SPIM
Thèse de Doctorat

école doctorale sciences pour l'ingénieur et microtechniques
UNIVERSITÉ DE FRANCHE-COMTÉ

N° 2 1 3 1 1 1 7 5

Security Analysis of Steganalyzers

By

Yousra Ahmed Fadil

A Dissertation Submitted to the

University Bourgogne Franche-Comté

in Partial Fulfillment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in Computer Science

Publicly supported on
May 9, 2017

Dissertation Committee:

Prof Pierre Spiteri	University of Toulouse	Reviewer
Prof Guillaume Bonfante	University of Lorraine	Reviewer
Prof Sylvain Contassot-Viver	University of Lorraine	Examiner
Prof Raphaël Couturier	University of Franche-Comté	Examiner
Prof Christophe Guyeux	University of Franche-Comté	Supervisor
Asst Prof Jean-Fançois Couchot	University of Franche-Comté	Co-supervisor

Contents

Table of Contents 3

List of Figures 6

List of Tables 6

List of Algorithms 6

List of Abbreviations 7

Dedication 9

Acknowledgements 11

Introduction 13

1. General Introduction 13
2. Motivation of the Dissertation 14
3. Main Contributions of this Dissertation 14
4. Dissertation Outline 15

I Scientific Background 17

1 Scientific background 19

1.1 Introduction 19

1.2 Image processing and description 20

1.2.1 Image preprocessing 20

1.2.2 Image enhancement 20

1.2.3 Image filtering 21

1.2.3.1 Presentation 21

1.2.3.2 Type of filters 22

1.2.4 Edge detection 23

1.2.5 Edge and Smooth area 24

1.2.5.1 Presentation 24
1.2.5.2 Techniques of Edge detection 25
1.2.6 Threshold 30
1.3 Steganography techniques 30
1.3.1 General presentation 30
1.3.2 Different types of steganography 33
1.3.3 Performance of a steganographic scheme 34
1.3.4 Steganography applications 34
1.3.5 Images steganography 34
1.4 Steganalysis techniques 38
1.4.1 Classification of image steganalysis 39
1.4.2 Methods of steganalysis technique 39
1.5 Related work 40
1.5.1 Features 40
1.5.2 Steganography algorithm 41
1.5.3 Steganalysis result 42
1.6 Conclusion 44

II Contributions 45

2 Performance Study of Steganalysis Techniques 47
2.1 Introduction 47
2.2 Experimental Setup 47
2.2.1 Image databases 48
2.2.2 Working procedure 48
2.3 Results Evaluation 49
2.3.1 Detection evolution when modifying payloads with nsF5 49
2.3.2 Classification sensibility to the steganography method 50
2.3.3 Modification on training and testing sets 54
2.4 Conclusion 55

3 Steganalyzer performances in operational contexts 57
3.1 Introduction 57
3.2 Training and testing stages use not the same steganographier 58
3.3 Trying to improve steganalyzer score by mixing learning steganographiers . 60

3.4 Uncertainty effects regarding payload 63
3.5 Operational contexts 64
3.6 Conclusion 67

4 A Second Order Derivatives based Approach for Steganography 69
4.1 Introduction 69
4.2 Derivatives in an Image 70
4.2.1 Hessian Matrix 70
4.2.2 Classical Gradient Image Approaches 71
4.2.3 Hessian Matrices induced by Gradient Image Approaches 71
4.3 Second Order Kernels for Accurate Level Curves 73
4.4 Polynomial Interpolation of Images for Hessian Matrix Computation 75
4.5 Distortion Cost 76
4.6 Experiments 77
4.6.1 Choice of parameters 78
4.6.2 Security Evaluation 78
4.7 Threshold choice 81
4.8 Conclusion 86

III Conclusion 87

5 Conclusion and Perspectives 89
5.1 Conclusion 89
5.2 Perspectives 90

Bibliographie 98

List of Figures

1.1 Types of noise . . . 22
1.2 Image Convolution . . . 23
1.3 Result of edge detection with Sobel filter . . . 24
1.4 Edge types . . . 25
1.5 Edges detection thanks to Sobel kernels . . . 26
1.6 Robert kernels . . . 27
1.7 Prewitt_kernel vertically and horizontally . . . 28
1.8 First and second edge derivatives to detect edges . . . 29
1.9 Laplacian of Gaussian . . . 30
1.10 The prisoners problem . . . 31
1.11 Steganography system . . . 32
1.12 Secret steganography . . . 33
1.13 Type of steganography . . . 34
1.14 LSB steganography . . . 35
1.15 Pseudorandom steganography . . . 36
1.16 Wavelet decomposition . . . 37
1.17 Steganalysis classifier . . . 43
1.18 Types of ROC curve . . . 44

2.1 ROC curves on different payloads. . . . 49
2.2 ROC curve when learning has been realized using nsF5 and J-UNIWARD. 51
2.3 ROC curve when using different methods of feature extraction CC-PEV, CC-JRM for nsF5 and J-UNIWARD. . . . 53
2.4 ROC curve according to the type of train and test sets. . . . 55

3.1 The steganographier used during the training stage is not the good one. . . 59
3.2 Mixing various steganographiers in the learning stage. . . . 62
3.3 Differences between host content with nsF5 when payload is respectively equal to 0.1 or 0.005. . . . 63
3.4 Train with nsF5 with a 0.1 payload and test with J-UNIWARD with a 0.005 payload . . . 65

4.1 Embedding changes instance with payload $\alpha = 0.4$ 77

4.2 The filter result 81

4.3 Threshold results 82

4.4 Result of the threshold experiment 85

ABBREVIATIONS

LoG	Laplacian of Gaussian
WOW	Wavelet Obtained Weights.
EA	Edge-Adaptive
HUGO	Highly Undetectable steGO
STCs	Syndrome-Trellis Codes
UNIWARD	UNIversal WAvelet Relative Distortion
ROC	Receiver Operating Characteristic
AUC	Area Under a Curve
LSB	Least Significant Bit
SVM	Support Vector Machine
NN	Neural Network
FLD	Fisher Linear Discriminant
SPAM	Subtractive Pixel Adjacency Model
JRM	JPEG Rich Model
CC-JRM	Cartesian Calibrated of JPEG Rich Model
KL	Kullback-Leibler

DEDICATION

∗∗ This thesis is dedicated to my amazing sisters: Ban, Susan, and Rasha who support and encourage me during all the challenges of life.

∗∗ I also dedicate this dissertation to my family who have helped me throughout the process. I will always appreciate all they have done for me.

∗∗ I want to thank my friends who led me to understand some of the most subtle challenges to our ability to thrive and they always believing in me.

ACKNOWLEDGEMENTS

The completion of my dissertation and subsequent Ph.D. has been a long journey. There are a number of people without whom this thesis might not have been written and I am grateful to them. Firstly, I would like to thank my supervisors: Prof.Dr.Christophe Guyeux and Assoc.Prof.Dr.Jean François Couchot for their patience, motiviation. Their guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my Ph.D study.
A special appreciation and thanks to Prof.Dr. Raphaël Coutourier for the stimulating discussions. Your advice on both research as well as on my career have been priceless.

I would also like to give a special thanks to Prof.Dr.Pierre Spiteri, Prof.Dr. Guillaume Bonfante and Prof.Dr.Sylvain Contassot-Vivier for accepting to review my for accepting to participate in my dissertation committee and for giving me their remarks.

My appreciation and thanks go to the members of the team AND for the warm and friendly relation that they provide to me over the three years ago. So special thanks to Mourad Hakem, Abdallah Makhoul, Stéphane Domas, David Laiymani, Gilles Perrot, Jean-Claude Charr, Michel Salomon, Karine Deschinkel and Ahmed Mostefaoui.
I would like to express my great thanks to my two dearest friends Suha K.Shihab and Suha Al-Nassar, thank you for your encouragement, support and most of all your love. You both kept things light and shining.

I would like to express my gratitude to my friends that I lived with them beautiful moments in France: Sylvian Mae Baile, Zedjiga Aidrous, Zeinab Fawaz, Nesrine Khernane, Ke Du, Ahmed Fanfakh, Carol Habib, Christian Salim, Pierre Saenger, Amor Lamar, Do Huu Quan, and Ali Alsaadi.
I hope I did not forget anyone, really, there are no words that can express my gratitude and appreciation for all the persons which stood with me by a small word of encouragement or by a particular effort.

Introduction

General Introduction

The continuing evolution of Internet brings with it modern ways of communication and transformation of different aspects of the life. Today the Internet gives a large space for exchanging the information and is a perfect carrier of information. This information may be in the form of images, videos, and audios. These media may be used to hide some information to transmit through the Internet. In some times the Internet users need to send or receive specific data. The transmitted information must be sometime understood only by the concerned persons. The increasing number of Internet users has naturally led to the unauthorized access to the online information, where unauthorized access of data has crossed the limits and confidential data has been penetrated. Until now, communication of secret data is a sensitive factor in information technology domain that continues to create a difficult challenge with growing levels of sophistication. In order to hide a secret message into a cover object, the cover object must contain an amount of noise or redundant data that is used by the embedding process to conceal secret messages. Images are considered as the most popular cover object that can be used in the domain of information hiding. The image is a set of numbers that represent different light intensities in a view, where the noise and redundant data are used to hide a secret information. One of the important topics that deals with the techniques of hiding information is the steganography field.

The word steganography comes from the Greek words ”stegano” and ”graphein”. The word ”stegano” means covered or concealed while the word ”graphein” means writing. Steganography is the art of covering up hidden messages into innocent like host contents as images. It has known numerous developments this last decade. The embedding process in a steganography technique basically starts by identifying bits that can be modified without creating too much obvious artifacts in cover media. These bits are then interchanged with the bits of the message to hide in a way that keeps media distortion minimal. The rule of steganography systems is the practice of embedding a secret message during communication in a way that no one except the recipient knows the existence of a message. A usual rule of thumb when designing such steganography algorithm is to follow the so-called Kerckhoffs's principle which is usually interpreted in the information hiding context as follows: the secret of the message must not lay on the secret of the algorithm (which must be publicly released), but on a secret key. In other words, the knowledge of the used steganography system should not give any information about the presence of the embedding message. Following this rule, many steganography methods have been proposed the last decades. However, steganography methods necessarily modify statistical properties of images, and these unnatural distortions may be captured.

Tools attempting to separate original contents from media with hidden message, further denoted as stego, are thus designed to work against digital steganography. These tools specifically designed for cover images are called steganalyzers. Recent developments in steganalysis have emphasized the research community to come up with novel steganography methods able to resist against a broad range of attacks on the cover media. Eventually, using all the pixels of an image to embed information is not a good idea because some modifications are obviously more detectable than others. For instance, a slight alteration in a uniformly colored area can be easily detected. The same assumption is not necessarily guaranteed when the alteration is located in the outline of a shape. Up-to-date steganographiers start by computing a map of pixels with low distortion cost. The way to build such a map is a key element in building steganographiers able to face steganalysers. Conversely, steganalysers have to deal with this new generation of information hiding techniques, by investigating which kind of image descriptors are able to signal an unusual, artificial modification of an image content.

Motivation of the Dissertation

The fields of steganography and steganalysis are becoming more and more important in the digital world, where the information is easily exchanged through the Internet. The ongoing requests of new methods to conceal a natural demand to preserve user privacy on the one hand, and conversely to be able to monitor activity of individuals suspected to perpetrate condemnable actions up to terrorism on the other hand, lead to a real need of new studies in the field of both steganography and steganalysis. Indeed, the domains of steganography and steganalysis are like a game between two teams. One of the team tries to find the possible way to embed a secret message in the cover object with the minimum artifacts on the cover object. This desire leads to the detection of the most noisy areas in cover images, to discreetly embed a secret message in it. In the opposite side, steganalysis systems try to detect any flaws in steganography systems, which may lead to a kind of signature in the host content, which can be detected by tools like ensemble classifiers working on extracted features.

Today, many algorithms have appeared in these two fields of information hiding, and a global critical look on the proposals is now possible. In particular, it becomes appropriate to make the competition fairer and more close to the reality life, that is, to investigate what has been done until now in a more operational context. This is what will be done in the first and the second contribution of this thesis. In particular, we will emphasize the fact that, up to now, the game is totally unfair for the steganographier, as the steganalyzer has access to too much information regarding the embedding process. With such an unrealistic access, he or she will be able to set up its artificial intelligence too, in order to achieve an acceptable classification between natural and steganographied content. After having raised such an issue, for the sake of completeness and by taking the advantage from the results to the first and second contribution described hereafter, we will use the knowledge we acquired to propose a new method of steganography that studies the best areas in an image by using the second derivative algorithm.

Main Contributions of this Dissertation

The main contributions of our dissertation can be summarized as follows:

- The first contribution studies the state of the art in the domain of steganalysis systems. The performance of steganalyzers has been investigated according to various parameters, encompassing the choice of the steganography, its payload, and the type of images, both during training and testing stages.

- The second contribution deals with understanding and optimizing parameters of steganalysis. With such learning, we have proposed a kind of universal steganalysis without any knowledge regarding the steganography side. The effects on the classification score of a modification of either parameters or methods between the learning and testing stages have been further evaluated.

- In most existing state of the art approaches, the embedded distortion function is based on image features preservation. Smooth regions or clean edges define image core. Even a small modification in these areas largely modifies image features and is thus easily detectable. These regions are characterized by disturbed level curves. We have presented a new distortion function for steganography that is based on second order derivatives, which are mathematical tools that usually evaluate level curves. This is the main part of the third contribution.

4. Dissertation Outline

The thesis is organized as follows: Chapter 1 presents a scientific background of the state of the art in both steganography and steganalysis fields of research, and the related literature with their schemes. In Chapter 2, steganalysis methods are evaluated according to parameters like payloads, features extraction, and the group of image used during both training and testing stages. In the second contribution, in Chapter 3, the operational context of steganalysis is regarded, according to various steganography methods. The third contribution, in Chapter 4, proposed a method of steganography that depends on the second derivative of the images. Finally, a conclusion and future works are presented in Chapter 5.

Publications

[**1**] Rola AL-SHARIF, Christophe GUYEUX, Yousra Ahmed FADIL, Abdallah MAkHOUL and Ali JABER. On the usefulness of information hiding techniques for wireless sensor networks security. In International Conference on Ad Hoc Networks. Springer International Publishing, p. 51-62, 2014.

[**2**] Bechara AL BOUNA, Jean François COUCHOT, Raphaël COUTURIER, Yousra Ahmed FADIL, and Christophe GUYEUX. Performance Study of Steganalysis Techniques. In IEEE International Conference on Applied Research in Computer Science and Engineering (ICAR), 2015. p. 1-7, 2015.

[**3**] Yousra Ahmed FADIL, Jean-François COUCHOT, Christophe Guyeux and Raphaël COUTURIER. Steganalyzer Performances in Operational Contexts. In IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP). p. 429-432, 2015.

[**4**] Jean-François COUCHOT, Raphaël COUTURIER, Yousra Ahmed FADIL and Christophe GUYEUX. A Second Order Derivatives based Approach for Steganography. In SECRYPT 2016, 13th International Conference on Security and Cryptography.

I

Scientific Background

1

Scientific background

1.1/ Introduction

Steganography is the art of creating invisible communication to provide a secret channel to exchange hidden information. Various steganography methods have recently been proposed focusing on digital media. However, these methods still face many problems related to payloads and classes of applied images. The increasing use of steganography is a result of Internet development leading to the transmission of numerous information using many types of social media like Twitter, Facebook, Instagram, etc. The transformation of the information on the world wide web requires efficient and reliable algorithms to transmit secret information, in any kind of computer files: images, audios, videos...

Images are considered as the most popular cover media to hide information, due to the existence of many redundant bits in the digital representation of an image. For this reason, using image files to transfer a secret information is the model that comes first in mind. Steganography techniques must embed the secret information in regions of a given image that do not lead to a big change in the cover. For instance, embedding in smooth areas causes a large modification in image features and, thus, it is easily detectable. Conversely, concealing a hidden information in noisy areas like edges should be hard to detect, since they make little changes in the image features.

Indeed, hiding information within a given image alter some of its characteristics, and such an alteration may reveal the existence of a secret message. The main issue for a specific steganalysis approach is to detect the presence of a secret information, knowing that this information was hidden by a specific steganography algorithm. The success of a specific steganalysis method is fully dependent on the available information about the steganography algorithm that is used for concealing secret information into the image, while the blind steganalysis system does not depend on the knowledge about the steganography schemes applied to the image.

This chapter, which provides the scientific background of our thesis, explores image processing and description in Section 1.2. The presentation of steganography techniques is provided in Section 1.3, while Section 1.4 describes the steganalysis schemes. Finally, the related work is explained in Section 1.5.

1.2/ Image processing and description

Visual information is the significant type of data perceived and recognized by the human brain. Digital images are the most prevalent and convenient way for transferring such kind of information. The visual information is then processed, manipulated, and interpreted by using the methods of digital image processing [6]. This processing plays an important role in many fields of research, as there is an increasing demand of image processing methods in various application areas like multimedia, security data, biomedical, astronomy, and so on. Most of the time, this processing consists of extracting various information from the considered digital images [95]. Indeed, the latter is a collection of a finite number of components, each one having a specific position and value. These components are referred as picture elements (pixels). Their values represent either the brightness or the color in the image. The type of operations applied during digital processing depends on what is finally expected. The latter encompass convolution, correlation, statistical operation, etc. [34, 77].

1.2.1/ Image preprocessing

Digital image processing manipulates the digital image in order to enhance it or to extract some information from it. Image processing often deals with digital images, but may concern analog images [61]. In general, it includes three main steps, listed bellow.

- **Image acquisition**: digital images are created by sensors inside different kind of light-sensitive cameras. The produced image depends on the type of sensors. In general, the pixel values depend on light intensity in one or several spectral bands. It can depend also on various physical measures, such as depth, absorption or reflectance [104].
- **Image preprocessing**: operations that are applied on raw images produced by the acquisition steps. They operate at low level of abstraction to enhance the visual quality of the images. The preprocessing operation contains image resampling, contrast enhancement, noise removal, and data compression [97]. It provides too methods to extract features and objects from image data. Let us finally notice that these enhancement techniques on images can lead to information loss if they are not used correctly.
- **Output image**: This is the last stage that concretely produces the digital image.

1.2.2/ Image enhancement

The goal of image enhancement is, as its name suggests, to improve the image quality. Image enhancements have been widely used in many fields of image processing, encompassing the following techniques: smooth edges, sharpening certain features of interest, making an image lighter or darker, increasing or decreasing the contrast, etc. Such image enhancements are also achieved by applying many filters in various ways, the main difficulty in such techniques being to provide an objective criterion to measure the enhancements [86, 101].

Image enhancement techniques can be split into two categories.

- **Spatial domain methods**: the pixel values are directly modified [67].
- **Frequency domain methods**: the image is first transformed in a frequency domain by applying a Discrete Fourier Transform (DFT), a Discrete Cosine Transform (DCT) or a Discrete Wavelet Transform (DWT) operation. Then, the enhancement operation is performed on the frequency coefficients. The intensity pixels of the output image are then computed using the invert transformation function applied to the frequency coefficients [50, 83].

1.2.3/ Image filtering

1.2.3.1/ Presentation

Images may have been corrupted during acquisition by intensity variations, contrast modification, illumination changes that may have occurred during the early stages of the acquisition. Therefore, some values may not reflect what were supposed to be embedded in pixels. Various other situations lead to the necessity to apply image filtering, for instance to decrease noise and/or take out important image features [68]. Indeed, noise can be considered as unwanted information in images, as it creates uncomfortable effects as artifacts, erroneous edges, invisible lines, blurred objects, and corners. There are many types of noise, which are listed hereafter:

- **Gaussian (or electronic) noise**: a noise whose probability density function is as follows,

  $$G(y) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(y-\mu)^2}{2\sigma^2}} \tag{1.1}$$

  where μ is the mean value and σ the standard deviation of the noise. The origin of a Gaussian noise in digital images is mainly caused by natural sources such as poor illumination, high temperature, or thermal vibration [13].

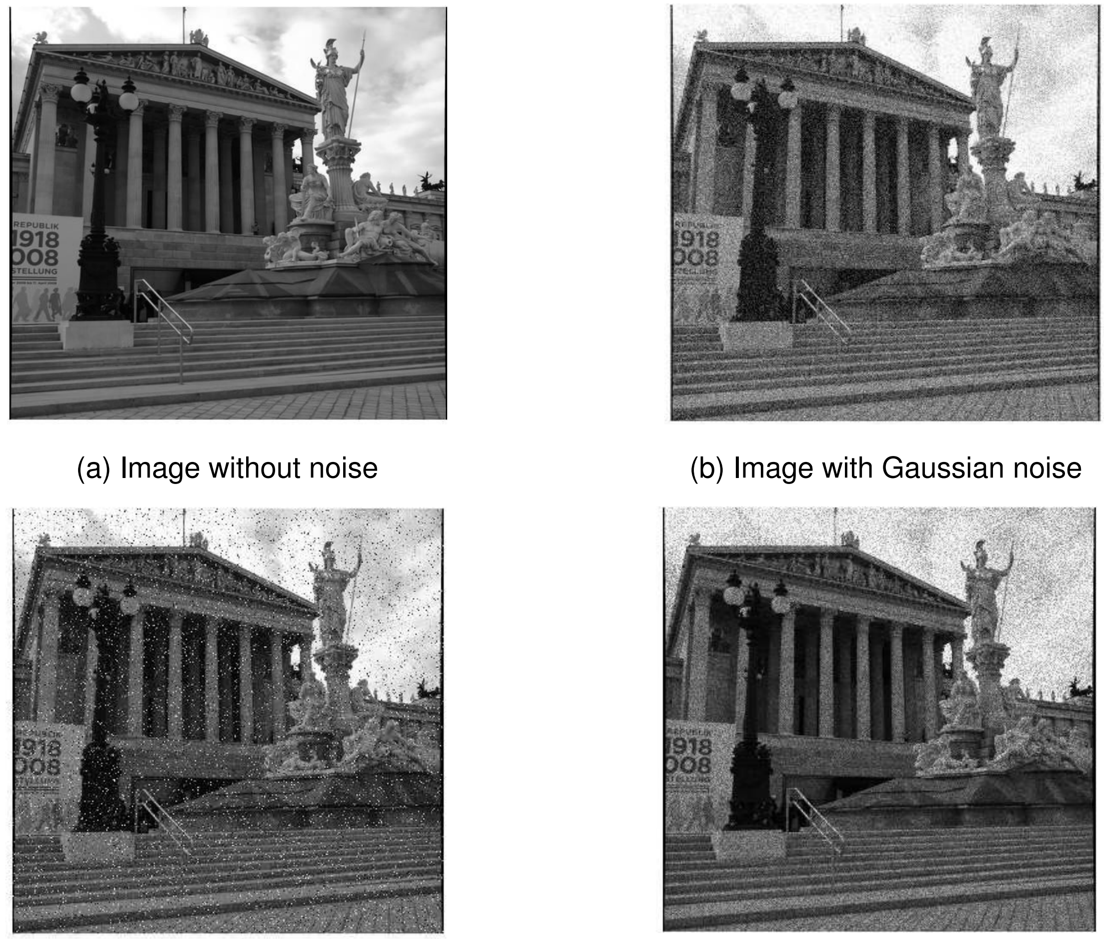
- **Salt and pepper noise**: in this kind of noise, the probability density function can be expressed as follows

  $$G(y) = \begin{cases} P_a, & \text{for } y = a \\ P_b, & \text{for } y = b \\ 0 & \text{otherwise} \end{cases}$$

  where y represents intensity values of pixels in a noisy image, while a and b are saturated values (if $b > a$, intensity b will appear as a light dot on the image and a appears as a dark dot [7]. The salt and pepper noise appears itself as randomly putting white and black pixels in the image, while the usual value for pepper noise a is 0, and for salt noise b, it is 255 [33].

- **Speckle noise**: this is a multiplicative noise. It reduces the visual evaluation in ultrasound imaging. Speckle noise can be modeled as random value multiply by the pixel value [103].

These different types of noise have been illustrated in Figure 1.1.



(a) Image without noise

(b) Image with Gaussian noise

(c) Image with salt and pepper noise

(d) Image with speckle noise

Fig. 1.1: Types of noise

1.2.3.2/ Type of filters

There are many techniques to reduce noise in grayscale and color images. The choice of a filter depends on the nature of the task and the type of the data. There are many factors that have effects on the filter function such as optical artifacts, color accuracy, and noise.

Linear filtering can be used to reduce particular categories of noise that are related to “grain” noise. Filters that can achieve such aim are, for instance, Gaussian filters or averaging. In this kind of filtering, the output pixel is a linear combination of the values in the input neighborhood of the considered pixel [51].

Linear filtering can be performed by using a convolution technique. Convolution is an operation in which each generated pixel is the total weighted sum of neighboring information pixels [90]. This convolution operation is performed at each point in the image, see Figure 1.2. It can be applied on an image to emphasize some useful information like edges. Indeed, many operations in digital image processing use such convolution, like filters for noise reduction.

Non-linear filters have been proposed to avoid the destruction of lines and sharp edges in images caused by linear filters. The median filter is a good example of such nonlinear filter, see, *e.g.*, [100, 87].

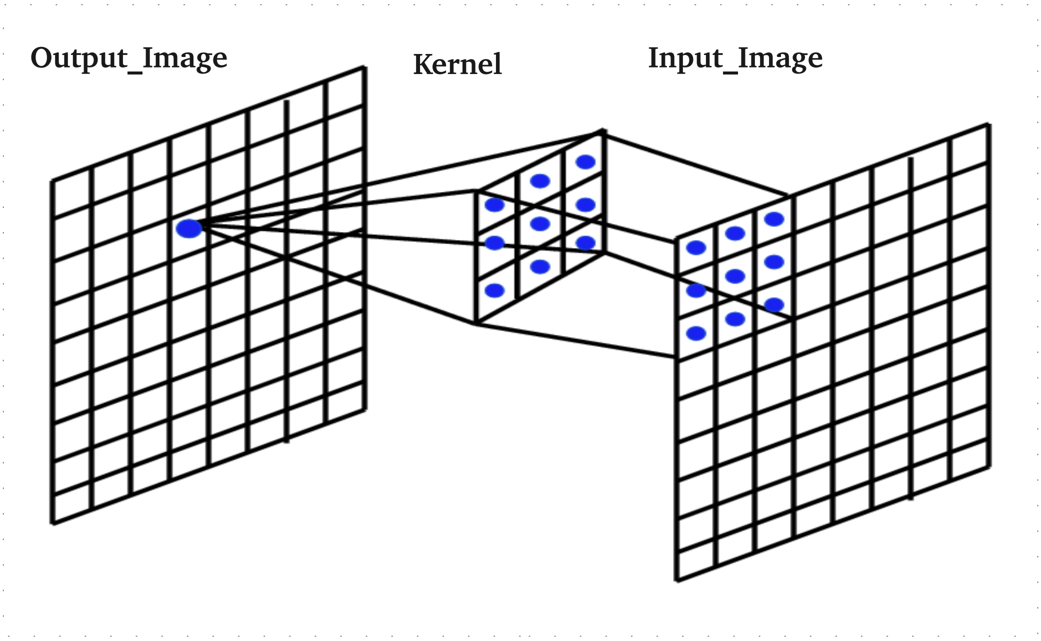


Fig. 1.2: Image Convolution

1.2.4/ Edge detection

Edges in an image are borders or contour at which important changes occur in some physical side of an image. These aspects can be represented such as a change in illumination, in color intensity, and so on. It can also be described as unexpected changes of continuities in an image. These discontinuities are the changes in pixel intensity that describe boundaries of objects in a given image.

Edge detection is a process which aims at finding the presence of these discontinuities and their location. General methods of edge detection include convolving the image with a two dimensions filter, which is structured to be sensible to large gradients in the image while giving values of zero in a clear area. Edge detection is a key element in many image processing techniques such as classification, segmentation, recognition, and object detection [91].

1.2.5/ Edge and Smooth area

1.2.5.1/ Presentation

Usually, an edge is a boundary between objects and background. Indeed, they represent the confines for single objects. Therefore, if the edges can be accurately specified, numerous properties like areas, shapes, and perimeters can be computed. The geometry of detection can be optimized to search for vertical, horizontal, or diagonal edges in an image. The edges detection task may be a difficult exercise if we take into consideration the presence of noise and redundant data in an image, where both the noise and edges have a high frequency for instance [70]. Figure 1.3 represents the edge detection with Sobel filter.

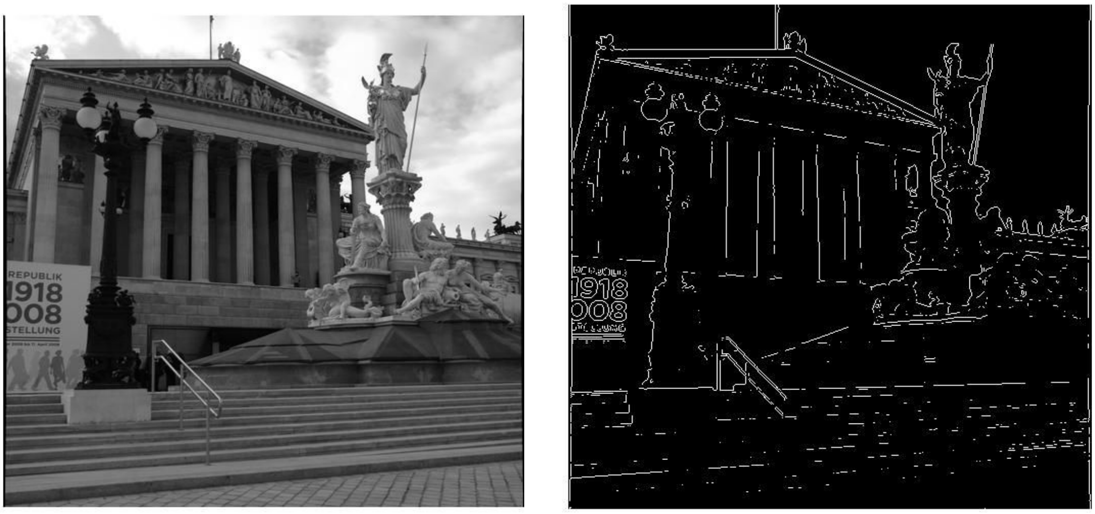


Fig. 1.3: Result of edge detection with Sobel filter

Many edge detection techniques are available. Each one is designed to be sensitive to certain types of edge. This detection technique is fundamentally not based on the pixel values, but is based on the surrounding of each pixel. Edges are described by two important features, the first one being the magnitude (*i.e.*, where is recognized the strength of the edge), while the second feature is the direction, which represents the angle of the edge.

Ideal edges can be classified according to their intensity as follow [89, 85]:

- **Step edge**: this type of edge occurs when the values suddenly change from one value on one side to a different value on the opposite side.
- **Ramp edge, Roof edge**: happens when intensity change happens over a finite distance (*i.e.*, not immediately).
- **Ridge edge**: when the intensity in an image changes instantly but then returns to the initial value in short distance.

Figure 1.4 represents ideal types of edges in an image.

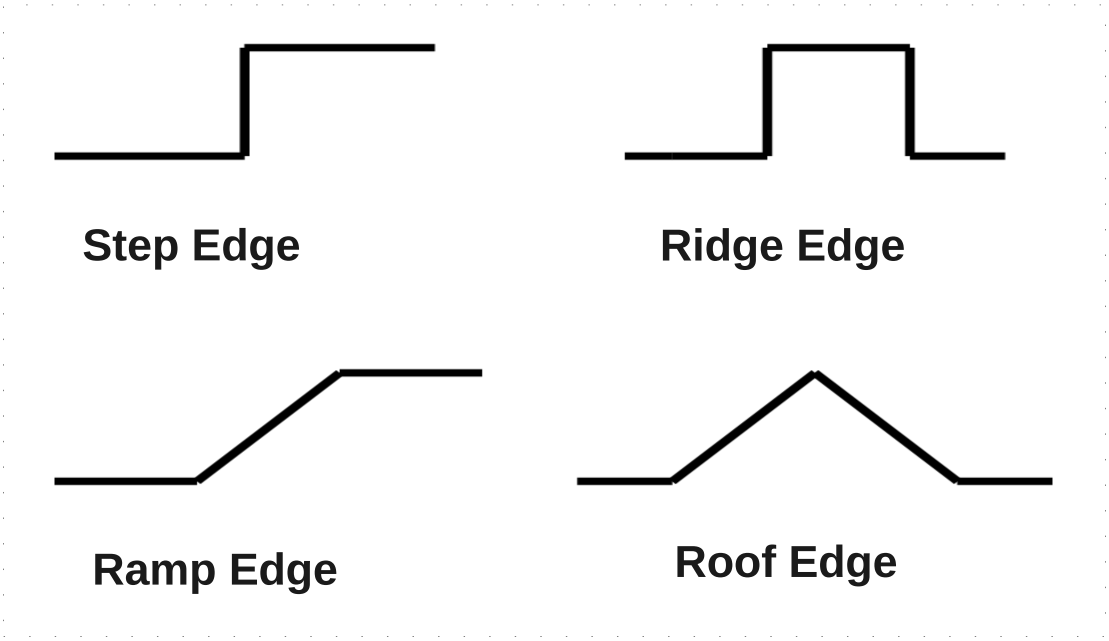


Fig. 1.4: Edge types

1.2.5.2/ Techniques of Edge detection

In general, the following steps are required to detect edges in image processing [88]:

- **Denoising**: It is used to reduce random variation in images caused by various types of noise. Most of the filtering techniques lead to lose some fine edges. The filtering operation can be described as a smoothing filter followed by applying a derivative operation. The smoothing is often performed by a convolution function [107].
- **Enhancement**: In edge detection, it is necessary to consider the variation in intensity in the neighborhood pixels. Enhancement techniques confirm where there is an important change in neighborhood pixels. This is commonly executed by calculating a gradient magnitude to the image. The enhancement technique in image processing helps to increase the detectability of the image details. The main objective of image enhancement techniques is to adjust image characteristics to make it more appropriate for the given task and to improve the clarity of an image for a human visual system [9].
- **Detection**: When the image gradients are computed, some of their values are nonzero, but not all of these points are edges. The method of edge detection should

be able to determine which value represents an edge point. The edge detection processes are a set of mathematical operations to compute high contrast and intensity differences in a digital image [59].

There are mainly two techniques for edge detection, which are based on the examination of the local discontinuity at each pixel.

1. **First Order Derivative Edge Detection**: the edge detector in this technique is based on measuring the intensity gradient, magnitude and orientation at pixels. Gradient $\triangledown F$ can be computed using the following equation:

$$\triangledown F = \begin{bmatrix} \frac{\partial}{\partial x} F \\ \frac{\partial}{\partial y} F \end{bmatrix}.$$

   Both magnitude and orientation can be computed with this formula. We are then left to find maximum and minimum values of this gradient, in order to detect the edges. Many kernels are proposed to find an edge in an image using this principle, some of these kernels are covered here [63]:

   - **Sobel kernel**: the Sobel kernel is a discrete differentiation operator. It is used to compute the gradient of the image intensity function. The Sobel kernel is convoluted with the image in the horizontal and vertical direction [78]. Figure 1.5 represents the result of applying the Sobel kernel.

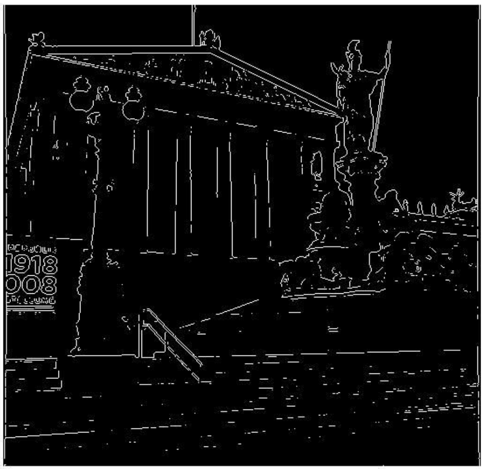


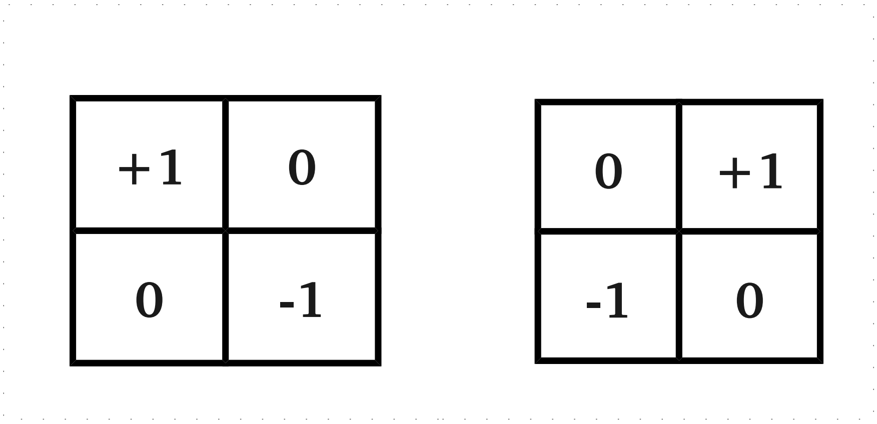
Fig. 1.5: Edges detection thanks to Sobel kernels

   - **Robert kernel**: It computes the 2-D spatial gradient for an image. The Robert kernel focuses on regions with high spatial gradient which in most cases belong to edges. The values of the pixels after applying this kernel represent the estimated absolute magnitude of the spatial gradient of the input image at this point. This kernel consists of a pair of $2x2$ convolution kernels as shown in Figure 1.6. One kernel is simply the other rotated by 90 degrees. This filter is applied to the input image to produce the gradient components. The two component of the gradient can be combined together to find the absolute magnitude of the gradient at each point and the orientation of these gradients [12]. The gradient magnitude is given by:

$$|G| = \sqrt{(Gx)^2 + (Gy)^2}, \tag{1.2}$$

while an approximate magnitude can be computed as follow:

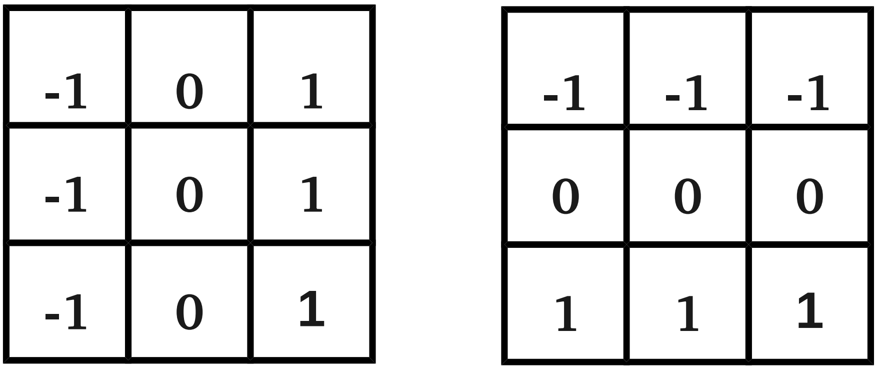
$$|G| = |Gx| + |Gy|. \tag{1.3}$$



$$\begin{array}{|c|c|}\hline +1 & 0 \\ \hline 0 & -1 \\ \hline \end{array} \qquad \begin{array}{|c|c|}\hline 0 & +1 \\ \hline -1 & 0 \\ \hline \end{array}$$

Fig. 1.6: Robert kernels

- **Prewitt kernel**: This kernel is based on the principle of central difference. The image is defined as a signal and the change in signal can be calculated using differentiation. Therefore all the kernels that are used for edges detection are known as derivative kernels. The Prewitt kernel detects horizontal and vertical edges by using a difference between corresponding pixel intensities of an image. It is similar to the Sobel filter [78].



-1	0	1
-1	0	1
-1	0	1

-1	-1	-1
0	0	0
1	1	1

Fig. 1.7: Prewitt_kernel vertically and horizontally

2. **Second Order Derivative Edge Detection**: This operator tries to find peaks in gradient magnitude. Zero crossings of the second derivative are more accurate for detecting the edges. Indeed, the first derivative operators are sensitive to noise, but the second derivative operators will be twice more sensitive [1]. Edge position can be described in the first and the second derivative as follow:

   - The edge represents the local maxima or minima in the first derivative.

   - It represents the zero-crossing in the second derivative.

   Figure 1.8 depicts first and second derivatives of a given signal.

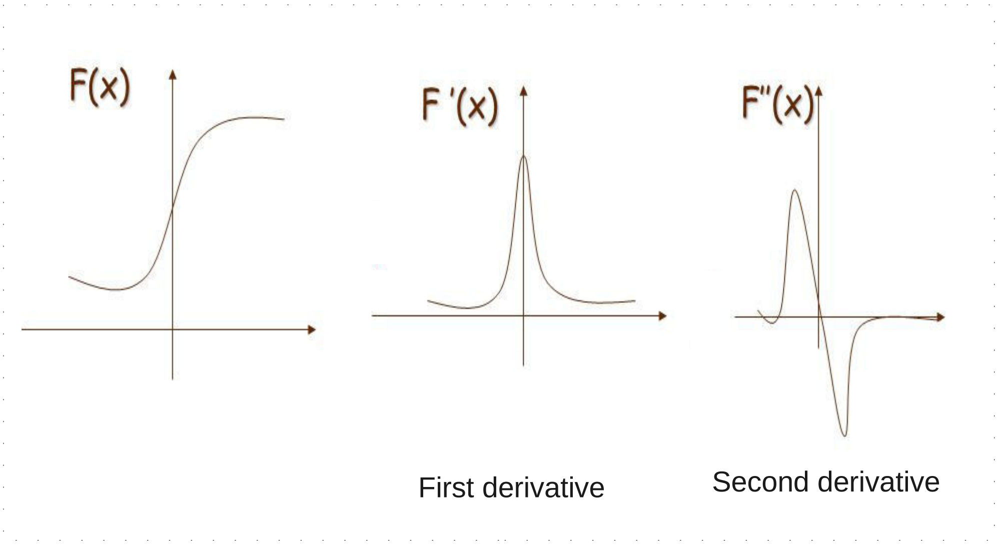


Fig. 1.8: First and second edge derivatives to detect edges

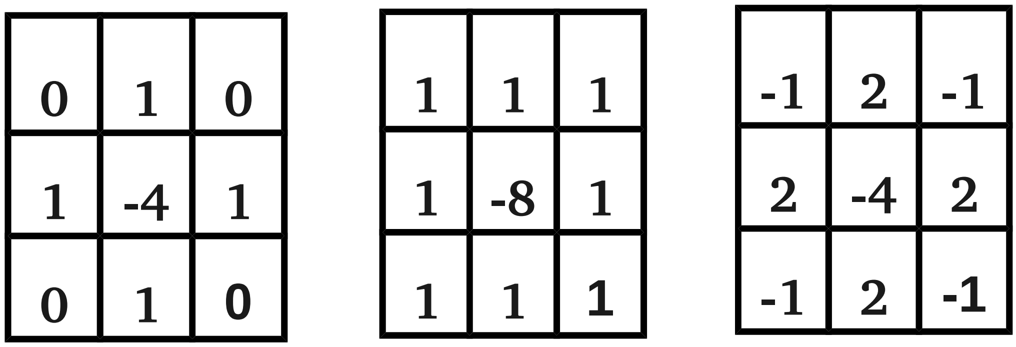
Some masks are used following the second derivative methods to detect the edges. Among them, we can evoke the Laplacian of Gaussian, described hereafter:

- **Laplacian of Gaussian (LoG)**: The Laplacian is a 2-D measure of the second spatial derivative of an image. It searches for zero crossing rate to detect the edges in images. The Laplacian highlights regions of the image that have rapid intensity changes. The image is smoothed with a Gaussian smoothing filter, in order to reduce its sensitivity to noise, and then the Laplacian filter is applied, which is given by:

  $$L(X,Y) = \frac{\partial^2 I}{\partial X^2} + \frac{\partial^2 I}{\partial Y^2}, \tag{1.4}$$

  where $I(X,Y)$ is the pixel intensity function.

The kernel process can be represented as the application of a smoothing filter, followed by a derivative process for countering the sensitivity to noise [66]. This process reduces the high frequency noise components before a differentiation step. Figure 1.9 contains three frequently used kernels.



0	1	0
1	-4	1
0	1	0

1	1	1
1	-8	1
1	1	1

-1	2	-1
2	-4	2
-1	2	-1

Fig. 1.9: Laplacian of Gaussian

1.2.6/ Threshold

All of these detectors use threshold techniques, whose values are determined thanks to experiments. Such techniques can be considered as a segmentation process, as they classify pixels depending on whether they belong to edges or not. Edge detection using threshold is important in many research areas for image processing.

Edge detection may be difficult to achieve in some situations. The success of detectors depends on factors like the presence of objects with similar intensities, noise and lighting conditions, and so on. Some of these problems are solved by choosing a good threshold, but no definitive method has been determined for automatically choosing such relevant values [108].

1.3/ Steganography techniques

1.3.1/ General presentation

Internet technologies have recently become one of the main way for communication and information sharing, and so information and data security has become a major concern. Various measures can be applied to guarantee data and information security during their transmission through the Internet, most of them are based on steganography and cryptography [15, 48].

Steganography encompasses all the methods allowing to conceal information in digital media, with traces of embedding operations as minimal as possible. It is very important during steganographic exchanges that nobody except the sender and the receiver knows that a hidden message exists. The general objective of a steganography systems is thus to not attract unwanted attention. A well known example illustrating that fact is the so-called Prisoners Problem. Alice and Bob are in prison, locked up in separate rooms far away from each other. They want to exchange an escape plan through messages [62] (see Figure 1.10: Alice and Bob represent the prisoners and Wendy is the warden).

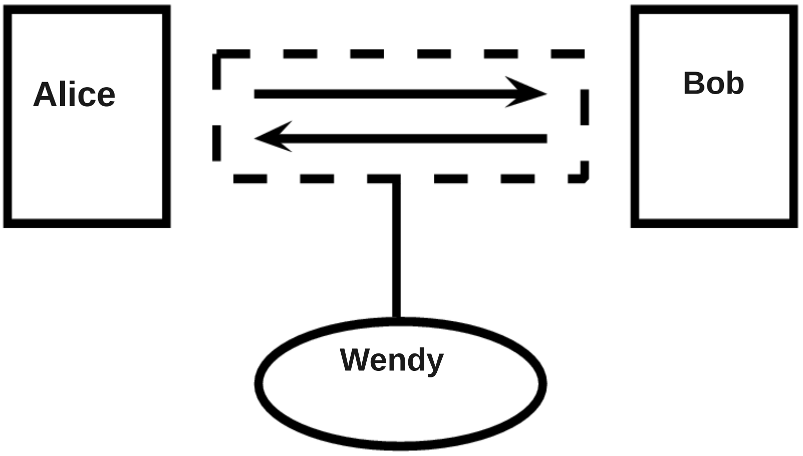


Fig. 1.10: The prisoners problem

Figure 1.11 shows the base diagram for digital steganography technique. It represents two parties, *i.e.*, sender and receiver. They are the two sides of the system, who communicate over a public channel.

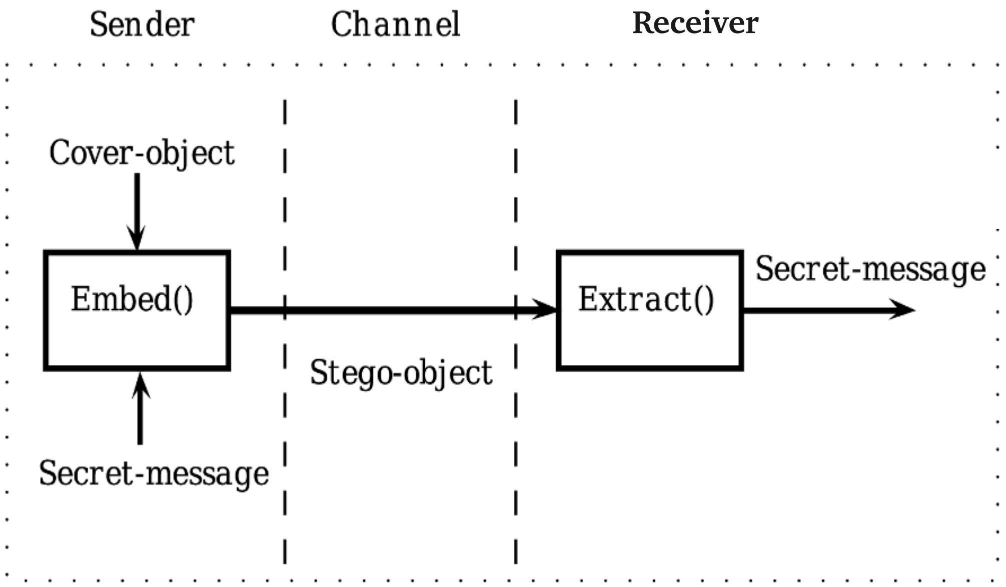


Fig. 1.11: Steganography system

The sender applies an embedding function: $Emb : C \times M \rightarrow S$.

In this equation, C represents the set of cover images as inputs and M refers to the set of secret messages that will be embedded in the cover objects. The stego object S is transferred to the receiver who extracts the secret message M from the stego object S.

The general steganography system can be classified into the following categories [35]:

- **Pure steganography**: is a steganography technique that does not need to modify the secret messages before sending it. For that, no information is required to begin the communication process. The embedding formulation can be written as a mapping:
  Emb: $C \times M \rightarrow S$
  where C is the set of possible covers and M the set of possible messages. The extraction process can be described as:
  Ext: $S \rightarrow M$
  extracting the secret message M from the stego image S.
- **Secret key Steganography**: is a steganography technique in which a key is used to embed the secret message in a cover object. This cover was chosen by the sender. When the receiver knows the secret key, he can obtain the evidence of the embedding message [16]. Figure 1.12 represents a secret steganography system. The embedding function that explains the secret key steganography can be expressed as follow:
  Emb: $C \times M \times K \rightarrow S$
  The sender embeds a message M in an image in the set C by using the set of key

K. The secret message can be extracted from the stego image when the receiver knows the secret key. The extracting process is expressed as follow:
Ext: $S \times K \rightarrow M$

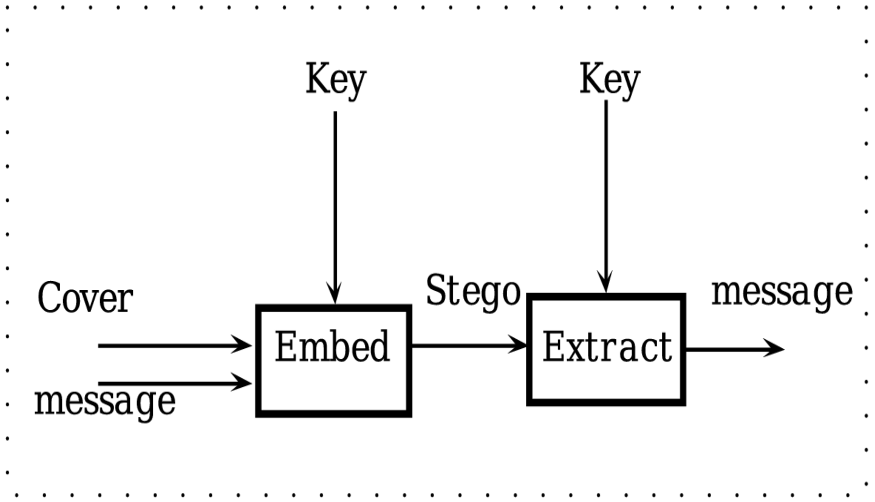


Fig. 1.12: Secret steganography

1.3.2/ Different types of steganography

All digital file formats can be used as a carrier in steganography methods, but digital formats with high redundancy are more convenient to this science domain. Indeed, redundant bits may be easily modified without attracting attention. Figure 1.13 represents general types of steganography, which are described bellow:

- **Text steganography**: Text steganography is achieved by changing either the format of the text or the characters sequence. These two modifications are used for embedding, leading to a cover object that contains the secret message [2]. Various ways have been proposed to embed such information in messages, like using word synonyms, omitting commas, or playing with spelling errors. Most of them degrade the text quality [11, 22]. This is why that this kind of steganography is not really used, which can be explained by the low presence of redundant information in text documents (when compared with images and audio files).

- **Image steganography**: It is considered as the most important steganography method nowadays. The secret messages can be embedded in images without inducing a big modification in the cover object [8, 45]. To do so, it is important to hide the message in noisy areas, to keep visible properties of the image. For instance, areas that contain many color variations and texture are more suitable to hide secret messages [64, 21, 11].

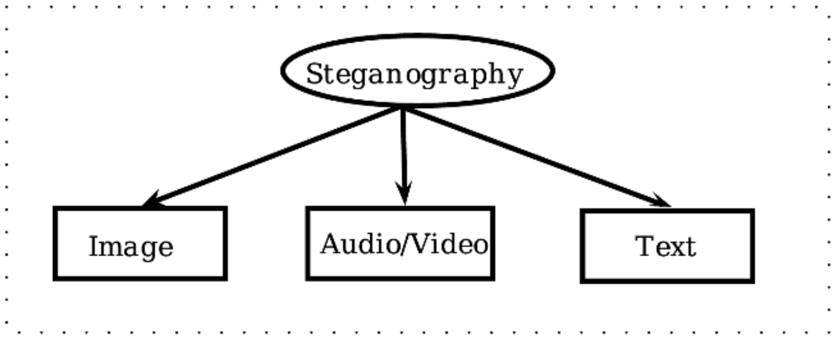


Fig. 1.13: Type of steganography

- **Video/Audio steganography**: steganography algorithm in video domains focuses on repeated frames, in which it is possible to conceal secret messages [32].

1.3.3/ Performance of a steganographic scheme

As any steganography system embeds a secret message in a cover object, its performance can be evaluated by the following criteria [8]:

- **Steganography capacity**: which represents the maximum length of the secret message that can be hidden in the cover object. The capacity depends on the embedding function and on the cover object properties [60].
- **Steganography security**: The objective of steganography is to merely embed a secret message in the cover media. The security of steganography is achieved by the difficulty to detect a secret message, which can be compared with the difficulty to read a message content in cryptography systems [36].

1.3.4/ Steganography applications

Steganography techniques have many applications, like copyright protection, identity cards in bank systems (where individual data are concealed in their photos), TV broadcasting, medical imaging systems, etc. Such methods can be used in secure military communications too, to transfer any sensitive information by using innocent like cover images or videos. Other applications in the information security field can be found, for instance, in [47].

1.3.5/ Images steganography

Today, most of the steganography systems use images to conceal secret messages, taking advantage from the limited human visual perception. Furthermore, noise in images gives space to embed secret data. There are many ways for achieving such information hiding inside images, such as steganography by cover selection, where sender and receiver exchange secret messages thanks to image contents (for example, the presence of arms in the image can indicate that there will be an attack). But, in this situation, it is hard to define a proper theoretical framework for evaluating the security of the method. This is why steganography by cover modification has been more studied in recent years, as it allows security evaluation in practice. The number of messages that can be transmitted between the sender and the receiver depends on both the properties of the cover and the steganography algorithm. For instance, in JPEG images, the sender can conceal one bit per non-zero DCT coefficient [92, 3], while in raw images, it can embed one bit per pixel if we consider LSB steganography (see bellow).

The cover modification based steganography techniques are classified into:

1. **Substitution techniques**: the substitution methods embed the secret message by replacing not important parts of the cover object with secret information. The secret message can be extracted by the receiver if he know the exact position where the hidden information has been embedded [4]. There are several approaches in this technique:

   - **Least Significant Bit (LSB) technique**: in this method, the sender chooses the least significant bit of some or all of the bytes of a cover image object to conceal the secret message, see Figure 1.14. The least significant bits are used to rebuild the secret information. This method, which embeds the secret message with little impacts to the cover, is characterized by its simplicity, but it is not secure [20, 72].

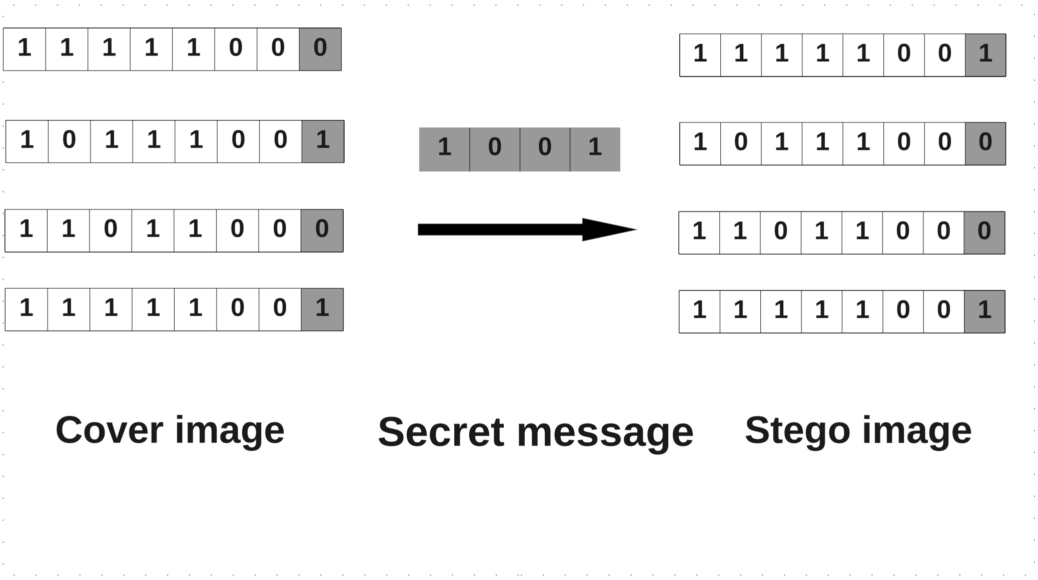


Fig. 1.14: LSB steganography

- **Pseudorandom Permutation technique**: instead of focusing on LSBs, we can use a pseudorandom number generator that randomly selects pixels and bits inside them, that will receive the secret information [84], as depicted in Figure 1.15. Bits of the secret message will then be distributed randomly over the whole cover object [42]. To recover the hidden message at the receiver side, the same generator is seeded with the same initial value, and thus the same random locations are indicated by the generator, allowing a message reconstruction.

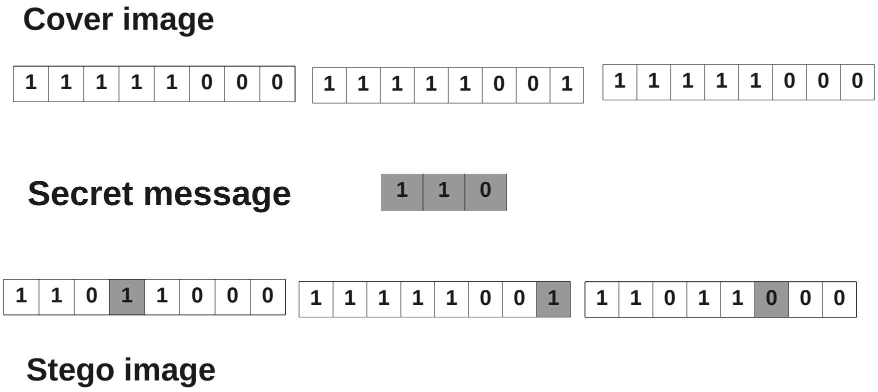


Fig. 1.15: Pseudorandom steganography

2. **Distortion technique**: today, the most popular steganography approaches embed a secret message according to the minimization of a distortion function. The design of this distortion function that indicates where to embed the secret message (locations that minimize distortion effects) is the main task of a steganography designer. The sender can produce the stego object by applying the sequence of changes in the cover object. The impact of the embedding process in images depends on the efficiency of the distortion function [96, 99]. This gives rise to perfectly secure steganography, which embeds a secret message while introducing the smallest possible distortion effects on the cover. Nowadays, most steganography algorithms [41, 39] depend on this principle for providing methods that can face steganalysis.

3. **Transform domain technique**: in these techniques, we manipulate frequency coefficients of the image rather than its pixels. The main principle is to compute a 2-D discrete unitary transform of the image, for instance using the discrete cosine transform (DCT), to manipulate the image in the transform domain, and then to perform the inverse transform. It is indeed possible to embed the secret message in various frequency bands of the cover image. Note that the frequency domain steganography is slower and more complicated than the simpler substitution technique in spatial domain [49].

   The most frequent applications of such a transform domain technique are either in DCT domain, or in DWT one. Let us detail them:

- **JPEG transformation**: JPEG format is used in image processing to reduce the size of the image file. Firstly, in color images, the RGB format is converted into a YUV representation, the Y component representing brightness while the U and V components act for color or chrominance. Then, the image is divided into blocks of 8×8 pixels, and these blocks are converted into discrete cosine transform 64-DCT coefficients. After that, the quantization step is computed. The encoding of coefficients is applied by using Huffman codes to reduce the size of the image file. In this context, a steganography system that uses the JPEG image format to conceal the secret message in the non-zero coefficients of discrete cosine transform is possible [73, 46, 71].

- **Wavelet transformation**: wavelets transform (WT) converts too the signal from the spatial domain to the frequency domain. It is a mathematical function that separates data into frequency components, which makes them suitable for image compression. The wavelet transform divides the image to the high frequency and the low frequency information, where the high frequency contains information about the edge elements and the low frequency is divided again into high and low frequency components [81, 93, 82]. Figure 1.16 represents an illustration of wavelet decomposition. As in the DCT case, we can embed the secrecy in the frequency coefficients.

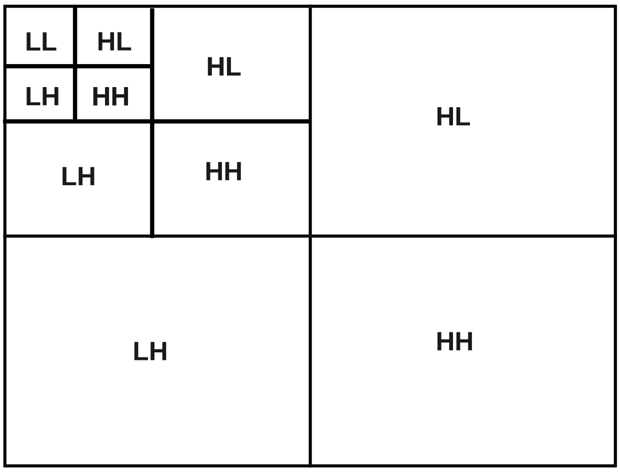


Fig. 1.16: Wavelet decomposition

- **Statistical technique**: It tries to modify some statistical properties of the cover image and to preserve them in the embedding process. A cover object is partitioned into disjoint blocks, each one containing modified bits. Each part finally corresponds to a single bit of the secret message [80, 94, 44].

1.4/ Steganalysis techniques

The counterpart of steganography is steganalysis, which is the other important field in information hiding domain. Steganalysis is the technology of detecting the presence of a secret information in a cover object. Steganalysis has recently received a lot of attention, as a countermeasure to the development of the steganography field. The art of steganalysis consists in magnifying the statistics or features that are sensitive to embedding operations.

The steganalyser is represented by the warden (Wendy) in Figure 1.10. His degree of knowledge varies according to the scenario. A popular one is when the warden does not have any information: neither about the cover object nor the steganography algorithm and the secret message. This difficult task requires analyzing all images going through the channel, to detect the presence of hidden messages. In this scenario, the warden needs steganalysis algorithms able to detect a wide range of steganography schemes. In another scenario, the warden has some information about secret messages or about the steganography algorithm. This information increases the rate of success for the warden [17, 14].

The warden has a stego detector as follows: $F : C \rightarrow \{0, 1\}$. The response of the warden is binary, where the answer is 0 for natural contents and 1 for stego ones. The detector can make two kind of errors. The false alarm (false positive) occurs when the warden decides that a hidden message is present, but in fact there is no hidden message. The second type of error is called missed detection (false negative), which occurs when the warden decides that a hidden message is absent while indeed it exists.

In a security point of view regarding the steganalysis system, there are three broad types of wardens, which can be described as follows [52]:

- **Passive warden**: it does not interfere with the content on the transmission channel. The steganalysis system goal in this case is to detect the mere presence of the secret communication between Alice and Bob in Fig. 1.10. The warden in this situation does not have the ability to destroy or modify the secret message that is detected by the steganalysis algorithm.

- **Active warden**: the objective of the active warden is to detect and slightly modify the communicated objects and then send virtually any content to the receiver side. Compression is an example of digital image modification that can be applied by an active warden.

- **Malicious warden**: the warden in this situation has the information about the key. The secret message can be modified in order to impersonate the sender and trick the receiver.

Under the Kerckhoffs's principle, the warden knows all details such as steganography algorithms, the probability distribution on cover objects, and other information about the transmission channel between the sender and the receiver – except the stego key. Note that this scenario rarely happens in reality [52].

1.4.1/ Classification of image steganalysis

The main goal of the steganalysis system is to detect the presence of a secret message in cover image. In general the steganalysis system can be divided into two parts as follow [10]:

1. **Specific or targeted Steganalysis**: these types of steganalysis systems fully depend on steganography algorithms that are being used to embed secret messages in cover images. They are sometimes also limited on the image format. These systems have high success rate but they are inflexible for working on other embedding algorithms. Bellow are listed some types of specific steganalysis approaches.
   - OutGuess attack [26],
   - MB1 attack [43],
   - LSB Matching Steganography attack [106],
   - YASS attack [58].
2. **Generic, Universal, or Blind Steganalysis**: the universal steganalysis systems are independent on the steganography algorithms. Generally, the universal steganalysis is preferred than the specific one. The embedding process for image leaves statistical artifacts on the image. These statistical artifacts are used to distinguish between natural and stego images. Among analysis that are used, we can evoke the following ones:
   - binary similarity [5],
   - wavelet based analysis [109],
   - feature based analysis [24].

1.4.2/ Methods of steganalysis technique

In most steganography techniques, a secret message was concealed either in a sequential manner or in a random one over the cover image. To discover the presence of a secret message, many techniques are used by the steganalysis algorithms. This process may be visually performed or by analyzing the structure of the image. The detection techniques can be described as follow [69]:

1. **Visual steganalysis**: visual inspection can detect the suspicious artifacts if these artifacts occur in connected areas of uniform color of the image, or if they occur in area of the image with values either 0 or 255. It is difficult to detect the presence of the hidden information in noisy images or textured ones. In this technique, the human eye represents the model of classification. The detection is applicable to palette image for LSB embedding, as reported for instance by Pfitzmann and Westfeld [105].
2. **Statistical steganalysis**: the statistical steganalysis performs the task of detection through some simplified model of the cover image, obtained by representing the cover image by a set of numerical features. There are many techniques that are used to detect the presence of a secret message by using a statistical approaches. Among them, we can report:

- **Chi-square Analysis**: Westfeld and Pfitzmann have proposed [105] a model of statistical analysis. They noticed the changes of the histogram of the color frequencies when embedding a secret message in a cover object. In this case, the embedding process changes the least significant bits of the colors in an image. The statistical steganalysis is more accurate and successful than the visual steganalysis, because it is able to discover small changes in a cover image.

3. **Structural steganalysis**: structural attacks are designed to take benefit from the properties of the used steganography algorithm. Each steganography algorithm leaves characteristics structure in a cover image when embedding the hidden information. The structure steganalysis may detect the presence of secret message by examining the changes in the characteristic of the structure. RS analysis, described below, is an example of structural attacks.

    - **RS steganalysis**: this is steganalysis system designed to detect LSB embedding schemes in color and grayscale images. To analyze an image, the approach specifies groups of pixels depending on some properties. Then it computes the relative frequencies of these groups for the given image to predict the embedding levels [25].

1.5/ Related work

1.5.1/ Features

Feature extraction techniques are useful in classifying the images into cover and stego contents in steganalysis fields. The embedding of secret messages in cover images can be considered as noise addition, due to the alterations made to the images during embedding. In order to detect such noise, features extraction is a crucial step for many steganalysis techniques, assuming that a non natural alteration of an original image can be signaled by significant changes in these features. Obviously, the steganalysis performance is highly dependent on the definition of such image features.

Both of targeted and universal steganalysis systems use features to detect hidden information in images. The effect of the embedding process in an image can be considered as adding noise of specific properties. This is why many features are proposed to be sensitive to adding noise and not sensitive to the image content. Then, after selecting the features set, the warden can construct the steganalysis system, mainly by using supervised classification.

At the beginning, features used for detection in a spatial domain steganography were different from features used in JPEG domain steganography. After that, spatial domain and JPEG domain features have merged. For instance, Subtractive Pixel Adjacency Model (SPAM) technique [74] is a method for detection of steganography scheme that embeds a secret message in a spatial domain. The SPAM features are computed in different directions, where the difference between pixels and transition probabilities are computed along same eight directions. These features focus on detection of LSB matching steganography. The SRMQ1 rich model [27], for its part, presented a general strategy for constructing steganography detectors for images. The idea starts with combining a rich model of the noise component as a union of different submodel,s that is composed of joint distributions of neighboring samples using linear and non-linear high-pass filters.

SRM method, proposed in [27], is an extended version of SRMQ1 features that increases feature diversity to detect the stego image. In JPEG domain, CC-PEV in [53, 76] presents features extracted directly from the DCT domain. It contains the DCT features that are extracted from the inter blocks dependencies among DCT coefficients, and the Markov features that are extracted from intra-block dependencies. They apply calibration to the extracted features to reduce their dimension. The JPEG domain rich model (JRM) feature [52, 28], for its part, consists of many diverse submodels taking from intra-block and inter-block among DCT coefficients. The CC-JRM in [52] is a cartesian calibration of the JRM features, which performs well for features directly extracted from the DCT domain. In [37], the DCTR features are computed from noise residuals obtained using the DCT coefficients. It has a lower computational complexity when compared with the other feature extractions methods. The JSRM features are merged between spatial domain SRMQ1 and JRM [52].

1.5.2/ Steganography algorithm

Recently, numerous algorithms have been developed in the field of images steganography. Some of these schemes work in the spatial domain, where they embed a secret message directly in image pixels. The other schemes transfer images in another domain, and then they use these coefficients to conceal the information. In what follows, we provide some explanations regarding these algorithms.

1. **Spatial domain**: steganography algorithms are based here on modifying the least significant bits of image. This principle depends on the fact that the least significant bits in an image could be considered from random noise, and to modify them when embed a secret message would have little effect on the image. Some steganography algorithms choose the LSB of pixels in a random manner, other techniques modify them in certain areas of the image, like texture region in an image. Bellow, some of the works are proposed in this domain.

    - **Edge-Adaptive (EA)**: this technique is proposed by Weiqi et al. in [65]. This algorithm modifies the pixel pairs that have a big difference in absolute value, for example the pixel in the region around edges in an image. They try to embed a secret message in sharper edge areas and leave the smooth areas unchanged.
    - **Wavelet Obtained Weights (WOW)**: in [41] Holub and Jessica Fridrich have proposed the distortion function to compute the cost of each pixel. This algorithm forces the embedding change to texture regions and leave the smooth regions. This steganography is more resistant to the steganalysis with rich model. The embedding algorithm uses syndrome trellis codes (STCs) to minimize the distortion for a given payload.
    - **S-UNIWARD**: this embedding process, proposed in [39], is similar to WOW algorithm. This technique uses the UNIWARD distortion function to embed a secret message in spatial domain. The pixel costs are computed from three directions depending on the horizontal, vertical, and diagonal wavelet coefficients.

- **STABYLO**: in [18], Couchot *et al.* embed a secret message based on a Canny edge detection filter. This algorithm is lighter than HUGO, WOW, and UNIWARD schemes.

- **MULTIVARIATE GAUSSIAN MODEL (MVG)**: in this method, the cover image is modeled as a sequence of independent distributed quantized Gaussians. The embedding process probabilities are derived to minimize the total Kullback-Leibler (KL) signal divergence when concealing the secret message using least-significant bit [31].

2. **Transform domain**: In transform steganography technique, the cover image is transformed to the required domain. Secret message bits are then embedded in the coefficients of the transformed cover image.

- **Jsteg**: the secret message is embedded in DCT coefficient after dividing the image into blocks then apply the DCT to each block [102].

- **F5**: this steganography algorithm [98] embeds a secret message in the nonzero AC DCT coefficient. It can be considered as the first method that uses the principle of the matrix encoding. The latter is used to improve the embedding efficiency.

- **nsF5**: Fridrich *et al.* in [29] have detailed an improvement of F5. This scheme removes the shrinkage that occurs due to the embedding in AC DCT coefficient, where a coefficient becomes zero after the embedding process. Shrinkage decreases the embedding efficiency.

- **OutGuess**: In [79], authors try to preserve the statistical properties of a cover image to prevent statistical detection. This algorithm embeds a secret information in the DCT coefficients.

- **J-UNIWARD**: Holub *et al.*, in [39], use the UNIWARD to embed a secret information in arbitrary domain. The distortion function is computed in a directional filter bank decomposition as a sum of relative changes between the cover and stego image applied in a wavelet domain. This steganography algorithm forces the embedding process to the regions that are difficult to model, in many directions such as noisy area and avoid the smooth areas in a cover object.

1.5.3/ STEGANALYSIS RESULT

The detector takes a learning based strategy that includes a training stage and a testing one. The extracted features are used both in training and testing stage. The trained classifier is extracted from the training stage. There are many classifiers in the field of artificial intelligence such as support vector machine (SVM), neural network (NN), Fisher linear discriminant (FLD), etc. Figure 1.17 explains the steganalysis classifier.

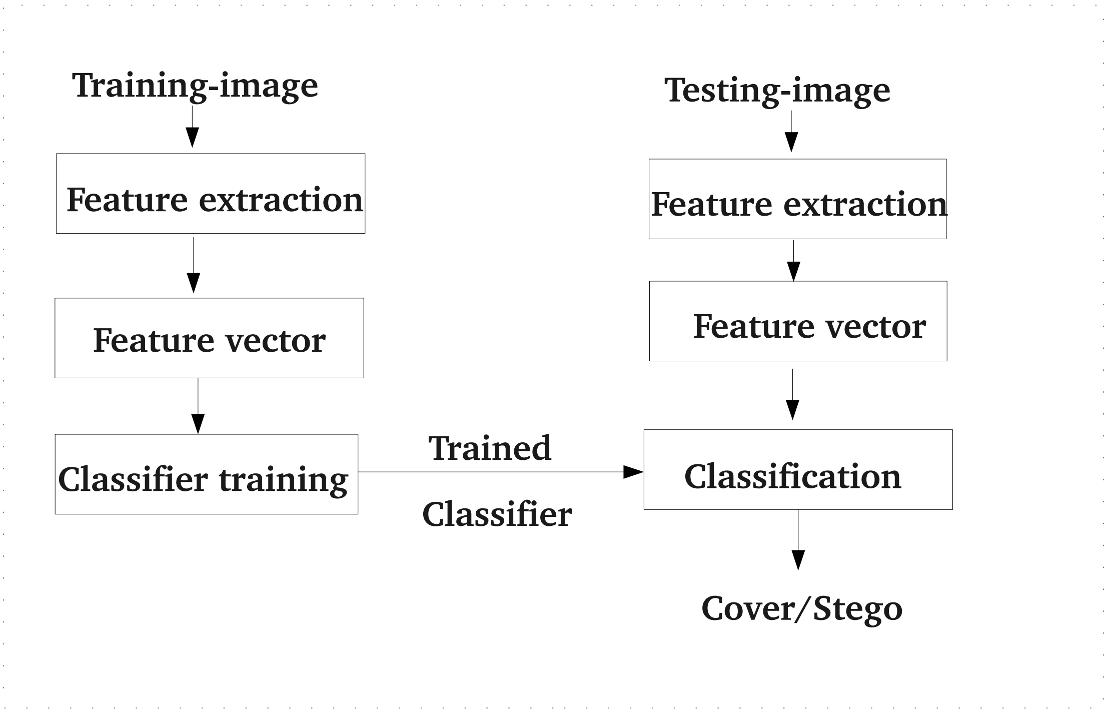
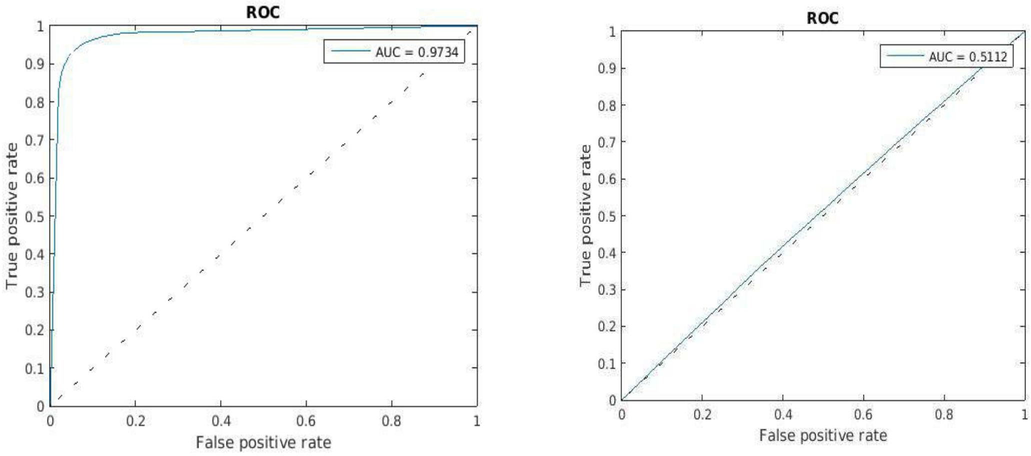


Fig. 1.17: Steganalysis classifier

To measure the performance of the steganalysis system, the receiver operating characteristic (ROC) approach is used. This evaluation process depends on the number of false alarm and missed detection. Steganalysis algorithms having their ROC curves close to the diagonal are not accurate. Conversely, a steganalysis technique s.t. the ROC curve has a big area under the curve (AUC) is accurate. Figure 1.18 illustrates two types of ROC curves.

Ensemble classifier [55] with Rich model [28] is used to detect steganography method that embeds a secret message in a spatial domain. Some schemes are detected by the following combination: Highly Undetectable steGO HUGO [23], where the detection rate of 10,000 images from BOSS base with payload 0.1 and 0.4 is 0.13 and 0.37 respectively. In this technique, the co-occurrence matrices size are increased exponentially with respect to the neighborhood length. This problem was solved by enlarging neighboring residual sizes and project them into random directions instead of extracting co-occurrence matrices [38]. This reduces the detection error rate for the payload 0.1 and 0.4 to 0.12 and 0.36 to the HUGO steganography method. The same experiment with the same set of the image was applied to WOW [41], S-UNIWARD [40] steganography methods. The error detection of the WOW method of payload 0.1 and 0.4 is 0.18 and 0.39 and for the S-UNIWARD scheme the error detection is 0.18 and 0.40 [38]. Denemark et al [19] have been obtained more accurate results, where they modified the calculus of the co-occurrence matrix by memorizing the maximum of the neighboring change probabilities



(a) Strong accurate detector

(b) Weak accurate detector

Fig. 1.18: Types of ROC curve

instead of their mean. According to this modification, the error detection was reduced to 0.19 and 0.37 for payload 0.1 and 0.4 to the S-UNIWARD, while for the WOW method the error detection is 0.15 and 0.30.

1.6/ Conclusion

This chapter has presented an overview of how to use images as covers in which to hide secret messages. Steganography techniques can use images in spatial domain by directly embedding the information in the image pixels. Similarly, they can insert information in a transformed frequency view of the image. Steganography schemes that work in spatial and frequency domains have been described. Additionally, steganalysis techniques that are used to detect secret messages in images have been explained. The receiver operating characteristics (ROC) has finally been presented, as a method that is useful for evaluating steganography algorithms.

II

CONTRIBUTIONS

2

Performance Study of Steganalysis Techniques

2.1/ Introduction

The presence of many methods of steganography led to study the effect of some of these methods on well-known steganalysis systems. The steganalysis system was examined according to changing some factor such as the payloads, the type of steganography schemes, and the set of images that are used in training and testing stage in the steganalysis operations. The reason to be of this work is to consider the evaluation process that is used to compare information hiding techniques. To do so, we will first determine the challenge corresponding to this evaluation process consisting mainly of 1) the rules of the game between the steganographier and the steganalyser, 2) the fairness in these rules, and 3) the data, in form of knowledge, shared by each player. This challenge will also be reasonably associated with some forensics situations. In addition, we will investigate more realistic challenges, not yet considered in the literature, to evaluate the behavior of up-to-date steganalysers in an operational context that corresponds more reasonably to real case attacks. In our evaluations, we aim at bridging the gap between laboratory approaches and real operational situations.
The performance of some state-of-the-art steganalysers is investigated according to various parameters, encompassing the choice of the steganographier, its payload, and the type of images both during training and testing stage. All these parameters are changed to determine their effects on steganalysis performance. Experiments are performed using large sets of grayscale JPEG images of different types. The results indicate that modifying parameters that are usually considered in the literature, and which are very specific, dramatically decreases steganalysis performances. This chapter is organized as follows. Section 2.2 describes all the experiments and explains the criteria used to evaluate the Ensemble Classifier steganalyser tool, including payloads and steganography methods. The evaluation results are described in Section 2.3. This work has been published in IEEE Applied Research in Computer Science and Engineering (ICAR), 2015.

2.2/ Experimental Setup

The performance of the Ensemble Classifier steganalyser has first been evaluated for its dependence on the image databases used in our training/testing stages. The databases used in these experiments are presented in Section 2.2.1.

2.2.1/ Image databases

It is reasonable to wonder whether the sensitivity of the Ensemble Classifier depends on the image databases used either during training or testing stage. The motivation behind this questioning is that, unlike the secret message recipient, the adversary[1] has not necessarily access to the same set of images used to train the Ensemble Classifier during the challenge. In some of our experiments we will investigate the image database sensitivity of the steganalysers results. We have used three coherent sets of images, downloaded from three different websites, as described in Table 2.1. We note that each image has been converted to a 512×512 grayscale JPEG.

Table 2.1: Sets of images used during experiments.

Category	Boss images	Art images	Plant and landscape
Number of images	7518	8745	10788
Origin	Boss	WGA	Various websites

2.2.2/ Working procedure

Steganography system detectors consist of two basic parts: an image modeling and a machine learning stage. The machine learning tool is trained using a set of features extracted from cover and stego images. The Ensemble Classifier classifies thereafter these feature vectors derived from both cover and stego images. In these experiments, we provided to the Ensemble Classifier two feature sets, namely CC-PEV and CC-JRM. Both nsF5 and J-UNIWARD have been considered to embed secret messages in the experiments described thereafter.

All the experiments are used to investigate the performance of the Ensemble Classifier against changes in various parameters like payload, steganography methods, and so on. In the following experiments, the secret message is embedded in cover images using steganographic methods to obtain stego images. Features are then extracted from both cover and stego images. These features are finally used by the steganalysis system to determine whether a secret message exists in the image.

[1] a person or a process with intentions of compromising the secret message

2.3/ Results Evaluation

The behavior of the Ensemble Classifier regarding payload modifications is evaluated in the next section. The impact of changing the steganography method is evaluated in Section 2.3.2. Finally, the effect of modifying the images during training and testing stages is investigated in Section 2.3.3.

2.3.1/ Detection evolution when modifying payloads with nsF5

Steganalysers are usually evaluated in the literature using a steganographier chosen by the evaluator. This latter is usually set at a payload of 0.2, which means that 1 bit of secret message is inserted to each 5 pixels of the host image. Here, we evaluate the performance of the state-of-the-art steganalysers using payloads that range from 0.004 to 0.2. In fact, we consider that a too large payload size is 1) not realistic, 2) totally unfair for the steganographier, and 3) can be too much lenient for the steganalyser side.

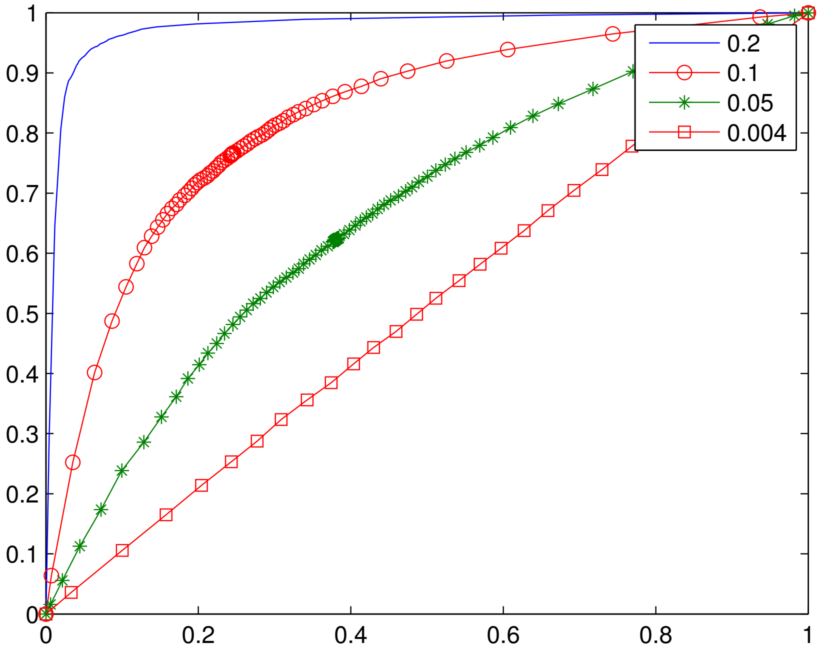


Fig. 2.1: ROC curves on different payloads.

In this run of tests, 27,051 JPEG images of various types have been considered as cover images, and the nsF5 steganography method has been used to embed a random secret message into the host content. Four different lengths of messages have been considered according to the payloads 0.004, 0.05, 0.1 and 0.2 which are respectively 1049, 13107, 26214 and 52429 bits. The smallest length corresponds to the length of tweet messages, which in practice can be enough to send useful information secretly.

The set of images in our experiments have then been randomly separated into two equal subsets; one has been used for training the Ensemble Classifier while the other has been provided to the testing stage. The performance obtained with the aforementioned payloads are presented in Figure 2.1. As it can be observed, the results of the Ensemble Classifier set with payloads used in the literature (0.1 and 0.2) are acceptable, while when the payloads are more realistic (0.05 or 0.004), the performance degradation is obvious. In other words, this state-of-the-art Ensemble Classifier totally fails in separating natural and stego contents when considering hidden messages of reasonable lengths like a twitt.

2.3.2/ Classification sensibility to the steganography method

A second disputable choice of the literature is to use the same steganographier in the learning stage of the steganalyser and in its evaluation. This means that the adversary knows which tool has been used by the steganographier. However in the reality it is rare, though possible, that the steganalyser has both this knowledge and the access to the steganographic tool.

The challenge in the literature is then “Knowing that the steganographier has used that tool, which is publicly available, can you separate original and stego-contents using your set of images and a known payload of 0.1”. Again, this game is easy to win, but totally unfair for the steganographier side. A more realistic challenge should be: “Given this set of images, can you answer the following questions: (1) Are there some hidden messages ? (2) Can you separate original and stego contents ? (3) Can you provide information on the steganographic tool ? (4) What about the secret message ?” Indeed, in operational steganalysis, the tool used by the steganographier is not known by the steganalyser. Furthermore, a steganographier aware of current information hiding advances will use neither publicly available tools nor public images, but instead he/she will prefer to design his/her own method to evaluate on his/her set of images.

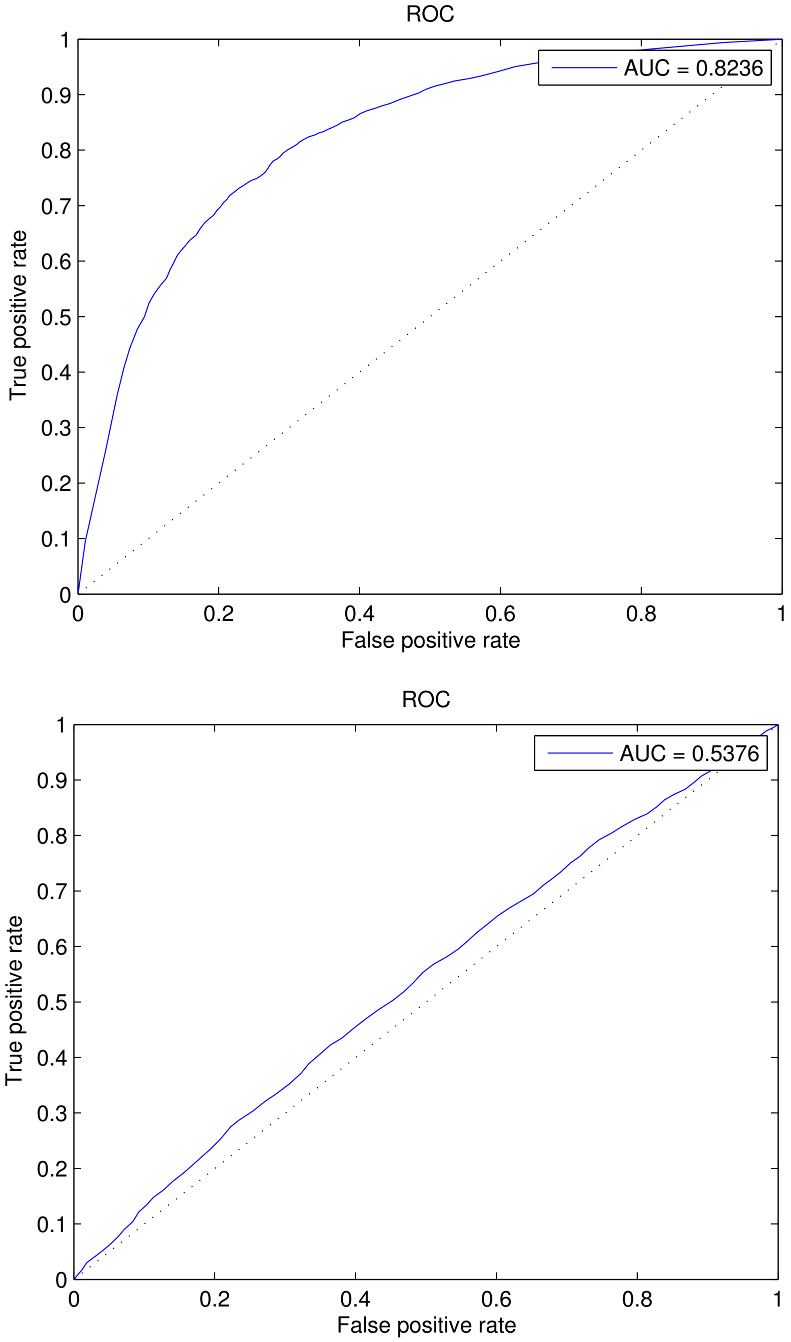


Fig. 2.2: ROC curve when learning has been realized using nsF5 and J-UNIWARD.

The objective of this section is thus to test the Ensemble Classifier versus several steganography methods. In this experiment, a set of 5,788 “natural” homogeneous images of size 512×512 is used with a payload of 0.1. CC-PEV548 is used to extract 548 features from each image. The steganalyser must classify images that are steganographied either using nsF5 (first experiment) or J-UNIWARD (second one).

As it can be observed in Figure 2.2, using this payload, the steganalyser is able to accurately identify original from stego contents for images steganographied using nsF5. However, when the J-UNIWARD is used instead, the steganalyser behavior is similar to a random black box, and it totally fails in separating these content. Indeed it is not surprising, as nsF5 operates on DCT coefficients while J-UNIWARD modifies the wavelet ones.

The JPEG rich model (JRM) CC-JRM [54] feature extraction library is now used in place of the CC-PEV548 one. CC-JRM extracts 22,510 features from a large number of smaller submodels of DCT distribution coefficients from both cover and stego images. When the CC-JRM features are used in the Ensemble Classifier, results are acceptable if nsF5 is used, see Figure 3.3. But the Ensemble Classifier ability to separate original and stego contents fails when using J-UNIWARD.

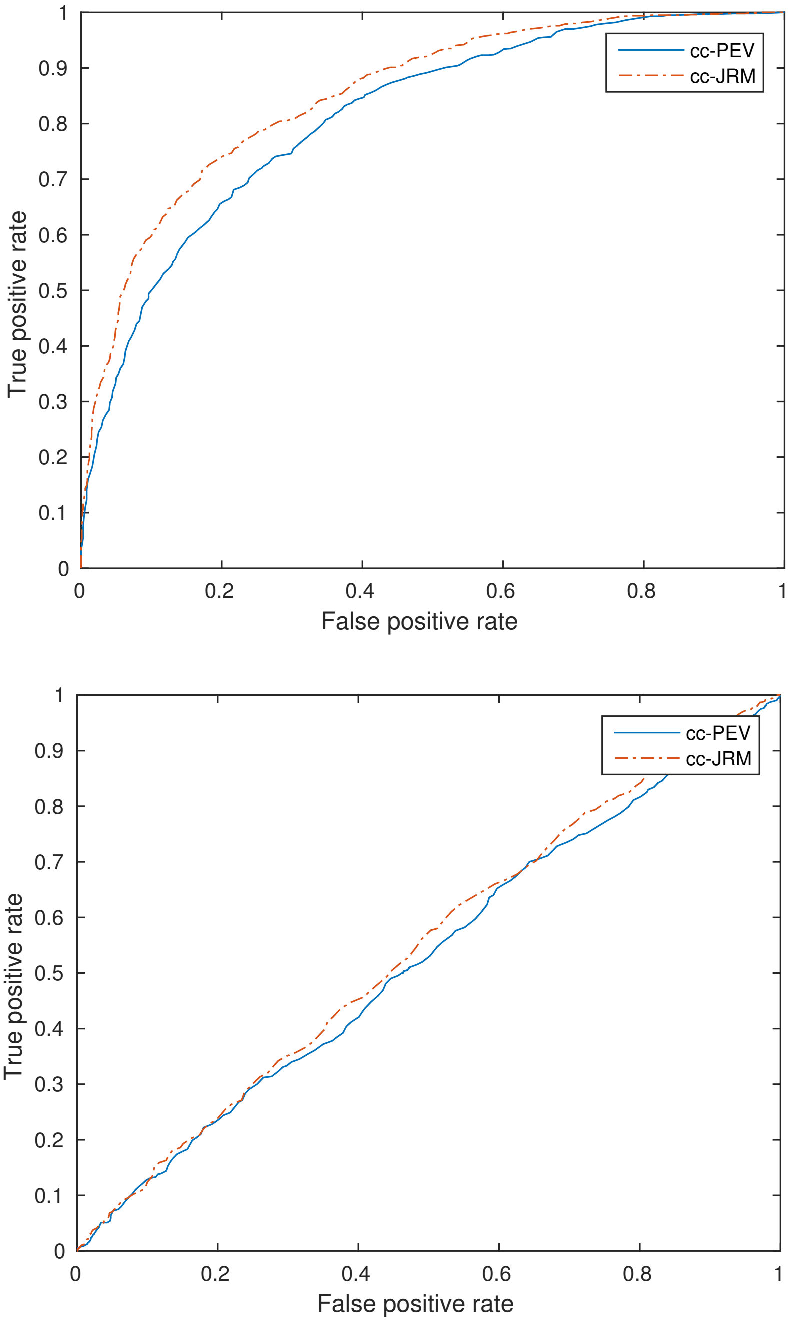


Fig. 2.3: ROC curve when using different methods of feature extraction CC-PEV, CC-JRM for nsF5 and J-UNIWARD.

To sum up, in the two scenarios investigated above and with two different features extraction, it is hard for the Ensemble Classifier to distinguish between cover and stego images, even when all accessible information (images, features and steganographiers) is assumed to be known during the challenge.

2.3.3/ Modification on training and testing sets

In real life, there is no restriction when choosing the groups of images used for embedding the secret message, and the steganalyser cannot force the steganographier to use a suitable set of pictures. This is why we now investigate what happens when there is a change in the groups of images between the training and testing stage. This corresponds to the situation where the steganalyser has not access to the set of original images used on the steganographier side, which should be the usual evaluation context: when both sides share the same set of images, the steganalyser has only to test if the images sent through the communication channel is in his set of images.

In this last experiment, 27,051 images have been used as covers and the same number as stego. nsF5 is used as steganography method with a payload of 0.1. The features are computed from the cover and stego images by using the CC-PEV548 method. However, in this test, the default strategy for building training and testing set is not used: images are selected here according to the difference D between the cover images X and its stego images *emb*(X, m) on fifty co-occurred features. These features depend on the neighboring DCT coefficients and they are determined as follows:

$$D = \sum_{f \in F} |f(X) - f(\textit{emb}(X, m))| \tag{2.1}$$

where F is the set of co-occurrence features.

This value is used to select the images that are more sensitive to the variations in the value of these specific features. According to this difference, we selected 15,000 images having a large difference between their cover and stego version, while 4,000 images are chosen with a small difference. When the Ensemble Classifier is trained with images that have large differences and tested with the images that have small ones as in Figure 2.4, we are able to deflate the classification, as illustrated by the evolution of the ROC curve from Figure 2.2 to Figure 2.4.

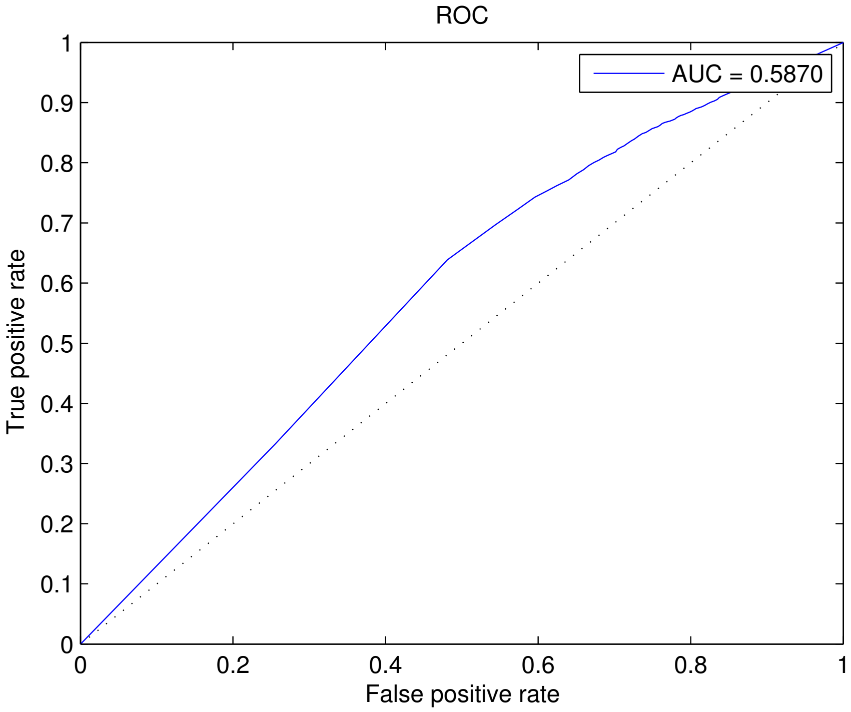


Fig. 2.4: ROC curve according to the type of train and test sets.

2.4/ Conclusion

We have initially investigated the ability of the Ensemble Classifier to separate original from stego contents when using different payloads. We have shown that the classifier cannot detect the presence of secret messages when the latter have small length such as a tweet message.

The Ensemble Classifier has been evaluated with several steganographiers. In this experiment we have demonstrated that the steganalysis results can be improved when the learning set is constructed with a much coarser steganographier (such as nsF5) while keeping in mind that the objective is to classify images that may be modified with a more efficient scheme (namely UNIWARD).

We showed that the Ensemble Classifier combined with CC-PEV548 and CC-JRM fails to distinguish between the cover and stego when the steganographier is based on wavelet coefficients.

And finally, we have evaluated the Ensemble Classifier in a scenario where the steganographic scheme is not previously known, leading again to a fail of separation between original and stego contents.

With the development of numerous steganography methods, it becomes easy to use various steganographic schemes in training and testing stages, as a means of improving the detection rate of the message presence in cover object (regardless all the other information that are required to detect a message in an image). This is what is investigated in the next section.

3

STEGANALYZER PERFORMANCES IN OPERATIONAL CONTEXTS

3.1/ INTRODUCTION

Steganalyzers of the literature are usually evaluated as follows. The steganographier scheme s is firstly chosen, while a large set of images are separated in two sets, half of each two parts being steganographied using s. Then the first set is used during the learning stage, while the steganalysis method is evaluated using the second set. Such an evaluation corresponds to the particular situation where the warden Eve (the steganalyzer) has the knowledge of which steganographier has been used, with which parameters (embedding payload, etc.) In this work, we will investigate a more realistic scenario where Eve only knows that images contain secret messages: she does not know which steganographic algorithm has been used, and the game consists of separating well original from stego contents. More precisely, in this research work, we show what happens when the learning stage has been realized with a wrong steganographier, and we ask whether it is useful to use more than one steganographier during the learning stage to face this problem. In this chapter, we determine whether it is possible to construct a kind of universal steganalyzer without any knowledge regarding the steganographier side. The effects on the classification score of a modification of either parameters or methods between the learning and testing stages are then evaluated, while the possibility to improve the separation score by merging many methods during learning stage is deeper investigated.

This chapter is organized as follows, in Section 3.2, we first investigate the effect of a wrong assumption on the steganographier during the learning stage. In Section 3.3, we wonder whether it is possible to solve this problem by mixing more than one steganographier during the learning stage, in order to design a kind of universal detector. Errors on payload assumption are then discussed in Section 3.4. All these situations are merged in Section 3.5, leading to what can be expected for operational contexts. This work has been published in IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP).

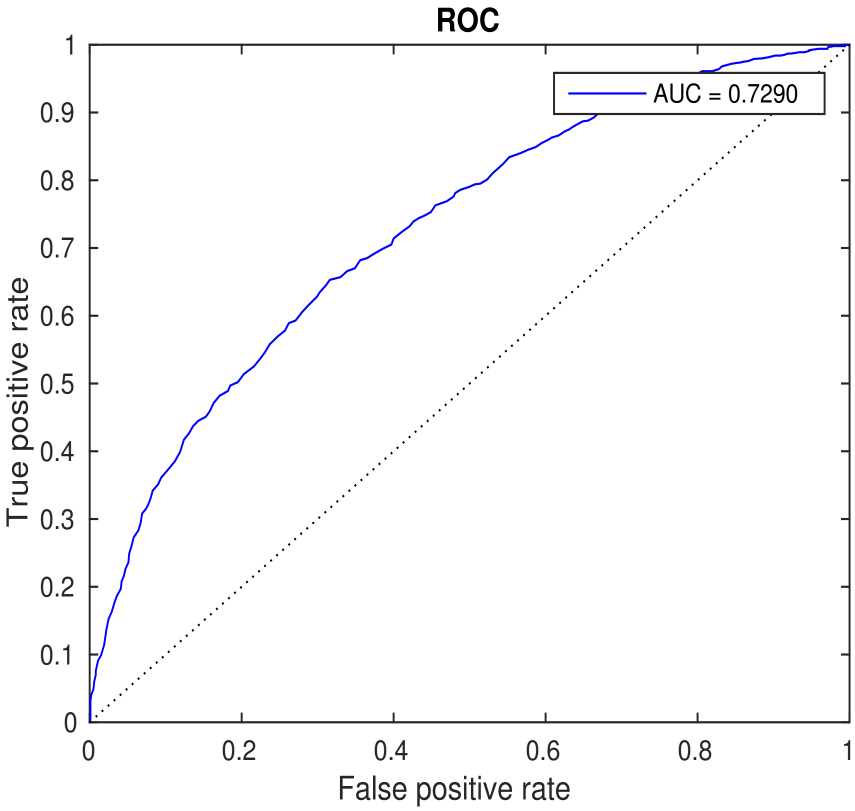
3.2/ Training and testing stages use not the same steganographier

Let us first measure the effects of modifying the steganographic method between the training and the testing stage. To investigate this question, $2,000$ original JPEG images have been used in our experiments. They are taken from the BOSS contest [75], their size is equal to 512×512, and they have been converted to JPEG. For stego images, an embedding payload of 0.1 is used. The method used for extracting the features from the images is CC-PEV. The same ensemble classifier has been used both in the training and in the testing stage, namely the one of [56].

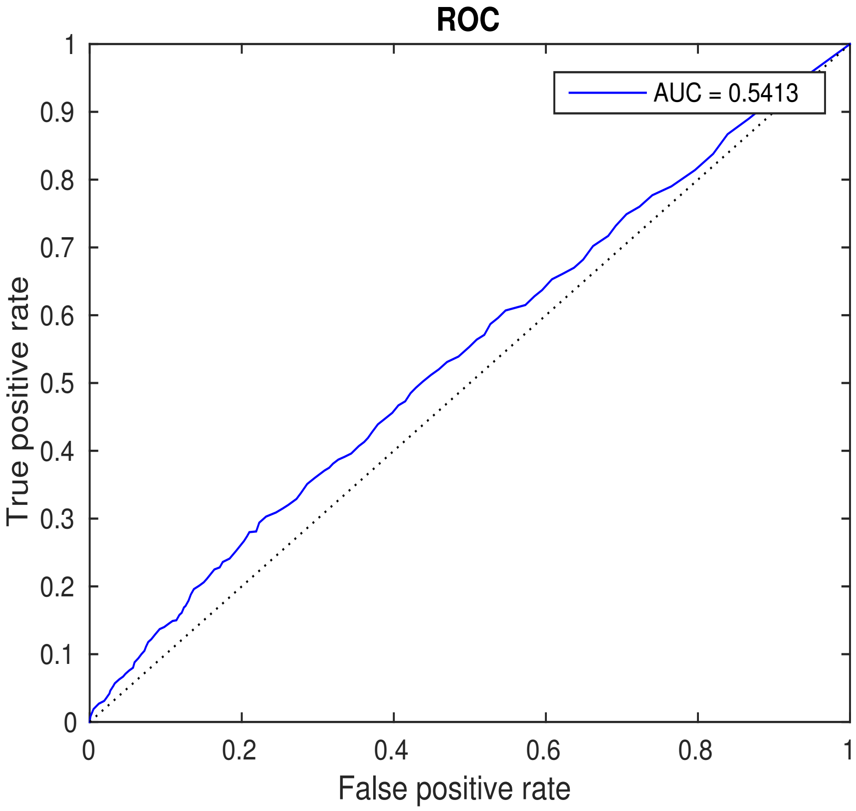
In the first experiment, the ensemble classifier is trained using 50% of the natural images and 50% of the same images steganographied by nsF5, while it is tested using the same rate of natural and J-UNIWARD images. Conversely, in the second set of experiments, J-UNIWARD is used during the training stage and nsF5 during the testing one. Obtained results are presented in Figure 3.1. It can be seen that the presence of hidden messages embedded with J-UNIWARD is more or less detected when the steganalyzer has been trained by using nsF5. Conversely, the detection of nsF5 is impossible when learning with J-UNIWARD. This asymmetric behavior may be explained by the fact that the use of nsF5 affects more the general aspect of the embedding image compared to J-UNIWARD. So, the ensemble classifier can learn more from the former than from the latter, and its classification is thus more efficient and trustworthy. This result has been obtained again when considering all other possible combinations, see Table 3.1: the only acceptable performances are obtained when nsF5 is used during the training stage.

Table 3.1: Errors when choosing the learning steganographier with payload 0.1

Learning stage	Testing stage	A.U.C.	A.T.E
nsF5	J-UNIWARD	0.7290	0.3569
J-UNIWARD	nsF5	0.5413	0.4675
nsF5	HUGO	0.7523	0.3345
HUGO	nsF5	0.5371	0.4737
J-UNIWARD	HUGO	0.5122	0.4912
HUGO	J-UNIWARD	0.5077	0.4915



(a) nsF5 for training, and J-UNIWARD for testing with payload 0.1



(b) J-UNIWARD for training and nsF5 for testing with payload 0.1

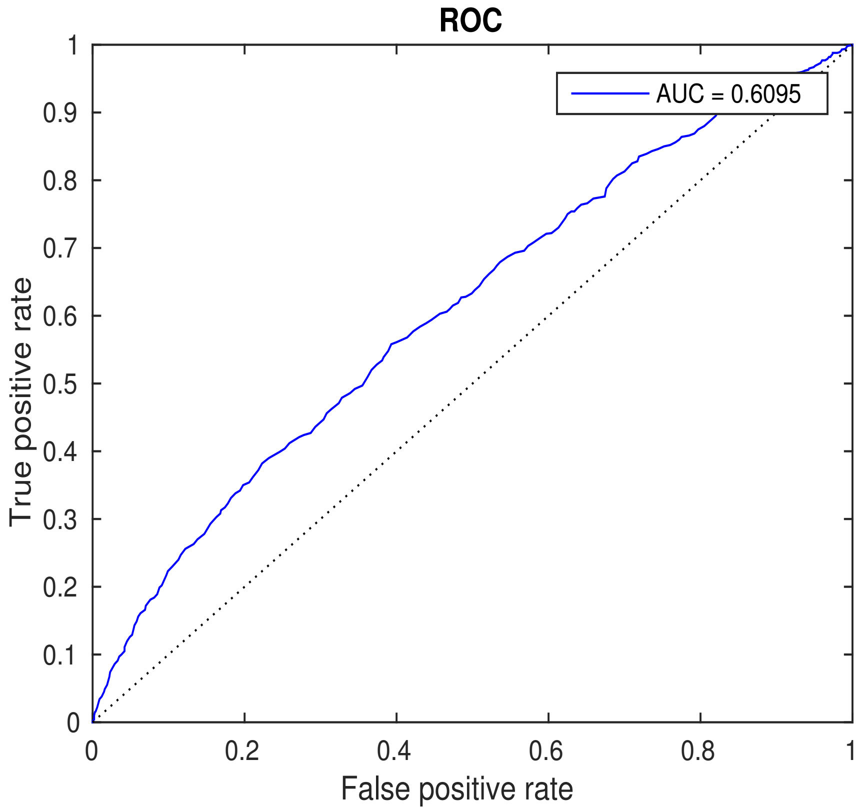
Fig. 3.1: The steganographier used during the training stage is not the good one.

3.3/ Trying to improve steganalyzer score by mixing learning steganographiers

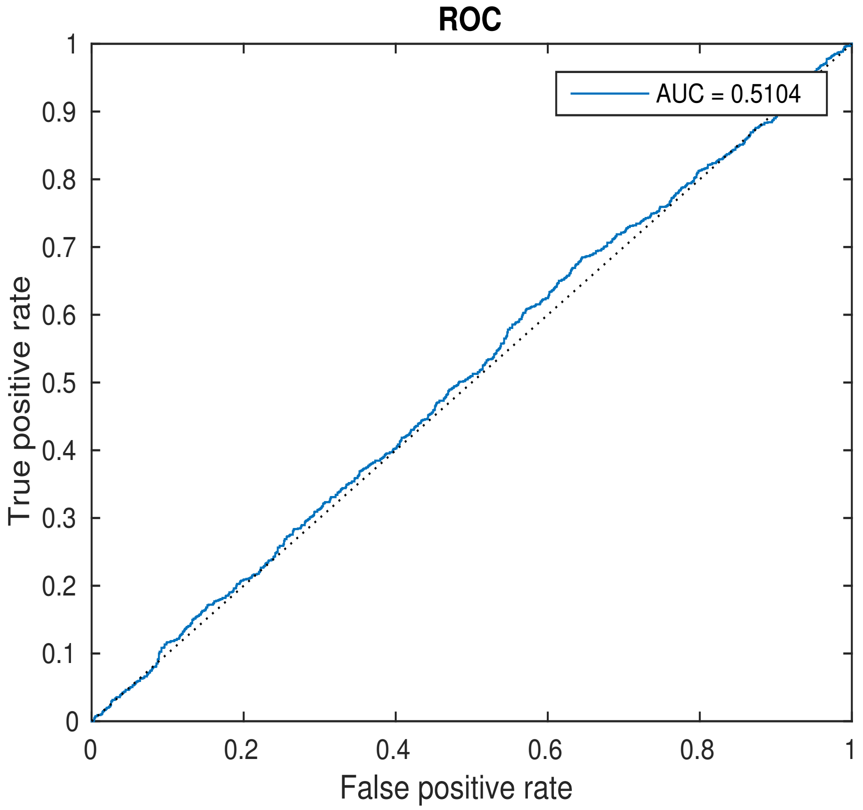
In this new scenario, we wonder whether the steganalysis performance can be improved by using more than one steganographier during the learning stage: if two or three steganographiers are suspected by Eve, can she use such a suspicion to produce a more accurate steganalyzer ? Or, to say this differently, is it possible to create a kind of universal steganalyzer by using a large set of steganographiers during the learning stage ? Results of these experiments are given in Table 3.2 and partially illustrated in receiver operating characteristic (ROC) curves of Figure 3.2. In this table, each row corresponds to an experiment where more than one steganographier has been used during the learning stage. Each tuple in this table gives the proportion of, respectively, natural images, HUGO, J-UNIWARD, and nsF5 stego-contents that has been used to constitute the set of 2,000 images, either during training or during testing stage. A payload of 0.1 has been used as previously. However, the area under the curve (AUC) obtained here never becomes larger than 0.7, while it was the case in Table 3.1, setting at naught the hope to constitute universal steganalyzer by mixing several tools when training.

Table 3.2: Study of accuracy by mixing various steganographiers when training. Each tuple represents the respective percentage of natural images, HUGO, J-UNIWARD, and nsF5 stego-contents.

Learning stage	Testing stage	A.U.C	A.T.E
(50, 25, 0, 25)	(50, 0, 0, 50)	0.6899	0.3584
(50, 25, 0, 25)	(50, 50, 0, 0)	0.6097	0.4269
(50, 0, 25, 25)	(50, 0, 50, 0)	0.6133	0.4284
(50, 0, 25, 25)	(50, 0, 0, 50)	0.6914	0.3518
(50, 25, 25, 0)	(50, 50, 0, 0)	0.5104	0.4920
(50, 25, 25, 0)	(50, 0, 50, 0)	0.5208	0.4855
(50, 25, 25, 0)	(50, 0, 0, 50)	0.5415	0.4692
(50, 25, 0, 25)	(50, 0, 50, 0)	0.6039	0.4306
(50, 0, 25, 25)	(50, 50, 0, 0)	0.6158	0.4149
(50, 25, 25, 0)	(50, 25, 0, 25)	0.5404	0.4718
(50, 25, 0, 25)	(50, 25, 25, 0)	0.6072	0.4303
(50, 0, 25, 25)	(50, 25, 25, 0)	0.6585	0.4199
(50, 25, 0, 25)	(50, 0, 50, 0)	0.6095	0.4311
(50, 0, 25, 25)	(50, 50, 0, 0)	0.6321	0.4155



(a) In the training stage: 1000 natural images, together with 500 nsF5 and 500 HUGO stego-contents. In the testing stage: 1000 natural and 1000 JUNIWARD.



(b) In the training stage: 1500 natural + 1500 JUNIWARD, while in the testing set: 500 natural and 500 JUNIWARD

Fig. 3.2: Mixing various steganographiers in the learning stage.

3.4/ Uncertainty effects regarding payload

The objective is now to emphasize the possible effects of payload ignorance on steganalyzer performances. Indeed, a large payload of 0.1 is always chosen for evaluating steganalyzers of the literature. By doing so, steganalyzer designers made strong assumptions that make life less complicated, and the game totally unfair in their own advantage. These two assumptions are that the steganographier will absurdly use a very large payload, and additionally this payload is known by the steganalyzer. Everything happens as if steganalyzer designers claim to be able to detect if a communication channel possibly contains stego images, while they finally answer to the challenge: “knowing the set of images, the presence of hidden information, the steganographier, and the payload, can we separate with a good accuracy the natural from the stego images.” On our side, we argue that it is not possible to expect exactly the payload value chosen by the steganographier in operational contexts.

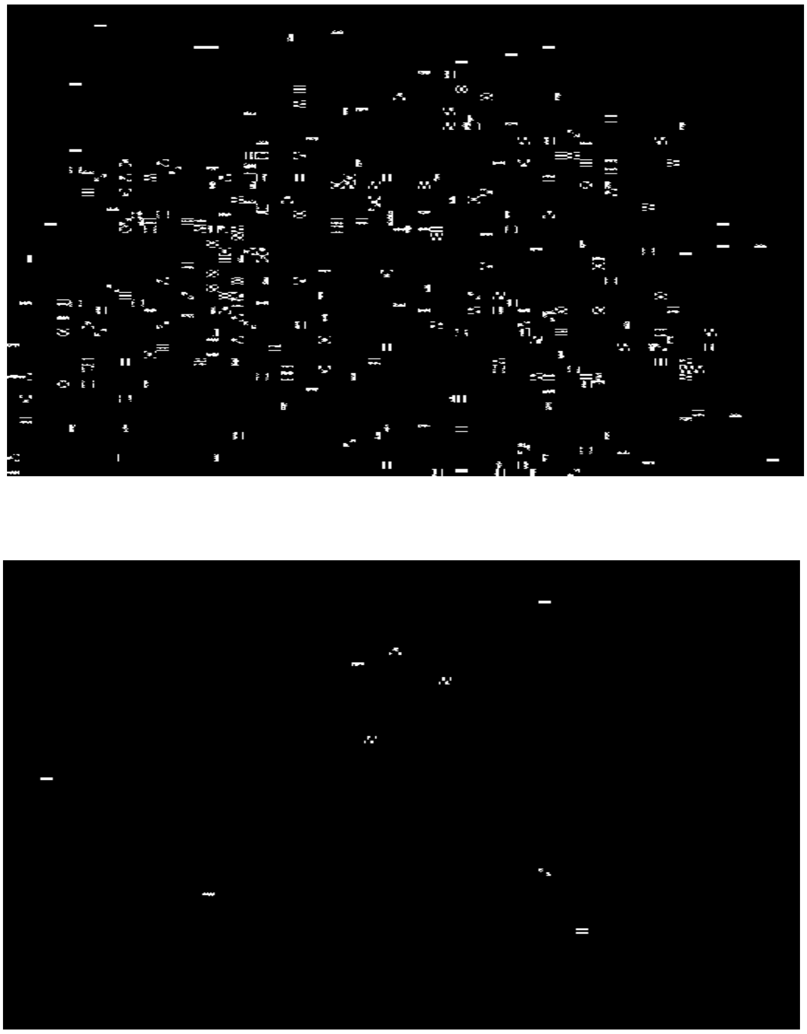


Fig. 3.3: Differences between host content with nsF5 when payload is respectively equal to 0.1 or 0.005.

In this new run of tests, images are steganographied by using respective payloads of 0.005, 0.05, and 0.1 (see Figure 3.3 to understand the effects of such payloads on host contents). CC-PEV features are used with ensemble classifier in both training and testing stages. nsF5, J-UNIWARD, and HUGO have been successively tested using the 3 payloads listed above, to illustrate the effects of such an error for both spatial and frequency embedding. Obtained results are summarized in Table 3.3. As can be seen, the only situation where the separation is acceptable is the nsF5 one, and when training with a large payload that helps the ensemble classifier to learn the embedding effects.

Table 3.3: Result of steganalysis when there is a payload error during training

	Train	Test	A.U.C	A.T.E
nsF5	0.1	0.05	0.7855	0.2854
	0.1	0.005	0.7723	0.3195
	0.05	0.1	0.6656	0.3933
	0.005	0.1	0.5408	0.4717
J-UNIWARD	0.1	0.05	0.5049	0.4949
	0.1	0.005	0.5091	0.4955
	0.05	0.1	0.5087	0.4958
	0.005	0.1	0.5035	0.4980
HUGO	0.1	0.05	0.5175	0.4898
	0.1	0.005	0.5161	0.4885
	0.05	0.1	0.5182	0.4867
	0.005	0.1	0.5192	0.4853

3.5/ Operational contexts

We now consider the most realistic scenario where the steganalyzer side only knows that one of the 3 most famous steganographier tools are used. But he is not sure about the chosen payload. Obtained results when mixing both the steganographier and its payload between training and testing stages have then been computed, and obtained results are summarized in Table 3.4.

As can be deduced from this table, the classification is acceptable only when the learning process has been realized with nsF5 and with a larger payload than the one that has been used during the tests. In this situation, it has been possible to separate, with a medium accuracy, images steganographied by either HUGO or J-UNIWARD. Remark that obtained results are better than what has been found in Table 3.3.

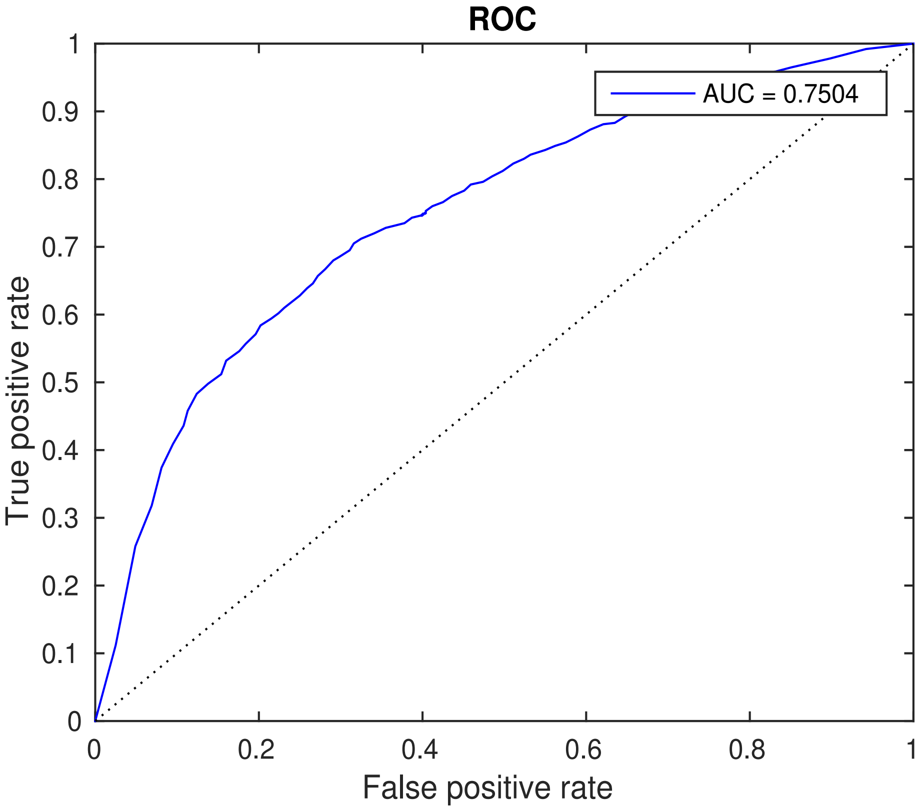


Fig. 3.4: Train with nsF5 with a 0.1 payload and test with J-UNIWARD with a 0.005 payload

Table 3.4: AUC scores in operational contexts

	Train	Test	A.U.C	A.T.E
Train: nsF5	0.1	0.05	0.7388	0.3530
Test:J-UNIWARD	0.1	0.005	0.7504	0.3325
	0.05	0.1	0.5926	0.4482
	0.005	0.1	0.5057	0.4952
Train:nsF5	0.1	0.05	0.7489	0.3320
Test:HUGO	0.1	0.005	0.7554	0.3327
	0.05	0.1	0.5791	0.4378
	0.005	0.1	0.5041	0.4971
Train:J-UNIWARD	0.1	0.05	0.5140	0.4905
Test:HUGO	0.1	0.005	0.5119	0.4916
	0.05	0.1	0.5035	0.4967
	0.005	0.1	0.5020	0.4997

3.6/ Conclusion

This chapter has focused on experiments in Kerckhoffs's context: everything about the used steganographic schemes, except the key, are known by steganalysis systems. Thanks to a large number of experiments, we indeed have shown that even J-UNIWARD can be detected when learning with other steganographic tools (namely HUGO and NSF5). This is observed even if the objective is to analyse a small payload based steganographic tool. In such a situation, it is sufficient to set a large payload in the learning step.

After studying the factors that have effects on the steganalysis system in the first contribution, then the steganalysis performance in operational context has been studied in the second one. It becomes interesting to develop a steganography scheme that embeds a secret message in cover object, but in noisy areas like texture and edges. This is the objective of the next chapter.

4

A Second Order Derivatives based Approach for Steganography

4.1/ Introduction

The steganography tools exploit the presence of edges in the cover image as a chaotic region to embed the secret messages in cover image object. The edge detection operation begins with the checking of the discontinuity at each pixel. Gradient, amplitude and orientation are important characteristics of possible edges. Depending on these properties, the edge detection algorithms have to determine whether a pixel is an edge or not. Most of edge detection algorithms are based in some way of measuring the intensity gradient at a point in the image. The gradient image magnitude represents the strength of the edge, it means the difference amount between pixels. The gradient orientation represents the direction to the greatest change, which perhaps is the direction across the edge. Edge detection in an image is traditionally applied by convolving the signal with some type of filter, generally a filter that approximates a first or second derivative operator.

Steganographic schemes are evaluated according to their ability to face steganalyser tools. An error is either a false positive decision or a false negative one. The average error is thus the mean of these two ones. Let us select a security level expressed as a number in $[0, 0.5]$, when developing a new steganographic scheme, the objective is to find an approach that maximizes the size of the message that can be embedded in any image with an average error larger than this security level.

Steganography schemes are designed with the objective of minimizing a defined distortion function. In most existing state of the art approaches, this distortion function is based on image feature preservation. Since smooth regions or clean edges define image core, even a small modification in these areas largely modifies image features and is thus easily detectable. On the contrary, textures, noisy or chaotic regions are so difficult to model that the features having been modified inside these areas are similar to the initial ones. These regions are characterized by disturbed level curves. This work presents a new distortion function for steganography that is based on second order derivatives, which are mathematical tools that usually evaluate level curves. Two methods are explained to compute these partial derivatives and have been completely implemented. In each edge detection techniques there are attempts to detect the most important edges by applying threshold techniques. For the first experiment, the second order derivative is chosen as a method for edge detection to select noisy regions. These chaotic regions are used to embed a secret message depending on the distortion function. The experiment gave an idea to use the second derivative as a base for steganography methods. To complete this work and to achieve the target of the steganography algorithm of the resistance to the existing steganalysis systems, we focus on selecting some points from chaotic regions to conceal the secret message. The selection of points depends on the threshold technique. This work first explains how such first and second order approximations can be computed on numerical images (Section 4.2). Two proposals to compute second order derivatives are proposed and proven (Section 4.3 and Section 4.4). An adaptation of an existing distortion function is studied in Section 4.5. A whole set of experiments is presented in Section 4.6. The threshold technique is explains in Section 4.7 Concluding remarks and future work are presented in the last section. Section 4.2 to Section 4.7 have been published in SECRYPT 2016.

4.2/ Derivatives in an Image

This section first recalls links between level curves, gradient, and Hessian matrix (Section 4.2.1). It next analyses them using kernels from signal theory

4.2.1/ Hessian Matrix

Let us consider that an image can be seen as a numerical function P that associates a value $P(x, y)$ to each pixel of coordinates (x, y). The variations of this function in (x_0, y_0) can be evaluated thanks to its gradient ∇P, which is the vector whose two components are the partial derivatives in x and in y of P:

$$\nabla P(x_0, y_0) = \left(\frac{\partial P}{\partial x}(x_0, y_0), \frac{\partial P}{\partial y}(x_0, y_0)\right).$$

In the context of two variables, the gradient vector points to the direction where the function has the highest increase. Pixels with close values thus follow level curve that is orthogonal to the one of highest increase.

The variations of the gradient vector are expressed in the Hessian matrix H of second-order partial derivatives of P.

$$H = \begin{bmatrix} \frac{\partial^2 P}{\partial x^2} & \frac{\partial^2 P}{\partial x \partial y} \\ \frac{\partial^2 P}{\partial y \partial x} & \frac{\partial^2 P}{\partial y^2} \end{bmatrix}.$$

In one pixel (x_0, y_0), the larger the absolute values of this matrix are, the more the gradient is varying around (x_0, y_0). We are then left to evaluate such an Hessian matrix.

This task is not as easy as it appears since natural images are not defined with differentiable functions from $\mathbb{R}^2$ to $\mathbb{R}$. Following subsections provide various approaches to compute these Hessian matrices.

4.2.2/ Classical Gradient Image Approaches

In the context of image values, the most used approaches to evaluate gradient vectors are the well-known “Sobel”, “Prewitt”, “Central Difference”, and “Intermediate Difference” ones.

Table 4.1: Kernels of usual image gradient operators

Name	Sobel	Prewitt
Kernel	$Ks = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}$	$Kp = \begin{bmatrix} -1 & 0 & +1 \\ -1 & 0 & +1 \\ -1 & 0 & +1 \end{bmatrix}$
Name	Central Difference	Intermediate Difference
Kernel	$Kc = \begin{bmatrix} 0 & 0 & 0 \\ -\frac{1}{2} & 0 & +\frac{1}{2} \\ 0 & 0 & 0 \end{bmatrix}$	$Ki = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$

Each of these approaches applies a convolution product $*$ between a kernel K (recalled in Table 4.1) and a 3×3 window of pixel values A. The result $A * K$ is an evaluation of the horizontal gradient, *i.e.*, $\frac{\partial P}{\partial x}$ expressed as a matrix in $\mathbb{R}$. Let $K^{\cdot}$ be the result of a $\pi/2$ rotation applied on K. The vertical gradient $\frac{\partial P}{\partial y}$ is similarly obtained by computing $A * K^{\cdot}$, which is again expressed as a matrix in $\mathbb{R}$.

The two elements of the first line of the Hessian matrix are the result of applying the horizontal gradient calculus first on $\frac{\partial P}{\partial x}$ and next on $\frac{\partial P}{\partial y}$. Let us study these Hessian matrices in the next section.

4.2.3/ Hessian Matrices induced by Gradient Image Approaches

First of all, it is well known that $\frac{\partial^2 P}{\partial x \partial y}$ is equal to $\frac{\partial^2 P}{\partial y \partial x}$ if the approach that computes the gradient and the one which evaluates the Hessian matrix are the same. For instance, in the Sobel approach, it is easy to verify that the calculus of $\frac{\partial^2 P}{\partial x \partial y}$ and of $\frac{\partial^2 P}{\partial y \partial x}$ are both the result of a convolution product with the Kernel Ks''_{xy} given in Table 4.2. This one summarizes kernels K''_{x^2} and K''_{xy} that allow to respectively compute $\frac{\partial^2 P}{\partial x^2}$ and $\frac{\partial^2 P}{\partial x \partial y}$ with a convolution product for each of the usual image gradient operator. The Sobel kernel Ks''_{x^2} allows to detect whether the central pixel belongs to a “vertical” edge, even if this one is noisy, by considering its vertical neighbours. The introduction of these vertical neighbours in this kernel is meaningful in the context of finding edges, but not very accurate when the objective is to precisely find the level curves of the image. Moreover, all the pixels that are in the second and the fourth column in this kernel are ignored. The Prewitt Kernel has similar drawbacks in this context.

Table 4.2: Kernels of second order gradient operators

Sobel	Prewitt
$Ks''_{x^2} = \begin{bmatrix} 1 & 0 & -2 & 0 & 1 \\ 4 & 0 & -8 & 0 & 4 \\ 6 & 0 & -12 & 0 & 6 \\ 4 & 0 & -8 & 0 & 4 \\ 1 & 0 & -2 & 0 & 1 \end{bmatrix}$	$Kp''_{x^2} = \begin{bmatrix} 1 & 0 & -2 & 0 & 1 \\ 2 & 0 & -4 & 0 & 2 \\ 3 & 0 & -6 & 0 & 3 \\ 2 & 0 & -4 & 0 & 2 \\ 1 & 0 & -2 & 0 & 1 \end{bmatrix}$
$Ks''_{xy} = \begin{bmatrix} -1 & -2 & 0 & 2 & 1 \\ -2 & -4 & 0 & 4 & 2 \\ 0 & 0 & 0 & 0 & 0 \\ 2 & 4 & 0 & -4 & -2 \\ 1 & 2 & 0 & -2 & -1 \end{bmatrix}$	$Kp''_{xy} = \begin{bmatrix} -1 & -1 & 0 & 1 & 1 \\ -1 & -1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & -1 & -1 \\ 1 & 1 & 0 & -1 & -1 \end{bmatrix}$

Central Difference	Intermediate Difference
$Kc''_{x^2} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{1}{4} & 0 & -\frac{1}{2} & 0 & \frac{1}{4} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$	$Ki''_{x^2} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$
$Kc''_{xy} = \begin{bmatrix} -\frac{1}{4} & 0 & \frac{1}{4} \\ 0 & 0 & 0 \\ \frac{1}{4} & 0 & -\frac{1}{4} \end{bmatrix}$	$Ki''_{xy} = \begin{bmatrix} 0 & -1 & 1 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$

The Central Difference kernel Kc''_{x^2} is not influenced by the vertical neighbours of the central pixel and is thus more accurate here. However, the kernel Kc''_{xy} again looses the values of the pixels that are vertically and diagonally aligned with the central one.

Finally, the Intermediate Difference kernel Ki''_{x^2} shifts to the left the value of horizontal variations of $\frac{\partial P}{\partial x}$: the central pixel $(0,0)$ exactly receives the value $\frac{P(0,2)-P(0,1)}{1} - \frac{P(0,1)-P(0,0)}{1}$, which is an approximation of $\frac{\partial P}{\partial x}(0,1)$ and not of $\frac{\partial P}{\partial x}(0,0)$. Furthermore the Intermediate Difference kernel Ki''_{xy} only deals with pixels in the upper right corner, loosing all the other information.

Due to these drawbacks, we are then left to produce another approach to find the level curves with strong accuracy.

4.3/ Second Order Kernels for Accurate Level Curves

This step aims at finding accurate level curve variations in an image. We do not restrict the kernel to have a fixed size (*e.g.*, 3×3 or 5×5 as in the aforementioned schemes). This step is thus defined with kernels of size $(2n+1) \times (2n+1)$, $n \in \{1, 2, \ldots, N\}$, where N is a parameter of the approach.

The horizontal gradient variations are thus captured thanks to $(2n+1) \times (2n+1)$ square kernels

$$Ky''_{x^2} = \begin{pmatrix} 0 & \ldots & 0 \\ \vdots & & \vdots \\ 0 & \ldots & 0 \\ \frac{1}{2n} & 0\ldots0 \, -\frac{2}{2n} \, 0\ldots0 & \frac{1}{2n} \\ 0 & \ldots & 0 \\ \vdots & & \vdots \\ 0 & \ldots & 0 \end{pmatrix}$$

When the convolution product is applied on a $(2n+1) \times (2n+1)$ window, the result is $\frac{1}{2}\left(\frac{P(0,n) - P(0,0)}{n} - \frac{P(0,0) - P(0,-n)}{n}\right)$, which is indeed the variation between the gradient around the central pixel. This proves that this calculus is a correct approximation of $\frac{\partial^2 P}{\partial x^2}$.

When n is 1, this kernel is a centered version of the horizontal Intermediate Difference kernel Ki''_{x^2} modulo a multiplication by $1/2$. When n is 2, this kernel is equal to Kc''_{x^2}.

The vertical gradient variations are again obtained by applying a $\pi/2$ rotation to each horizontal kernel Ky''_{x^2}.

The diagonal gradient variations are obtained thanks to the $(2n+1) \times (2n+1)$ square kernels Ky''_{xy} defined by

$$Ky''_{xy} = \frac{1}{4} \left(\begin{array}{ccccccccc} \frac{1}{n^2} & \ldots & \frac{1}{2n} & \frac{1}{n} & 0 & -\frac{1}{n} & -\frac{1}{2n} & \ldots & -\frac{1}{n^2} \\ \vdots & 0 & & & \ldots & & & 0 & \vdots \\ \frac{1}{2n} & 0 & & & \ldots & & & 0 & -\frac{1}{2n} \\ \frac{1}{n} & 0 & & & \ldots & & & 0 & -\frac{1}{n} \\ 0 & & & & \ldots & & & & 0 \\ -\frac{1}{n} & 0 & & & \ldots & & & 0 & \frac{1}{n} \\ -\frac{1}{2n} & 0 & & & \ldots & & & 0 & \frac{1}{2n} \\ \vdots & 0 & & & \ldots & & & 0 & \vdots \\ -\frac{1}{n^2} & \ldots & -\frac{1}{2n} & -\frac{1}{n} & 0 & \frac{1}{n} & \frac{1}{2n} & \ldots & \frac{1}{n^2} \end{array} \right).$$

When n is 1, Ky''_{xy} is equal to the kernel Kc''_{xy}, and the average vertical variations of the horizontal variations are

$$\begin{aligned}
&\frac{1}{4}\,[((P(0,1)-P(0,0))-(P(1,1)-P(1,0)))+\\
&\quad((P(-1,1)-P(-1,0))-(P(0,1)-P(0,0)))+\\
&\quad((P(0,0)-P(0,-1))-(P(1,0)-P(1,-1)))+\\
&\quad((P(-1,0)-P(-1,-1))-(P(0,0)-P(0,-1)))]\\
&=\frac{1}{4}\,[P(1,-1)-P(1,1)-P(-1,-1)+P(-1,1)]\,.
\end{aligned}$$

which is Ky''_{xy}.

Let us now consider any number n, $1 \leq n \leq N$. Let us first investigate the vertical variations related to the horizontal vector $\overrightarrow{P_{0,0}P_{0,1}}$ (respectively $\overrightarrow{P_{0,-1}P_{0,0}}$) of length 1 that starts from (resp. that points to) $(0,0)$. Like in the case $n = 1$, there are 2 new vectors of length 1, namely $\overrightarrow{P_{n,0}P_{n,1}}$ and $\overrightarrow{P_{-n,0}P_{-n,1}}$ (resp. $\overrightarrow{P_{n,-1}P_{n,0}}$, and $\overrightarrow{P_{-n,-1}P_{-n,0}}$), that are vertically aligned with $\overrightarrow{P_{0,0}P_{0,1}}$ (resp. with $\overrightarrow{P_{0,-1}P_{0,0}}$).

The vertical variation is now equal to n. Following the case where n is 1 to compute the average variation, the coefficients of the first and last line around the central vertical line are thus from left to right: $\frac{1}{4n}$, $\frac{-1}{4n}$, $\frac{-1}{4n}$, and $\frac{1}{4n}$.

Cases are similar with vectors $\overrightarrow{P_{0,0}P_{0,1}}, \ldots \overrightarrow{P_{0,0}P_{0,n}}$ which respectively lead to coefficients $-\frac{1}{4\times 2n}, \ldots, -\frac{1}{4\times n.n}$ (the proof is omitted). Finally, let us consider the vector $\overrightarrow{P_{0,0}P_{0,1}}$ and its vertical variations when δy is $n-1$. As in the case where $n = 1$, we thus obtain the coefficients $\frac{1}{4\times(n-1)n}$ and $-\frac{1}{4\times(n-1)n}$ (resp. $-\frac{1}{4\times(n-1)n}$ and $\frac{1}{4\times(n-1)n}$) in the second line (resp. in the penultimate line) since the vector has length n and δy is $n-1$. Coefficient in the other lines are similarly obtained and the proof is thus omitted.

We are then left to compute an approximation of the partial second order derivatives $\frac{\partial^2 P}{\partial x^2}$, $\frac{\partial^2 P}{\partial y^2}$, and $\frac{\partial^2 P}{\partial x \partial y}$ with the kernels, Ky''_{x^2}, Ky''_{y^2}, and Ky''_{xy} respectively. However, the size of each of these kernels is varying from 3×3 to $(2N+1)\times(2N+1)$. Let us explain the approach on the former partial derivative. The other can be immediately deduced.

Since the objective is to detect large variations, the second order derivative is approximated as the maximum of the approximations. More formally, let n, $1 \leq n \leq N$, be an integer number and $\frac{\partial^2 P}{\partial x^2}_n$ be the result of applying the Kernel Ky''_{x^2} of size $(2n+1)\times(2n+1)$. The derivative $\frac{\partial^2 P}{\partial x^2}$ is defined by

$$\frac{\partial^2 P}{\partial x^2} = \max\left\{\left|\frac{\partial^2 P}{\partial x^2}_1\right|, \ldots, \left|\frac{\partial^2 P}{\partial x^2}_N\right|\right\}. \tag{4.1}$$

The same iterative approach is applied to compute approximations of $\frac{\partial^2 P}{\partial y \partial x}$ and of $\frac{\partial^2 P}{\partial y^2}$. Next section studies the suitability of approximating second order derivatives when considering an image as a polynomial.

4.4/ Polynomial Interpolation of Images for Hessian Matrix Computation

Let $P(x,y)$ be the discrete value of the pixel (x,y) in the image. Let n, $1 \leq n \leq N$, be an integer such that the objective is to find a polynomial interpolation on the $(2n+1)\times(2n+1)$ window where the central pixel has index $(0,0)$. There exists an unique polynomial $L : \mathbb{R}\times\mathbb{R} \to \mathbb{R}$ of degree $(2n+1)\times(2n+1)$ defined such that $L(x,y) = P(x,y)$ for each pixel (x,y) in this window. Such a polynomial is defined by

$$\begin{aligned} L(x,y) = {} & \textstyle\sum_{i=-n}^{n} \sum_{j=-n}^{n} \\ & P(i,j)\left(\prod_{\substack{-n\leq j'\leq n\\ j'\neq j}} \frac{x-j'}{i-j'}\right)\left(\prod_{\substack{-n\leq i'\leq n\\ i'\neq i}} \frac{x-i'}{i-i'}\right) \end{aligned} \tag{4.2}$$

It is not hard to prove that the first order horizontal derivative of the polynomial $L(x,y)$ is

$$\begin{aligned} \frac{\partial L}{\partial x} = {} & \textstyle\sum_{i=-n}^{n} \sum_{j=-n}^{n} P(i,j)\left(\prod_{\substack{-n\leq j'\leq n\\ j'\neq j}} \frac{y-j'}{j-j'}\right) \\ & \left(\sum_{\substack{-n\leq i'\leq n\\ i'\neq i}} \frac{1}{i-i'} \prod_{\substack{-n\leq i''\leq n\\ i''\neq i,i'}} \frac{x-i''}{i-i''}\right) \end{aligned} \tag{4.3}$$

and thus to deduce that the second order ones are

$$\begin{aligned} \frac{\partial^2 L}{\partial x^2} = {} & \textstyle\sum_{i=-n}^{n} \sum_{j=-n}^{n} P(i,j)\left(\prod_{\substack{-n\leq j'\leq n\\ j'\neq j}} \frac{y-j'}{j-j'}\right) \\ & \left(\sum_{\substack{-n\leq i'\leq n\\ i'\neq i}} \frac{1}{i-i'} \sum_{\substack{-n\leq i''\leq n\\ i''\neq i,i'}} \frac{1}{i-i''} \prod_{\substack{-n\leq i'''\leq n\\ i'''\neq i,i',i''}} \frac{x-i'''}{i-i'''}\right) \end{aligned} \tag{4.4}$$

$$\begin{aligned} \frac{\partial^2 L}{\partial y \partial x} = {} & \textstyle\sum_{i=-n}^{n} P(i,j) \\ & \left(\sum_{\substack{-n\leq j'\leq n\\ j'\neq j}} \frac{1}{j-j'} \prod_{\substack{-n\leq j''\leq n\\ j''\neq j,j'}} \frac{y-j''}{j-j''}\right) \\ & \left(\sum_{\substack{-n\leq i'\leq n\\ i'\neq i}} \frac{1}{i-i'} \prod_{\substack{-n\leq i''\leq n\\ i''\neq i,i'}} \frac{x-i''}{i-i''}\right) \end{aligned} \tag{4.5}$$

These second order derivatives are computed for each moving window and are associated to the central pixel, *i.e.*, to the pixel $(0,0)$ inside this one.

Let us first simplify $\frac{\partial^2 L}{\partial x^2}$, defined in Equation (4.4), and when $(x,y) = (0,0)$. If j is not null, the index j' is going to be null and the product $\left(\prod_{\substack{-n\leq j'\leq n\\ j'\neq j}} \frac{-j'}{j-j'}\right)$ is null too. In this equation, we thus only consider $j = 0$. It is obvious that the product indexed with j' is thus equal to 1. This equation can thus be simplified in:

$$\begin{aligned} \frac{\partial^2 L}{\partial x^2} = {} & \textstyle\sum_{i=-n}^{n} P(i,0) \\ & \left(\sum_{\substack{-n\leq i'\leq n\\ i'\neq i}} \frac{1}{i-i'} \sum_{\substack{-n\leq i''\leq n\\ i''\neq i,i'}} \frac{1}{i-i''} \prod_{\substack{-n\leq i'''\leq n\\ i'''\neq i,i',i''}} \frac{i'''}{i'''-i}\right) \end{aligned} \tag{4.6}$$

and then in:

$$\frac{\partial^2 L}{\partial x^2} = \sum_{i=-n}^{n} P(i,0) \left(\sum_{\substack{-n \leq i' < i'' \leq n \\ i', i'' \neq i}} \frac{2}{(i-i')(i-i'')} \prod_{\substack{-n \leq i''' \leq n \\ i''' \neq i, i', i''}} \frac{i'''}{i'''-i} \right). \tag{4.7}$$

From this equation, the kernel allowing to evaluate horizontal second order derivatives can be computed for any n. It is further denoted as Ko''_{x^2}. Instances of such matrix when $n = 2$, 3, and 4 are given in Table 4.3.

Table 4.3: Kernels Ko''_{x^2} for second order horizontal derivatives induced by polynomial interpolation

n	Ko''_{x^2}
2	$\left[\frac{-1}{12}, \frac{4}{3}, \frac{-5}{2}, \frac{4}{3}, \frac{-1}{12}\right]$
3	$\left[\frac{1}{90}, \frac{-3}{20}, \frac{3}{2}, \frac{-49}{18}, \frac{3}{2}, \frac{-3}{20}, \frac{1}{90}\right]$
4	$\left[\frac{-1}{560}, \frac{8}{315}, \frac{-1}{5}, \frac{8}{5}, \frac{-205}{72}, \frac{8}{5}, \frac{-1}{5}, \frac{8}{315}, \frac{-1}{560}\right]$

Table 4.4: Kernels for second order diagonal derivatives induced by polynomial interpolation

n	Ko''_{xy}
2	$\begin{bmatrix} \frac{1}{4} & 0 & \frac{-1}{4} \\ 0 & 0 & 0 \\ \frac{-1}{4} & 0 & \frac{1}{4} \end{bmatrix}$
3	$\begin{bmatrix} \frac{1}{144} & \frac{-1}{18} & 0 & \frac{1}{18} & \frac{-1}{144} \\ \frac{-1}{18} & \frac{4}{9} & 0 & \frac{-4}{9} & \frac{1}{18} \\ 0 & 0 & 0 & 0 & 0 \\ \frac{1}{18} & \frac{-4}{9} & 0 & \frac{4}{9} & \frac{-1}{18} \\ \frac{-1}{144} & \frac{1}{18} & 0 & \frac{-1}{18} & \frac{1}{144} \end{bmatrix}$

From Equation 4.5, kernels allowing to evaluate diagonal second order derivatives (*i.e.*, $\frac{\partial^2 L}{\partial y \partial x}$) are computed. They are denoted as Ko''_{xy}. Table 4.4 gives two examples of them when $n = 1$ and $n = 2$. Notice that for $n = 1$, the kernel Ko''_{xy} is equal to Kc''_{xy}.

4.5/ Distortion Cost

The distortion function has to associate to each pixel (i, j) the cost ρ_{ij} of its modification by ± 1.

The objective is to map a small value to a pixel when all its second order derivatives are high and a large value otherwise. In WOW and UNIWARD the distortion function is based on the Hölder norm with

$$\rho^{w}_{ij} = \left(\left|\xi^{h}_{ij}\right|^{p} + \left|\xi^{v}_{ij}\right|^{p} + \left|\xi^{d}_{ij}\right|^{p} \right)^{-\frac{1}{p}}$$

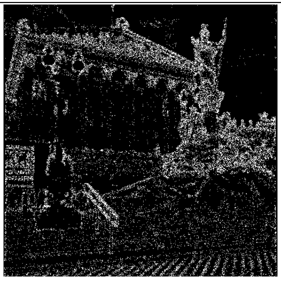
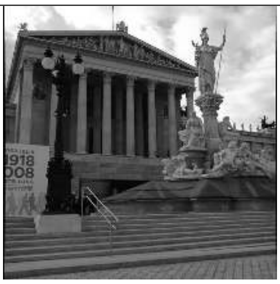
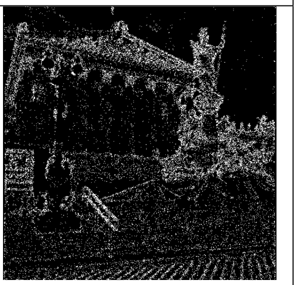
Scheme	Stego. content	Changes with cover
Ky based approach		
Ko based approach		

Fig. 4.1: Embedding changes instance with payload $\alpha = 0.4$

where p is a negative number and ξ^h_{ij} (resp. ξ^v_{ij} and ξ^d_{ij}) represents the horizontal (resp. vertical and diagonal) suitability. A small suitability in one direction means an inaccurate position to embed a message.

We propose here to adapt such a distortion cost as follows:

$$\rho_{ij} = \left(\left| \frac{\partial^2 P}{\partial x^2}(i,j) \right| + \left| \frac{\partial^2 P}{\partial y^2}(i,j) \right| + \left| \frac{\partial^2 P}{\partial y \partial x}(i,j) \right| \right)^{-\frac{1}{p}}$$

It is not hard to check that such a function has large values when at least one of its derivatives is null. Otherwise, the larger the derivatives are, the smaller the returned value is.

4.6/ Experiments

First of all, the whole steganographic approach code is available online[1].

Figure 4.1 presents the results of embedding data in a cover image from the BOSS contest database [75] with respect to the two second order derivative schemes presented in this work. The Ky based approach (resp. the Ko based one) corresponds to the scheme detailed in Section 4.3 (resp. in Section 4.4). The payload α is set to 0.4 and kernels are computed with $N = 4$. The central column outputs the embedding result whereas the right

[1] https://github.com/stego-content/SOS

one displays differences between the cover image and the stego one. It can be observed that pixels in smooth area (the sky, the external access steps) and pixels in clean edges (the columns, the step borders) are not modified by the approach. On the contrary, an unpredictable area (a monument for example) concentrates pixel changes.

4.6.1/ Choice of parameters

The two methods proposed in Section 4.3 and in Section 4.4 are based on kernels of size up to $(2N + 1) \times (2N + 1)$. This section aims at finding the value of the N parameter that maximizes the security level. For each approach, we have built 1,000 stego images with $N = 2, 4, 6, 8, 10, 12,$ and 14 where the covers belong to the BOSS contest database. This set contains 10,000 grayscale 512×512 images in a RAW format. The security of the approach has been evaluated thanks to the Ensemble Classifier [57] based steganalyser, which is considered as a state of the art steganalyser tool. This steganalysis process embeds the rich model (SRM) features [30] of size 34,671. For a payload α, either equal to 0.1 or to 0.4, average testing errors (expressed in percentages) have been studied and are summarized in Table 4.5. Thanks to these experiments, we observe that the size $N = 4$ (respectively $N = 12$) obtains sufficiently large average testing errors for the Ky based approach (resp. for the Ko based one). In what follows, these values are retained for these two methods.

Table 4.5: Average Testing Errors with respect to the the Kernel Size

	α	N						
		2	4	6	8	10	12	14
Average testing error for Kernel K_y	0.1	39	40.2	39.7	39.8	40.1	39.9	39.8
	0.4	15	18.8	19.1	19.0	18.6	18.7	18.7
Average testing error for Kernel K_o	0.1	35.2	36.6	36.7	36.6	37.1	37.2	37.2
	0.4	5.2	6.8	7.5	7.9	8.1	8.2	7.6

4.6.2/ Security Evaluation

As in the previous section, the BOSS contest database has been retained. To achieve a complete comparison with other steganographic tools, the whole database of 10,000 images has been used. Ensemble Classifier with SRM features is again used to evaluate the security of the approach.

We have chosen 4 different payloads, 0.1, 0.2, 0.3, and 0.4, as in many steganographic evaluations. Three values are systematically given for each experiment: the area under the ROC curve (AUC), the average testing error (ATE), and the OOB error (OOB).

All the results are summarized in Table 4.6. Let us analyse these experimental results. The security approach is often lower than those observed with state of the art tools: for instance with payload $\alpha = 0.1$, the most secure approach is WOW with an average testing error equal to 0.43 whereas our approach reaches 0.38. However these results are promising and for two reasons. First, our approaches give more resistance towards

Ensemble Classifier (contrary to HUGO) for large payloads. Secondly, without any optimisation, our approach is not so far from state of the art steganographic tools. Finally, we explain the lack of security of the Ko based approach with large payloads as follows: second order derivatives are indeed directly extracted from polynomial interpolation. This easy construction however induces large variations between the polynomial L and the pixel function P.

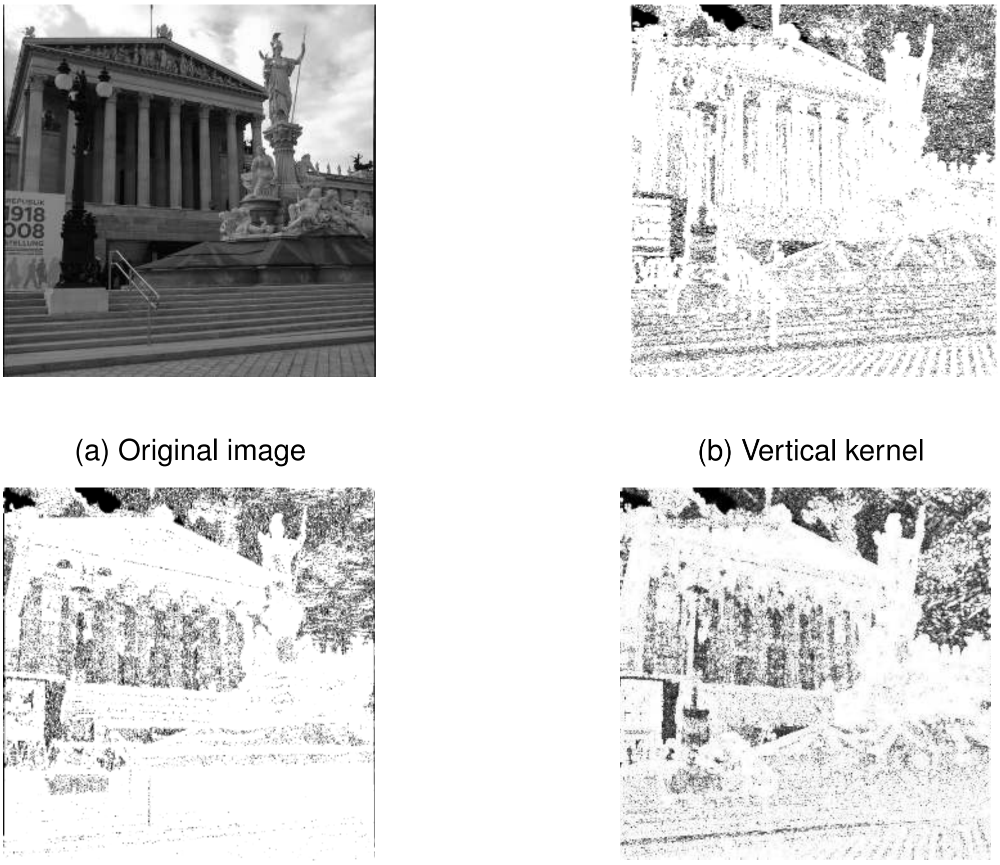
Table 4.6: Summary of experiments

	Payload	AUC	ATE	OOB
WOW	0.1	0.6501	0.4304	0.3974
	0.2	0.7583	0.3613	0.3169
	0.3	0.8355	0.2982	0.2488
	0.4	0.8876	0.2449	0.1978
SUNIWARD	0.1	0.6542	0.4212	0.3972
	0.2	0.7607	0.3493	0.3170
	0.3	0.8390	0.2863	0.2511
	0.4	0.8916	0.2319	0.1977
MVG	0.1	0.6340	0.4310	0.4124
	0.2	0.7271	0.3726	0.3399
	0.3	0.7962	0.3185	0.2858
	0.4	0.8486	0.2719	0.2353
HUGO	0.1	0.6967	0.3982	0.3626
	0.2	0.8012	0.3197	0.2847
	0.3	0.8720	0.2557	0.2212
	0.4	0.9517	0.1472	0.1230
Ky based approach	0.1	0.7378	0.3768	0.3306
	0.2	0.8568	0.2839	0.2408
	0.3	0.9176	0.2156	0.1710
	0.4	0.9473	0.1638	0.1324
Ko based approach	0.1	0.6831	0.3696	0.3450
	0.2	0.8524	0.1302	0.2408
	0.3	0.9132	0.1023	0.1045
	0.4	0.9890	0.0880	0.0570

4.7/ THRESHOLD CHOICE

As we know, an edge is represented by a local maximum in gradient values (*i.e.*, first derivatives) and by a zero crossing in second derivatives. Edge points are detected by finding the zero crossings of the second derivative of the image intensity, but this step is very sensitive to noise. To counter this sensitivity in edge detection, it is desirable to filter noise before edge detection. Threshold technique can be used to solve this problem by focusing on regions of interest. The threshold can perform operation on the gradient magnitudes and output a binary image, which is a matrix of Boolean values, to determine the edge. Removing noise is relevant when detecting the edges in an image. Indeed, well-known edge detectors like Sobel or Canny filters use smoothing filters for this reason.

Image thresholding is an usual task in the field of computer vision. The objective is here to divide pixels of an image as either dark or light. The pixel is considered as an edge location if $f(x, y)$ exceeds threshold T. Edge detection scheme ignores all edges that are not stronger than threshold. In this experiment the threshold technique is used to choose the most important pixels, *i.e.*, to compute the distortion map. After applying second derivatives on the image, we got three images with horizontal, vertical, and diagonal directions. The three images are depicted in Figure 4.2.



(a) Original image

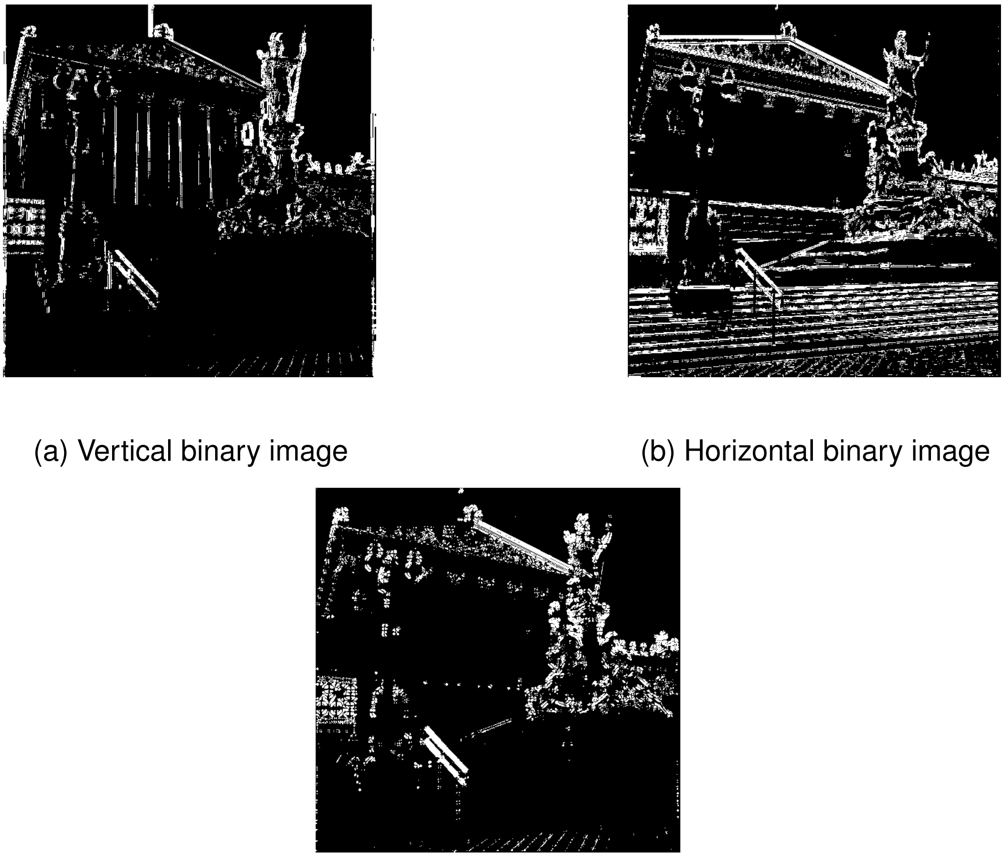
(b) Vertical kernel

(c) Horizontal kernel

(d) Diagonal

Fig. 4.2: The filter result

The three kernels emphasize the vertical, horizontal, and diagonal edges. These edges are affected by the noise and the weak edges. It is thus hard to give correct description of strong edges in each direction. The threshold method converts each kernel results (vertical horizontal and diagonal) into a binary image according to threshold value. This is done by separating the pixels into two regions according to the threshold value, and then by chaining the values of 1 in a the binary image with the original value of the kernel. This is clear in Figure 4.3.



(a) Vertical binary image

(b) Horizontal binary image

(c) Diagonal binary image

Fig. 4.3: Threshold results

The choice of the initial threshold takes into consideration the minimum and the maximum value in each kernel as in Equation 4.8.

$$Initial\_threshold = \frac{min(Im\_value) + max(Im\_value)}{2} \tag{4.8}$$

The next threshold is computed as described in Equation 4.9.

$$Next\_threshold = \frac{mean(Im\_value(Greater\ than\ threshold)) + mean(Im\_value(Less\ than\ threshold))}{2} \tag{4.9}$$

When applying this algorithm, we obtain a binary representation of the image; but, in this case, the payload effect is removed. Algorithms 1 and 2 contain the full description of our technique, which is used to focus on the region of interest when applying the second derivative schemes. The threshold algorithm is applied to each coefficient kernel in the horizontal, vertical, and diagonal direction. This scheme is applied to reduce the number of weak edges and noise in each direction. The threshold process represents a refinement operation to the coefficients in the vertical, horizontal, and diagonal kernels that are produced by the second order derivative. The embedding process selects the largest values in each direction, where we find the edge and noisy area. It preserves the smallest values, that indicate smooth areas in which hidden messages are easy to predict. This formula has been used with 10,000 images from BOSS basis of images.

Algorithm 1: Threshold computing

Input: Image that results from the convolution between the kernel and cover image (Hor-Ver-Diag):$Image\_k$

Output: The image that depends on the threshold value: Im

```
$T \leftarrow 0.5 * [min(Image\_k(:)) + max(Image\_k(:))]$
$Flag = false$
while $\sim Flag$ do
    $g \leftarrow image\_k >= T$
    $Tnext \leftarrow 0.5 * [mean(Image\_k(g)) + mean(Image\_k(\sim g))]$
    $Flag \leftarrow abs(T - Tnext) < 0.5$
    $T = Tnext$;
end
$Im \leftarrow Image\_k >= T$;
$Im\_result = replace(Im, Image\_k)$;
```

Algorithm 2: Replacement function

Input: The binary image outputted by Algorithm 1: Im, $Image\_k$

Output: Image containing values that are grater than the threshold: Im_result

```
for $i \leftarrow 1$ to $size(Im, 1)$ do
    for $i \leftarrow 1$ to $size(Im, 2)$ do
        if $Im(i, j) == 1$ then
            $Im\_result(i, j) = Image\_k(i, j)$;
        end
    end
end
```

Table 4.7 represents the result of the ensemble classifier with 10,000 images. It contains the Area Under the Curve, the Average Testing Error, and the Out Of Bag error related to this experiment. These results are compared with the methods in the state of the art, represented in Table 4.6. We can seen that focusing on regions of interest using second order derivatives gives a result near to the state of the art with a payload of 0.1.

Table 4.7: Summary of threshold experiments

	AUC	ATE	OOB
Threshold experiment	0.6358	0.4360	0.4076

Finally, Figure 4.4 shows the area under the curve after applying the threshold to the horizontal, vertical, and diagonal kernels.

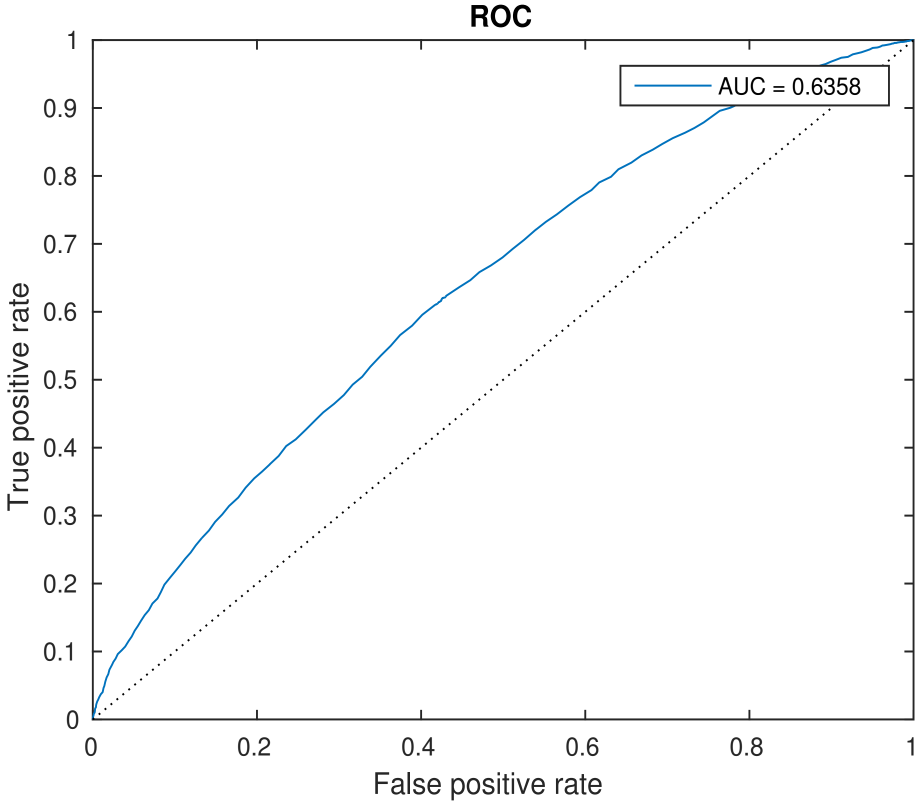


Fig. 4.4: Result of the threshold experiment

4.8/ CONCLUSION

The first contribution of this chapter is a distortion function that is based on second order derivatives. These partial derivatives allow to accurately compute the level curves and thus to look favorably on pixels without clean level curves. Two approaches to build these derivatives have been proposed. The first one is based on revisiting kernels usually embedded in edge detection algorithms. The second one is based on the polynomial approximation of the bitmap image. These two methods have been completely implemented. The first experiments have shown that the security level is slightly lower than the one of the most stringent approaches. These first promising results encourage us to deeply investigate this direction. The last part of this chapter studies the embedding in the chaotic regions. These regions are selected depending on the adaptive threshold technique. The result of the threshold technique gives a reasonable level of security towards ensemble classifier schemes.

III

Conclusion

5

Conclusion and Perspectives

5.1/ Conclusion

In this manuscript, three main contributions related to the information hiding field of research have been realized.

In the first contribution, we have studied various state of the art steganography methods against many factors, to emphasize the gap between laboratory evaluations and the reality. One of the main factors is the payload, on the sensibility of which ensemble classifier was evaluated: nsF5, for instance, shows surprising results. The assessment of Ensemble classifier was done according to changes in feature extraction. In the third parts, we have focused on the group of images that is used during both training and testing stages. For this first contribution, we have seen that the effectiveness of various steganalysis system changes when their parameters are modified. Note that these tests have been performed using Boss images and other ones that were taken from the Internet. It means that standard and normal images were used in these experiments, and so they do not have impacted the differences that have occurred.

In the second contribution, the focus was on the principle of Kerckhoff's. In this experiments, everything about the used steganography schemes without any information about the key that is used in this algorithms is known in the steganalysis system. In this contribution, the steganography method is changed in the training and testing stage but with the same payload in these two stages. From the first results, we observed that the J-UNIWARD and HUGO can be detected when we used the nsF5 in the training stage with the same payload in the two stages.

In the second part of this work, more than one methods are used in the training and the testing stage, but the payload is the same. The methods that are used in these tests are nsF5, J-UNIWARD, and HUGO. All these tests are achieved with the objective to build a universal steganalysis approach. In the third test, we evaluate the effect of changing the payloads in the training stage and the testing stage. These tests were done in two manners. The first one is when the same steganography scheme is used in the training and testing stage but with different payloads. In the second one, the training and the testing stage did not use the same embedding schemes: we changed the payloads and the steganography method. From the result of these experiments, it is clear that steganalysis can detect the presence of hidden messages in spite of the small payloads that that have been considered here. The result of these tests are acceptable when using steganography schemes with appropriate payloads in the training stage.

In the third contribution, a new steganography method has been proposed. This steganography scheme is dependent on level-curves in the image, where the noisy area is more appropriate to embed secret messages. In this work, the search of the level-curves and the way to compute the distortion function depend on the second derivative of the image. Two derivatives were built, the first one revisiting kernels while the second one depends on a polynomial approximation on the image. Boss images were used in this contribution with different payloads. Assessment of the results was done by the Ensemble classifier combined with the SRM features. The second part of this contribution was depending on choosing more convenient regions in an image to embed a secret message. Choosing these areas was achieved using threshold techniques, leading to more acceptable results that are more close to the state of the art in steganography domain.

5.2/ Perspectives

The process of steganography and steganalysis in image domain are always in a state of competition. This led to think about many choices to improve the work. We need first to focus on internal factors, like edges in the image, that have effects on the steganography scheme. These factors help to face steganalysis system. Sharpened to some edges in images before the steganography method may lead to more resistance against steganalysis system.

We plan to more deeply study which of the challenges can be won by steganalysis or steganography. Many other steganography tools, classifiers, and feature extractions libraries will be considered. Finally, a theoretical framework will be proposed to rigorously investigate new steganalysis challenges having steganography and steganalysis as parameters.

We plan to focus on other approaches to provide second order derivatives with larger discrimination power. Then, the objective will be to deeply investigate whether the Holder norm is optimal when the objective is to avoid null second order derivatives, and to give priority to the largest second order values.

The huge computation of the features takes a long time and was made with the help of the “Mésocentre de calcul de l’Université de Franche-Comté”. This reason leads us to think about the field of dimensionality reduction of the features to fastly obtain results that are more accurate to steganalysis systems.

Bibliography

[1] Pinaki Acbarjya, Das Ritaban, and Dibyendu Ghoshal. A study on image edge detection using the gradients. *International Journal of Scientific and Research Publications*, 2(12), 2012.

[2] Monika Agarwal. Text steganographic approaches: a comparison. *arXiv preprint arXiv:1302.2718*, 2013.

[3] Hayat Al-Dmour and Ahmed Al-Ani. A steganography embedding method based on edge identification and xor coding. *Expert systems with Applications*, 46:293–306, 2016.

[4] R Amirtharajan, R Akila, and P Deepikachowdavarapu. A comparative analysis of image steganography. *International journal of computer applications*, 2(3):41–47, 2010.

[5] İsmail Avcıbaş, Mehdi Kharrazi, Nasir Memon, and Bülent Sankur. Image steganalysis with binary similarity measures. *EURASIP Journal on Advances in Signal Processing*, 2005(17):1–9, 2005.

[6] Pedram Azad. Fundamentals of image processing. In *Visual Perception for Manipulation and Imitation in Humanoid Robots*, pages 67–89. Springer, 2009.

[7] Leah Bar, Nir Sochen, and Nahum Kiryati. Image deblurring in the presence of salt-and-pepper noise. In *International Conference on Scale-Space Theories in Computer Vision*, pages 107–118. Springer, 2005.

[8] Johann Barbier, Éric Filiol, and Kichenakoumar Mayoura. Universal jpeg steganalysis in the compressed frequency domain. In *International Workshop on Digital Watermarking*, pages 253–267. Springer, 2006.

[9] SS Bedi and Rati Khandelwal. Various image enhancement techniques-a critical review. *International Journal of Advanced Research in Computer and Communication Engineering*, 2(3), 2013.

[10] Souvik Bhattacharyya. A survey of steganography and steganalysis technique in image, text, audio and video as cover carrier. *Journal of global research in computer science*, 2(4), 2011.

[11] Ahmad Bitar, Rony Darazi, Jean-François Couchot, and Raphaël Couturier. Blind digital watermarking in pdf documents using spread transform dither modulation. *Multimedia Tools and Applications*, *(*):***–***, 2015. accepted on 2015/10/21. To appear.

[12] Om Pavithra Bonam and Sridhar Godavarthy. Edge detection. *Computer Vision (CAP 6415: Project 2)(University of South Florida, 2003)*, 2003.

[13] Ajay Kumar Boyat and Brijendra Kumar Joshi. A review paper: Noise models in digital image processing. *arXiv preprint arXiv:1505.03489*, 2015.

[14] Yambem Jina Chanu, Kh Manglem Singh, and Themrichon Tuithung. Image steganography and steganalysis: A survey. *International Journal of Computer Applications*, 52(2), 2012.

[15] Abbas Cheddad, Joan Condell, Kevin Curran, and Paul Mc Kevitt. Digital image steganography: Survey and analysis of current methods. *Signal processing*, 90(3):727–752, 2010.

[16] Abbas Cheddad, Joan Condell, Kevin Curran, and Paul Mc Kevitt. Digital image steganography: Survey and analysis of current methods. *Signal processing*, 90(3):727–752, 2010.

[17] Rita Chhikara and Latika Singh. A review on digital image steganalysis techniques categorised by features extracted. *image*, 3(4), 2013.

[18] Jean-François Couchot, Raphael Couturier, and Christophe Guyeux. Stabylo: steganography with adaptive, bbs, and binary embedding at low cost. *annals of telecommunications-annales des télécommunications*, 70(9-10):441–449, 2015.

[19] Tomas Denemark, Vahid Sedighi, Vojtech Holub, Rémi Cogranne, and Jessica Fridrich. Selection-channel-aware rich model for steganalysis of digital images. In *2014 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 48–53. IEEE, 2014.

[20] Kshetrimayum Jenita Devi. *A Sesure Image Steganography Using LSB Technique and Pseudo Random Encoding Technique*. PhD thesis, National Institute of Technology-Rourkela, 2013.

[21] Nameer N. El-Emam and Mofleh Al-Diabat. A novel algorithm for colour image steganography using a new intelligent technique based on three phases. *Appl. Soft Comput.*, 37:830–846, 2015.

[22] Vincenzo Ficarra, Giacomo Novara, Walter Artibani, Andrea Cestari, Antonio Galfano, Markus Graefen, Giorgio Guazzoni, Bertrand Guillonneau, Mani Menon, Francesco Montorsi, et al. Retropubic, laparoscopic, and robot-assisted radical prostatectomy: a systematic review and cumulative analysis of comparative studies. *European urology*, 55(5):1037–1063, 2009.

[23] Tomas Filler and Jessica Fridrich. Gibbs construction in steganography. *Information Forensics and Security, IEEE Transactions on*, 5(4):705–720, 2010.

[24] Jessica Fridrich. Feature-based steganalysis for jpeg images and its implications for future design of steganographic schemes. In *International Workshop on Information Hiding*, pages 67–81. Springer, 2004.

[25] Jessica Fridrich, Miroslav Goljan, and Rui Du. Reliable detection of lsb steganography in color and grayscale images. In *Proceedings of the 2001 workshop on Multimedia and security: new challenges*, pages 27–30. ACM, 2001.

[26] Jessica Fridrich, Miroslav Goljan, and Dorin Hogea. Attacking the outguess. In *Proceedings of the ACM Workshop on Multimedia and Security*, volume 2002. Juan-les-Pins, France, 2002.

[27] Jessica Fridrich and Jan Kodovsky. Rich models for steganalysis of digital images. *Information Forensics and Security, IEEE Transactions on*, 7(3):868–882, 2012.

[28] Jessica Fridrich and Jan Kodovsky. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3):868–882, 2012.

[29] Jessica Fridrich, Tomáš Pevnỳ, and Jan Kodovskỳ. Statistically undetectable jpeg steganography: dead ends challenges, and opportunities. In *Proceedings of the 9th workshop on Multimedia & security*, pages 3–14. ACM, 2007.

[30] Jessica J. Fridrich and Jan Kodovský. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3):868–882, 2012.

[31] Jessica J Fridrich and Jan Kodovskỳ. Multivariate gaussian model for designing additive distortion for steganography. In *ICASSP*, pages 2949–2953, 2013.

[32] Mukesh Garg and AP Gurudev Jangra. An overview of different type of data hiding scheme in image using steganographic techniques. *International Journal of Advanced Research in Computer Science and Software Engineering*, 4(1), 2014.

[33] Tina Gebreyohannes and Dong-Yoon Kim. Adaptive noise reduction scheme for salt and pepper. *arXiv preprint arXiv:1201.2050*, 2012.

[34] Rafael C Gonzalez and Richard E Woods. Digital image processing. *Nueva Jersey*, 2008.

[35] Nagham Hamid, Abid Yahya, R Badlishah Ahmad, and Osamah M Al-Qershi. Image steganography techniques: an overview. *International Journal of Computer Science and Security (IJCSS)*, 6(3):168–187, 2012.

[36] Ali K Hmood, Hamid A Jalab, ZM Kasirun, BB Zaidan, and AA Zaidan. On the capacity and security of steganography approaches: An overview. *Journal of Applied Sciences*, 10:1825–1833, 2010.

[37] Vojtěch Holub and Jessica Fridrich. Low-complexity features for jpeg steganalysis using undecimated dct. *IEEE Transactions on Information Forensics and Security*, 10(2):219–228, 2015.

[38] Vojtěch Holub, Jessica Fridrich, and Tomáš Denemark. Random projections of residuals as an alternative to co-occurrences in steganalysis. In *IS&T/SPIE Electronic Imaging*, pages 86650L–86650L. International Society for Optics and Photonics, 2013.

[39] Vojtěch Holub, Jessica Fridrich, and Tomáš Denemark. Universal distortion function for steganography in an arbitrary domain. *EURASIP Journal on Information Security*, 2014(1):1–13, 2014.

[40] Vojtěch Holub, Jessica Fridrich, and Tomáš Denemark. Universal distortion function for steganography in an arbitrary domain. *EURASIP Journal on Information Security*, 2014(1):1–13, 2014.

[41] Vojtéch Holub and Jessica J Fridrich. Designing steganographic distortion using directional filters. In *WIFS*, pages 234–239, 2012.

[42] J Hossain. Information-hiding using image steganography with pseudorandom permutation. *Bangladesh Research Publications Journal, ISSN*, 2003:215–225, 1998.

[43] Donghui HU, Lina WANG, Xiaqiu JIANG, Dengpan YE, and Shiguo LIAN. A specific steganalysis approach to effective attacking the mb1. *Chinese Journal of Electronics*, 18(4), 2009.

[44] Md. Rafiqul Islam, A. W. Naji, A. A. Zaidan, and B. B. Zaidan. New system for secure cover file of hidden data in the image page within executable file using statistical steganography techniques. *CoRR*, abs/1002.2416, 2010.

[45] Saiful Islam, Mangat Rai Modi, and Phalguni Gupta. Edge-based image steganography. *EURASIP J. Information Security*, 2014:8, 2014.

[46] Matúš Jókay and Tomáš Moravčík. Image-based jpeg steganography. *Tatra Mountains Mathematical Publications*, 45(1):65–74, 2010.

[47] Mamta Juneja, Parvinder S Sandhu, and Ekta Walia. Application of lsb based steganographic technique for 8-bit color images. *World Academy of Science, Engineering and Technology*, 50:423–425, 2009.

[48] Geeta Kasana, Kulbir Singh, and Satvinder Singh Bhatia. Block-based high capacity multilevel image steganography. *Journal of Circuits, Systems, and Computers*, 25(8):1–21, 2016.

[49] Gurmeet Kaur and Aarti Kochhar. Transform domain analysis of image steganography. *International Journal for Science and Emerging Technologies with Latest Trends" 6 (1)*, pages 29–37, 2013.

[50] Stephen Mark Keating. Digital image enhancement, June 6 2000. US Patent 6,072,538.

[51] Oleg Kobylin and Vyacheslav Lyashenko. Comparison of standard image edge detection techniques and of method based on wavelet transform. *International Journal of Advanced Research*, 2(8):572–580, 2014.

[52] Jan Kodovskỳ. *Steganalysis of Digital Images Using Rich Image Representations and Ensemble Classifiers*. PhD thesis, State University of New York, 2012.

[53] Jan Kodovskỳ and Jessica Fridrich. Calibration revisited. In *Proceedings of the 11th ACM workshop on Multimedia and security*, pages 63–74. ACM, 2009.

[54] Jan Kodovskỳ and Jessica Fridrich. Steganalysis of jpeg images using rich models. In *IS&T/SPIE Electronic Imaging*, pages 83030A–83030A. International Society for Optics and Photonics, 2012.

[55] Jan Kodovsky, Jessica Fridrich, and Vojtěch Holub. Ensemble classifiers for steganalysis of digital media. *IEEE Transactions on Information Forensics and Security*, 7(2):432–444, 2012.

[56] Jan Kodovsky, Jessica Fridrich, and Vojtech Holub. Ensemble classifiers for steganalysis of digital media. *Information Forensics and Security, IEEE Transactions on*, 7(2):432–444, 2012.

[57] Jan Kodovský, Jessica J. Fridrich, and Vojtech Holub. Ensemble classifiers for steganalysis of digital media. *IEEE Transactions on Information Forensics and Security*, 7(2):432–444, 2012.

[58] Jan Kodovskỳ, Tomáš Pevnỳ, and Jessica Fridrich. Modern steganalysis can detect yass. In *IS&T/SPIE Electronic Imaging*, pages 754102–754102. International Society for Optics and Photonics, 2010.

[59] Kanika Lakhani, Bhawna Minocha, and Neeraj Gugnani. Analyzing edge detection techniques for feature extraction in dental radiographs. *Perspectives in Science*, 2016.

[60] Chin-Feng Lee, Yi-Ren Wang, and Chin-Chen Chang. A steganographic method with high embedding capacity by improving exploiting modification direction. In *Intelligent Information Hiding and Multimedia Signal Processing, 2007. IIHMSP 2007. Third International Conference on*, volume 1, pages 497–500. IEEE, 2007.

[61] James K Lein. Fundamentals of image processing. In *Environmental Sensing*, pages 83–109. Springer, 2012.

[62] Bin Li, Junhui He, Jiwu Huang, and Yun Qing Shi. A survey on image steganography and steganalysis. *Journal of Information Hiding and Multimedia Signal Processing*, 2(2):142–172, 2011.

[63] Bo Li, Aleksandar Jevtic, Ulrik Söderström, Shafiq Ur Réhman, and Haibo Li. Fast edge detection by center of mass. In *The 1st IEEE/IIAE International Conference on Intelligent Systems and Image Processing 2013 (ICISIP2013)*, pages 103–110, 2013.

[64] Jiu-fen Liu, Yu-guo Tian, Tao Han, Junchao Wang, and Xiangyang Luo. Stego key searching for LSB steganography on JPEG decompressed image. *SCIENCE CHINA Information Sciences*, 59(3):32105:1–32105:15, 2016.

[65] Weiqi Luo, Fangjun Huang, and Jiwu Huang. Edge adaptive image steganography based on lsb matching revisited. *IEEE Transactions on information forensics and security*, 5(2):201–214, 2010.

[66] Raman Maini and Himanshu Aggarwal. Study and comparison of various image edge detection techniques. *International journal of image processing (IJIP)*, 3(1):1–11, 2009.

[67] Raman Maini and Himanshu Aggarwal. A comprehensive review of image enhancement techniques. *arXiv preprint arXiv:1003.4053*, 2010.

[68] Aziz Makandar and Bhagirathi Halalli. Image enhancement techniques using highpass and lowpass filters. *International Journal of Computer Applications*, 109(14), 2015.

[69] Yoan Miche. *Developing fast machine learning techniques with applications to steganalysis problems*. PhD thesis, Institut National Polytechnique de Grenoble-INPG, 2010.

[70] R Muthukrishnan and M Radha. Edge detection techniques for image segmentation. *International Journal of Computer Science & Information Technology*, 3(6):259, 2011.

[71] Hideki Noda, Michiharu Niimi, and Eiji Kawaguchi. High-performance jpeg steganography using quantization index modulation in dct domain. *Pattern Recognition Letters*, 27(5):455–461, 2006.

[72] Komal Patel, Sumit Utareja, and Hitesh Gupta. Information hiding using least significant bit steganography and blowfish algorithm. *International Journal of Computer Applications*, 63(13), 2013.

[73] Pritesh Pathak and S Selvakumar. Blind image steganalysis of jpeg images using feature extraction through the process of dilation. *Digital Investigation*, 11(1):67–77, 2014.

[74] Tomáš Pevny, Patrick Bas, and Jessica Fridrich. Steganalysis by subtractive pixel adjacency matrix. *IEEE Transactions on Information Forensics and Security*, 5(2):215–224, 2010.

[75] Tomás Pevný, Tomás Filler, and Patrick Bas. Break our steganographic system, 2010. Available at http://www.agents.cz/boss/.

[76] Tomas Pevny and Jessica Fridrich. Merging markov and dct features for multi-class jpeg steganalysis. In *Electronic Imaging 2007*, pages 650503–650503. International Society for Optics and Photonics, 2007.

[77] Ioannis Pitas. *Digital image processing algorithms and applications*. John Wiley & Sons, 2000.

[78] Sobel Prewitt. Scharr gradient 5x5 convolution matrices guennadi (henry) levkine email: hlevkin at yahoo. com vancouver, canada. *First draft, February*, 2011.

[79] Niels Provos. Defending against statistical steganalysis. In *Usenix security symposium*, volume 10, pages 323–336, 2001.

[80] Kazem Qazanfari and Reza Safabakhsh. A new steganography method which preserves histogram: Generalization of lsb++. *Information Sciences*, 277:90–101, 2014.

[81] M Indra Sena Reddy and AP Siva Kumar. Secured data transmission using wavelet based steganography and cryptography by using aes algorithm. *Procedia Computer Science*, 85:62–69, 2016.

[82] Sumanth Sakkara, DH Akkamahadevi, K Somashekar, and K Raghu. Integer wavelet based secret data hiding by selecting variable bit length. *International Journal of Computer Applications*, 48(19):7–11, 2012.

[83] S Abdul Saleem and T Abdul Razak. Survey on color image enhancement techniques using spatial filtering. *International Journal of Computer Applications*, 94(9), 2014.

[84] Phil Sallee. Model-based steganography. In *International workshop on digital watermarking*, pages 154–167. Springer, 2003.

[85] Shubhashree Savant. A review on edge detection techniques for image segmentation. *Int. J. Comp. Sci. Inform. Technol*, 5(4):5898–5900, 2014.

[86] HK Sawant and Mahentra Deore. A comprehensive review of image enhancement techniques. *International Journal of Computer Technology and Electronics Engineering (IJCTEE)*, 1(2):39–44, 2010.

[87] Andre Schmeißer, Raimund Wegener, Dietmar Hietel, and Hans Hagen. Smooth convolution-based distance functions. *Graphical Models*, 82:67–76, 2015.

[88] N Senthilkumaran and R Rajesh. Edge detection techniques for image segmentation–a survey of soft computing approaches. *International journal of recent trends in engineering*, 1(2), 2009.

[89] Mahdi Setayesh, Mengjie Zhang, and Mark Johnston. Detection of continuous, smooth and thin edges in noisy images using constrained particle swarm optimisation. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pages 45–52. ACM, 2011.

[90] Janvi Shah, Nupoor Patel, Hiral Tandel, Neelam Soni, and Ghanshyam I Prajapati. A hybrid approach for edge detection using fuzzy logic and canny method. *International Journal of Engineering Research & Technology (IJERT)*, 2(3), 2013.

[91] Pooja Sharma, Gurpreet Singh, and Amandeep Kaur. Different techniques of edge detection in digital image processing. *International Journal of Engineering research and Applications*, 3(3):458–461, 2013.

[92] Falesh M Shelke, Ashwini A Dongre, and Pravin D Soni. Comparison of different techniques for steganography in images. *International Journal of Application or Innovation in Engineering & Management*, 3(2), 2014.

[93] K Sathish Shet, AR Aswath, et al. Image steganography using integer wavelet transform based on color space approach. In *Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014*, pages 839–848. Springer, 2015.

[94] Kaushal Solanki, Kenneth Sullivan, Upamanyu Madhow, BS Manjunath, and Shivkumar Chandrasekaran. Statistical restoration for robust and secure steganography. In *IEEE International Conference on Image Processing 2005*, volume 2, pages II–1118. IEEE, 2005.

[95] Chris Solomon and Toby Breckon. *Fundamentals of Digital Image Processing: A practical approach with examples in Matlab*. John Wiley & Sons, 2011.

[96] GS Sravanthi, Mrs B Sunitha Devi, SM Riyazoddin, and M Janga Reddy. A spatial domain image steganography technique based on plane bit substitution method. *Global Journal of Computer Science and Technology Graphics & Vision, 12 (15)*, 2012.

[97] Jean-Luc Starck, Fionn D Murtagh, and Albert Bijaoui. *Image processing and data analysis: the multiscale approach*. Cambridge University Press, 1998.

[98] High Capacity Despite Better Steganalysis and Andreas Westfeld. F5—a steganographic algorithm. In *Information Hiding: 4th International Workshop, IH 2001, Pittsburgh, PA, USA, April 25-27, 2001. Proceedings*, volume 2137, page 289. Springer Science & Business Media, 2001.

[99] C. P. Sumathi, T. Santhanam, and G. Umamaheswari. A study of various steganographic techniques used for information hiding. *CoRR*, abs/1401.5561, 2014.

[100] TK Thivakaran and RM Chandrasekaran. Nonlinear filter based image denoising using amf approach. *arXiv preprint arXiv:1003.1803*, 2010.

[101] Scott E Umbaugh. *Computer imaging: digital image analysis and processing*. CRC press, 2005.

[102] D Upham. Steganographic algorithm jsteg. *Software available at http://zooid. org/˜ paul/crypto/jsteg*, 1993.

[103] Rohit Verma and Dr Jahid Ali. A comparative study of various types of image noise and efficient noise removal techniques. *International journal of advanced research in computer science and software engineering*, 3(10):617–622, 2013.

[104] Brian A Wandell, Abbas El Gamal, and Bernd Girod. Common principles of image acquisition systems and biological vision. *Proceedings of the IEEE*, 90(1):5–17, 2002.

[105] Andreas Westfeld and Andreas Pfitzmann. Attacks on steganographic systems. In *International workshop on information hiding*, pages 61–76. Springer, 1999.

[106] Zhihua Xia, Lincong Yang, Xingming Sun, Wei Liang, Decai Sun, and Zhiqiang Ruan. A learning-based steganalytic method against lsb matching steganography. *Radioengineering*, 20(1):102–109, 2011.

[107] Peng Xu and Haisong Xu. Filter selection based on representative training samples for multispectral imaging. *Optik-International Journal for Light and Electron Optics*, 127(20):9743–9754, 2016.

[108] Jing Zheng, Su-ping Peng, and Feng Yang. A novel edge detection for buried target extraction after svd-2d wavelet processing. *Journal of Applied Geophysics*, 106:106–113, 2014.

[109] Han Zong, Fen-lin Liu, and Xiang-yang Luo. Blind image steganalysis based on wavelet coefficient correlation. *Digital Investigation*, 9(1):58–68, 2012.

Document generated with LaTeX and:
the LaTeX style for PhD Thesis created by S. Galland — http://www.multiagent.fr/ThesisStyle
the `tex-upmethodology` package suite — http://www.arakhne.org/tex-upmethodology/

Abstract:

In the recent time, the field of image steganalysis and steganography became more important due to the development in the Internet domain. It is important to keep in mind that the whole process of steganography and steganalysis can be used for legal or illegal operations like any other applications. The work in this thesis can be divided inthree parts. The first one concentrates on parameters that increase the security of steganography methods against steganalysis techniques. In this contribution the effect of the payload, feature extractions, and group of images that are used in the learning stage and testing stage for the steganalysis system are studied. From simulation, we note that the state of the art steganalyzer fails to detect the presence of a secret message when some parameters are changed. In the second part, we study how the presence of many steganography methods may influence the detection of a secret message. The work takes into consideration that there is no ideal situation to embed a secret message when the steganographier can use any scheme with any payloads. In the third part, we propose a method to compute an accurate distortion map depending on a second order derivative of the image. The second order derivative is used to compute the level curve and to embed the message on pixels outside clean level curves. The results of embedding a secret message with our method demonstrate that the result is acceptable according to state of the art steganography.

Keywords: Steganography, Stegananlysis, information security.

Résumé :

De nos jours, le développement de la steganalyse et la stéganographie est incontournable, et peut être utilisé à des fins légales comme illégales, comme dans toute autre application. Le travail présenté dans cette thèse, se concentrant sur ces questions, est divisé en trois parties. La première partie concerne les paramètres permettant d'accroître le niveau de sécurité de la stéganographie afin de faire face aux techniques de steganalyse. La contribution apportée dans cette première partie concerne l'étude de l'effet de la charge utile, l'extraction des caractéristiques, ainsi que le groupe d'images utilisés dans la phase d'apprentissage et la phase de test. Les résultats des simulations montrent que les techniques de steganalyse de l'état de l'art échouent dans la detection des messages secrets intégrés dans les images quand les paramètres changent entre l'apprentissage et le test. Dans la deuxième partie, nous étudions l'impact de la combinaison de plusieurs méthodes stéganographiques sur la détection des messages secrets. Ce travail prend en considération qu'il n'existe pas une procedure idéale, mais que le steganographieur pourra utiliser n'importe quel schéma ainsi que n'importe quel taux d'embarquement. Dans la troisième et dernière partie, on propose une méthode qui calcule une carte de distorsion précise, en fonction de la dérivée seconde de l'image. La dérivée seconde est utilisée afin de calculer les courbes de niveau, ensuite le message va être caché dans l'image en écartant les courbes de niveaux inférieurs à un certain seuil. Les résultats expérimentaux démontrent que le niveau de sécurité est acceptable comparé aux méthodes stéganographiques de l'état de l'art.

Mots-clés : Stéganographie, steganalyse, sécurité informatique.

■ École doctorale SPIM 16 route de Gray F - 25030 Besançon cedex
■ tél. +33 (0)3 81 66 66 02 ■ **ed-spim@univ-fcomte.fr** ■ www.ed-spim.univ-fcomte.fr