image Steganography** and **Statistical Steganalysis**.

For **Image Steganography** we used an existing R library called **stegosaur** [26] and for **Image Statistical Steganalysis** we considered to reproduce as much as possible the work done by [27] and include some additional **Data Mining Techniques**: SVM Linear and Neural Networks Classification [28].

8.1 Image Steganography (stegasaur)

stegasaur is an R library that implements Steganography on different type of images based on **Least Significant Bit (LSB)** replacement embedding algorithm. The encoding/decoding functions could be used with different type of objects (we test it using simple text, R objects) and shows to work well.

In the following example we embedded a simple text ("This is a test of Steganography using K library stegasaur LSB") into the Cover Image and is not possible to identify any visual difference:

Original Image
Canada

New Image
Canada

The file size change between the Source (Cover) and Encoded (Stego) image for the different test was the following:

Object Type	Object Size	File Size (Cover)	File Size (Stego)
Simple Text	152	11351	11537
R Object (Data Frame)	1952	11351	12609

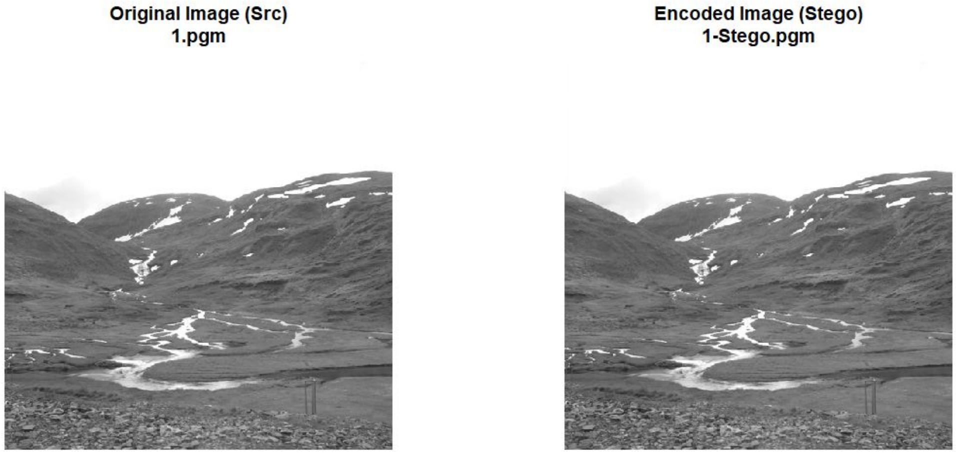
8.2 Statistical Data Mining Image Steganalysis

We implemented three different Data Mining Image Steganalysis Techniques:

- Clustering based on K-means
- Support Vector Machine (SVM) Classification (Using Gaussian and Linear Kernel)
- Neural Networks (NN) Classification

8.2.1 Algorithm

- Find a dataset (images) that will be used as a cover images. We used a typical steganalysis dataset related with the contest named **Break our Steganalysis System (BOSS)** [29]. We used the version named **BOSSbase v1.01** [30]. The dataset include 1000 images in **Netpbm Grayscale Image Format (PGM)** named ([1-1000].pgm). We used 200 images of a typical database used for steganalysis
- Create the **Stego** images using a Steganography Algorithm. We used the **stegosaur** library. For each image we will encode randomly a different object (a text or a R object) and save the final image as Stego. The following is an example:



Original Image (Src)
1.pgm

Encoded Image (Stego)
1-Stego.pgm

- Split the dataset into training and test (2/3 + 1/3), we follow the idea in [27] to include in the same set the Cover and related Stego Image.
- Get the images features. We used the **Subtractive Pixel Adjacency Model SPAM features** defined in [2] (include a total of 686 variables), to get that we used a Matlab code provided [31]
- Do Classification / Clustering. As was mentioned earlier, we try different Data Mining Methods (not only the ones in the reference work):
  - Clustering Using K-means

  - Using Data Reduction (PCA) with the most important 60 variables
  - Without Data Reduction
- Classification using Support Vector Machines (SVM):
  - Using a Gaussian Kernel with the same Hyperparameters in the reference work
  - Using a Linear Kernel
- Classification using Neural Networks

8.2.2 Main Findings

Method	Prediction Accuracy in Training Set	Prediction Accuracy in Test Set	RMSE Test Set	Time Elapsed (Seconds)
Kmeans Clustering PCA	0.5000	0.5000	0.7071	21
Kmeans Clustering	0.5000	0.5000	0.7071	201
SVM Gaussian Kernel	0.9361	0.6642	0.5795	1175
SVM Linear Kernel	0.9887	0.6866	0.5599	180
Neural Networks	0.5446	0.6940	0.5531	227

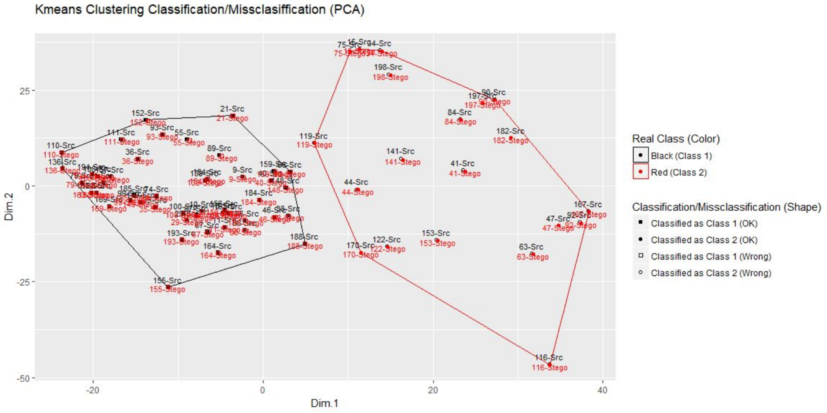
The best results were using **Neural Network (accuracy of 69% in Test Set and RMSE 0.5531). The SVM with Linear and Gaussian Kernels** provide good results but the time elapsed for the **Gaussian Kernel** make it not useful.

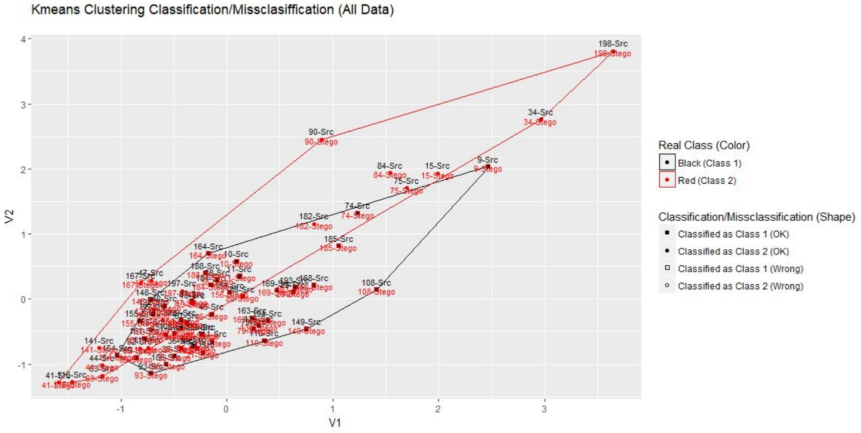
With only 21 seconds, the Kmeans Clustering Using PCA (60 variables) shows to be a good option if the idea is make a first round classification to use later a more advanced technique. Because the methods used include a higher random degree is clear that the results could be different (especially in the case of Neural Networks).

In the following sections the most important information for each scenario is showed.

8.2.3 Clustering Using K-means

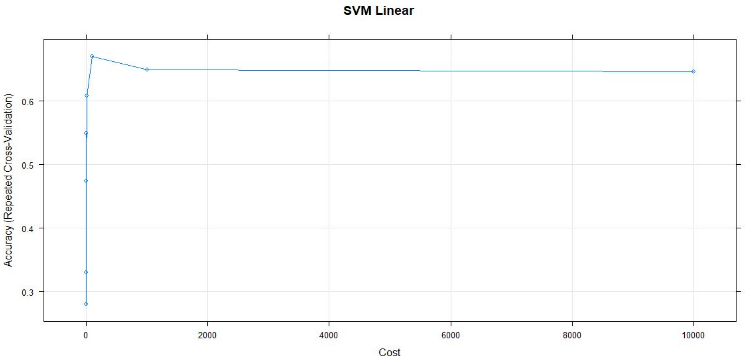
The following pictures shows the Classifications/Miss classifications results using PCA and without it:

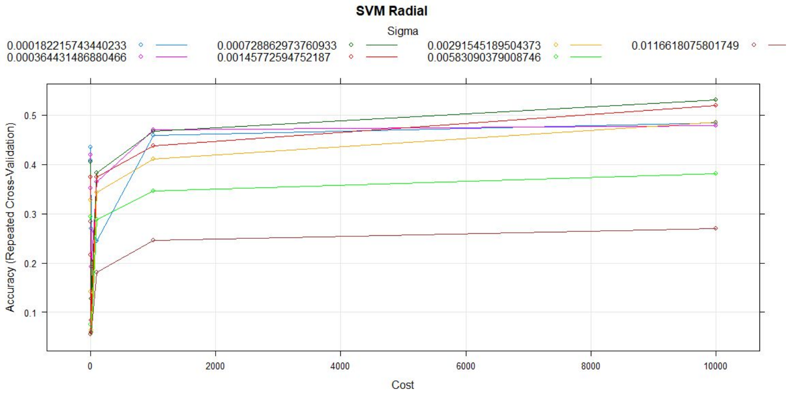




8.2.4 Classification Using Support Vector Machine (SVM)

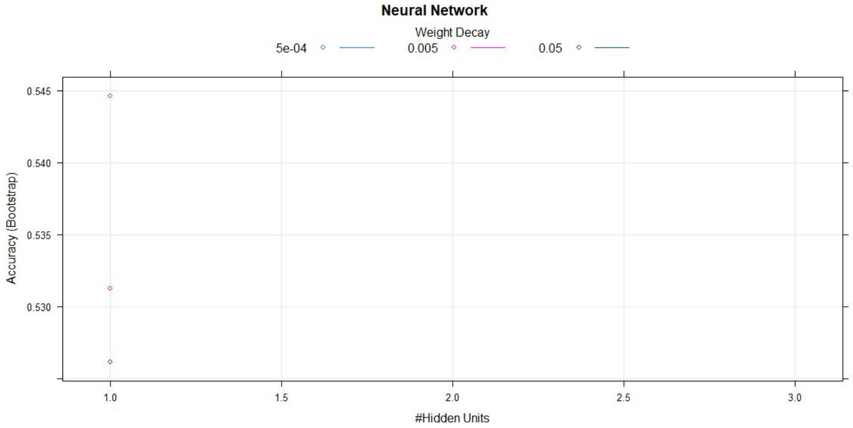
The following pictures shows the Accuracy Results using different parameters:





8.2.5 Classification Using Neural Networks (NN)

The following picture shows the Accuracy Results using different parameters:



9. References

[1] U. Budia, D. Kundur and T. Zourntos, “Digital Video Steganalysis Exploiting Statistical Visibility in the Temporal Domain,” *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 4, pp. 502 – 516, December 2006.

[2]J. S. Jainsky, D. Kundur and D. R. Halverson, “Towards Digital Video Steganalysis using Asymptotic Memoryless Detection,” Proceedings of the 9th International Workshop on Multimedia and Security, pp. 161 – 168, Dallas, TX, USA, 2007.

[3] J. Fridrich and M. Long, “Steganalysis of LSB Encoding in Color Images,” Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), vol. 3, pp. 1279 – 1282, New York, NY, USA, July – August 2000.

[4] A. Westfeld, “F5 – A Steganographic Algorithm,” Lecture Notes in Computer Science, vol. 2137, pp. 289 – 302, January 2001.

[5] Outguess – Universal Steganography: http://www.outguess.org

[6] W. Zeng, H. Ai and R. Hu, “A Novel Steganalysis Algorithm of Phase Coding in Audio Signal,” Proceedings of the 6th International Conference on Advanced Language Processing and Web Information Technology, pp. 261 – 264, August 2007.

[7] W. Zeng, H. Ai and R. Hu, “An Algorithm of Echo Steganalysis based on Power Cepstrum and Pattern Classification,” Proceedings of the International Conference on Information and Automation, pp. 1667 – 1670, June 2008.

[8] Paraskevas and E. Chilton, “Combination of Magnitude and Phase Statistical Features for Audio Classification,” Acoustical Research Letters Online, Acoustical Society of America, vol. 5, no. 3, pp. 111 – 117, July 2004.

[9] M. K. Johnson, S. Lyu, H. Farid, “Steganalysis of Recorded Speech,” Proceedings of Conference on Security, Steganography and Watermarking of Multimedia, Contents VII, vol. 5681, SPIE, pp. 664– 672, May 2005.

[10] H. Ozer, I. Avcibas, B. Sankur and N. D. Memon, “Steganalysis of Audio based on Audio Quality Metrics,” Proceedings of the Conference on Security, Steganography and Watermarking of Multimedia, Contents V, vol. 5020, SPIE, pp. 55 – 66, January 2003.

[11] A.C. Rencher, Methods of Multivariate Data Analysis, 2nd Edition, John Wiley, New York, NY, March 2002.

[12] P. Pudil, J. Novovicova and J. Kittler, “Floating Search Methods in Feature Selection,” Pattern Recognition Letters, vol. 15, no. 11, pp. 1119 – 1125, November 1994.

[13] P. Huttenlocher, G. A. Klanderman and W. J. Rucklidge, “Comparing Images using Hausdorff Distance,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 9, pp. 850– 863, September 1993.

[14] T. Holotyak, J. Fridrich and S. Voloshynovskiy, “Blind Statistical Steganalysis of Additive Steganography using Wavelet Higher Order Statistics,” Lecture Notes in Computer Science, vol. 3677, pp. 273 – 274, September 2005.

[15] I.Avcibas, “Audio Steganalysis with Content-independent Distortion Measures,” IEEE Signal Processing Letters, vol. 13, no. 2, pp. 92 – 95, February 2006.

[16] Q. Liu, A. H. Sung and M. Qiao, “Novel Stream Mining for Audio Steganalysis,” Proceedings of the 17th ACM International Conference on Multimedia, pp. 95 – 104, Beijing, China, October 2009.

[17] C. Kraetzer and J. Dittmann, “Pros and Cons of Mel cepstrum based Audio Steganalysis using SVM Classification,” Lecture Notes in Computer Science, vol. 4567, pp. 359 – 377, January 2008.

[18] U. Budia, D. Kundur and T. Zourntos, “Digital Video Steganalysis Exploiting Statistical Visibility in the Temporal Domain,” *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 4, pp. 502 – 516, December 2006.

[19] I. Cox, J. Kilian, F. Leighton and T. Shamoon, “Secure Spread Spectrum Watermarking for Multimedia,” IEEE Transactions on Image Processing, vol. 6, no. 12, pp. 1673 – 1687, December 1997.

[20] J. S. Jainsky, D. Kundur and D. R. Halverson, “Towards Digital Video Steganalysis using Asymptotic Memoryless Detection,” *Proceedings of the 9th International Workshop on Multimedia and Security*, pp. 161 – 168, Dallas, TX, USA, 2007.

[21]Y. Su, C. Zhang, L. Wang and C. Zhang, “A New Video Steganalysis based on Mode Detection,” Proceedings of the International Conference on Audio, Language and Image Processing, pp. 1507– 1510, Shanghai, China, July 2008.

[22]V. Pankajakshan and A. T. S. Ho, “Improving Video Steganalysis using Temporal Correlation,” Proceedings of the 3rd International Conference on Intelligent Information Hiding and Multimedia Signal Processing, vol. 1, pp. 287 – 290, November 2007.

[23] K. Kancherla and S. Mukkamala, “Video Steganalysis using Spatial and Temporal Redundancies,” Proceedings of International Conference on High Performance Computing and Simulation, pp. 200–207, June 2009.

[24] C. Zhang, Y. Su and C. Zhang, “Video Steganalysis based on Aliasing Detection,” Electronic Letters, vol. 44, no. 13, pp. 801 – 803, June 2008.

[25] B. Liu, F. Liu and P. Wang, “Inter-frame Correlation based Compression Video Steganalysis,” Proceedings of the Congress on Image and Signal Processing, vol. 3, pp. 42 – 46, May 2008

[26] stegasaur, https://github.com/richfitz/stegasaur

[27] T. Pevny, P. Bas, and J. Fridrich, Steganalysis by Subtractive Pixel Adjacency Matrix IEEE Trans. on Info. Forensics and Security, vol. 5(2), pp. 215–224, 2010.

[28] Mohammadi, F.G., Abadeh, M.S.: A survey of data mining techniques for Steganalysis. Recent Advances in Steganography, pp. 1–25 (2012)

[29] Break Our Steganograpghy System BOSS, http://agents.fel.cvut.cz/boss/index.php?mode=VIEW&tmpl=materials

[30] BOSS base 1.01, http://dde.binghamton.edu/download/ImageDB/BOSSbase_1.01.zip

[31] SPAM 686 features, http://dde.binghamton.edu/download/feature_extractors/download/spam686.m

[32] S. Bhattacharyya, I. Banerjee, G. Sanyal, A Survey of Steganography and Steganalysis Technique in Image, Text, Audio and Video as Cover Carrier PDF

[33] S. Bhattacharyya, I. Banerjee, G. Sanyal, AUDIO STEGANALYSIS OF LSB AUDIO USING MOMENTS AND MULTIPLE REGRESSION MODEL PDF

[34]Meghanathan, N., & Nayak, L. (2010, January 1). *STEGANALYSIS ALGORITHMS FOR DETECTING THE HIDDEN INFORMATION IN IMAGE, AUDIO AND VIDEO COVER MEDIA*[PDF]. Jackson: International Journal of Network Security & Its Application (IJNSA).

[35]Ortega, D. R. (n.d.). *Statistical Steganalysis 1*[PDF]. Miami: Florida International University.

[36]Ker, A., Bas, P., Böhme, R., Pevný, T., Craver, S., Filler, T., & Fridrich, J. (2013, June). *Moving Steganography and Steganalysis from the Laboratory into the Real World*[PDF]. Montpellier, France: ACM IH-MMSEC.