STEGANOGRAPHY AND STEGANALYSIS: NEW APPROACHES FROM NATURAL IMAGES

WU SONGTAO

Ph.D

The Hong Kong Polytechnic University

2018

The Hong Kong Polytechnic University
Department of Computing

Steganography and Steganalysis: New Approaches from Natural Images

Wu Songtao

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

November 2016

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____:(Signature)

Wu Songtao :(Name of Student)

Abstract

Recent advances of network technology provide a great convenience for data communication. A key problem of data communication on the Internet is to transmit data from a sender to its receiver safely, without being eavesdropped, illegally accessed or tampered. Steganography, which is the art or science that hides secret message in an appropriate multimedia carrier including text, image, audio, or video [1], provides an effective solution. Unlike cryptography which emphasizes protecting the information security by making messages illegible, steganography intends to conceal the fact that a secret message is being sent and thus will not raise an opponents suspicion. Owing to this benefit, steganography plays a crucial role in many important applications such as military and commercial communications.

In contrast to steganography, steganalysis aims to reveal the presence of secret messages embedded in digital medias [80]. This technique tries to make the steganography disable by determining whether a given carrier signal has hidden message, estimating the amount of hidden message, or, if possible, recovering the hidden message. For this nature, steganalysis is usually used as a measure to evaluate the security performance of steganographic algorithms.

Natural images, which denote various photographs of typical environment we live in, are the most popular image files on the internet. Natural images are highly non-random, showing structural richness and strong local correlations. In this thesis, we focus on improving the performance of steganography and steganalysis by exploring these two properties of natural images. Following this idea, we investigate steganogra-

phy and steganalysis from the following two aspects:

For steganography, we improve its undetectability via selecting suitable natural cover images. Natural images have rich and complex structures, which provide steganographier enough space to hide secret messages. Unlike most existing works focusing on designing data embedding algorithms to preserve the structure of natural images, this work aims to improve the performnace of steganographic algorithms by selecting suitable natural cover images. A novel measure, which is only determined by the probability distribution of images, is proposed to analyze their hiding abilities. Based on statistical models of natural images, we prove that the proposed measure is an upper bound of the Kullback-Leibler (KL) divergence, a theoretical measure of steganographic security, both for spatial domain images and compressed domain images. With the measure, we investigate what properties that intrinsically make the stego images undetectable. Our conclusion is that the undetectability of the stego image relates to three factors: the entropy of the statistical model to represent the image, the energy of varying pixels across the image, and the number of nonzero DCT coefficients to reconstruct the image.

For steganalysis, we improve its detection ability by modeling natural images with Convolutional Neural Networks (CNN). Natural images have strong spatially local correlation. This local correlation is distorted when secret messages are embedded, making it different from the normal correlation in natural images. Due to this fact, we propose to use CNN for image steganalysis. A unified model have been designed from two aspects. For the first, different from existing CNN based steganalytic algorithms that use a predefined highpass kernel to preprocess input images, we integrate the highpass filtering operation into the proposed network by building a content suppression subnetwork. Highpass kernels in this subnetwork are adaptively updated in the network training, allowing more powerful discriminative features come into the subsequent network than that of CNN models with a predefined kernel. For the second, we propose a novel subnetwork to actively preserve and further strengthen the weak stego signal

generated by secret messages based on residual learning, making the whole network capture the difference between cover images and stego images. Theoretically, we prove that the residual learning can preserve the weak stego signal for the deep model with any depths. Extensive experiments demonstrate that the proposed network can detect the state of the art steganography with better accuracy than previous methods when cover images and their stego images are paired in training and testing. We further discuss the proposed method in more general case and analyze the limitation of a CNN model with batch normalization layers for image steganalysis.

Empirical validations have demonstrated that the performance of steganography and steganalysis can be improved with appropriate natural image statistical models. Our future work will focus on two aspects: design advanced steganographic algorithms based on CNN models; develop CNN models without batch normalization layers to detect steganography in more general case and further extend them into the compressed domain image.

Keywords: Steganography, steganalysis, image selection, convolutional neural network.

Publications

Journal Papers

1. Shenghua Zhong, Yan Liu, **Songhtao Wu**, Yang Liu, and Xiapu Luo "What Makes the Stego Image Undetectable ? ", *IEEE Transactions on Image Processing* (Under Review).

2. **SongtaoWu**, Shenghua Zhong, and Yan Liu. "Deep residual network for image steganalysis", *Multimedia Tools and Applications*, doi:10.1007/s11042-017-4440-4, pp.1-17, 2017.

3. Yan Liu, Yang Liu, Shenghua Zhong, and **Songtao Wu**. "Implicit visual learning: image recognition via dissipative learning model", *ACM Transactions on Intelligent Systems and Technology*, 8(2):1-31, 2017.

Conference Papers

1. **Songtao Wu**, Shenghua Zhong, and Yan Liu. "Residual convolution network based steganalysis with adaptive content suppression", *IEEE International Conference on Multimedia and Expo (ICME)*, 2017.

2. **Songtao Wu**, Yan Liu, Shenghua Zhong, Yang Liu, "What Makes the Stego Image Undetectable ? ", in *ACM International Conference on Internet Multimedia Computing and Service*, Hunan, China, August 19-21, 2015.

3. **Songtao Wu**, Shenghua Zhong, and Yan Liu. "Steganalysis via deep residual network", *IEEE International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 1233-1236, 2016.

4. Shenghua Zhong, Yan Liu, Kien A. Hua, **Songtao Wu**. "Is noise always harmful? Visual learning from weakly-related data", *International Conference on Orange Technologies (ICOT)*, 2015.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisor, Dr. Yan Liu, who offered me the opportunity to pursue PhD study in Polyu. Dr. Liu not only tried her best to train me to be qualified researcher, but also provided continuous guidance to make me be a kind person. Her immense knowledge, unique insight and professional supervision gave me the chance to explore high quality research. Her kindness, patience and tolerance encouraged me to move forward when I fell into depression. Undoubtedly, it is impossible to finish this work without her help.

I would like to give special thanks other members of Dr. Liu′s group - Yang Liu and Shenghua Zhong. They helped me a lot during the last three years and their strong will to make every thing better set up good examples for me. It is my fortune to work with them.

Finally, I would like to thank my parents, my wife and my sister. Thank you for your unconditional love and support. Your understanding and trust gave the strength to finish my PhD study.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

1.1 Steganography and Steganalysis: Definition and Theoretical Formulation

Fig.1.1 shows the general idea of image steganography and steganalysis. For steganography, the sender hides the message $\mathbf{m}$ in the cover image X. By applying the message embedding algorithm $Emb(X, \mathbf{m}, k)$ and the key k on X, the stego image Y is generated and then passed to the receiver. By applying the message extraction algorithm $Ext(Y, k)$ and key k on Y, the receiver can recover the secret message $\mathbf{m}$. In the figure, the stego image Y and cover image X denote the images with and without hidden information, respectively. During the communication, the sender and the receiver should pledge that any intended observer in the channel cannot differentiate Y from X. For steganalysis, however, it represents some observers in the communication channel that attempt to discriminate the stego image Y against the cover image X. Further, steganalysis needs to estimate the amount of hidden message or even recover the hidden message. In this thesis, we only consider the basic requirement that a steganalyzer should determine whether an image contains secret messages or not.

Modern researchers formulate the steganography and steganalysis as a prisoner escaping problem [32], which is shown as Fig.1.2. Assume Alice and Bob are two

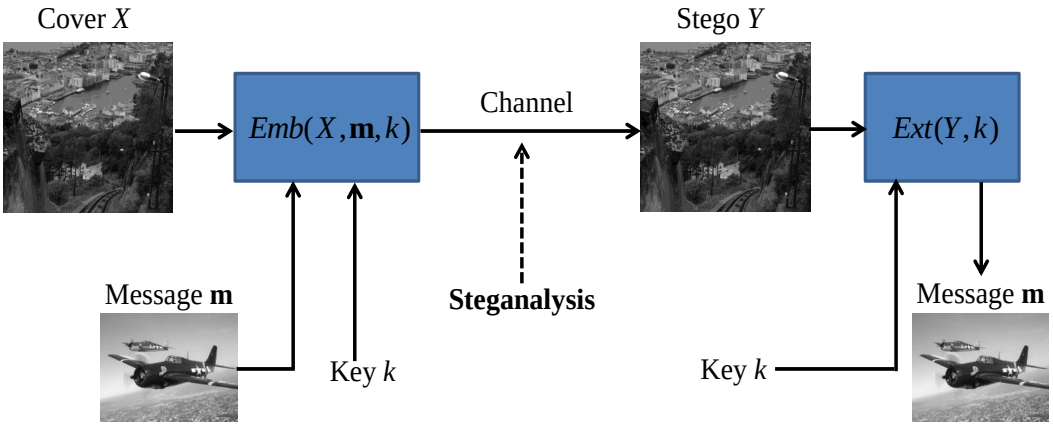


Figure 1.1: Schematic illustration to steganography and steganalysis.

prisoners, they want to find methods to escape the jail. Since Alice and Bob are not in the same room, the only way then can communicate with each other is to write messages in the paper. A policeman named Wendy checks every paper they have written. In order to communicate with each other, Alice and Bob use a key shared by themselves to add secret messages into their paper. Alice and Bob succeed if they can exchange the information and do not arouse Wendy's suspicion. In this model, Alice and Bob play the role of steganographier, while Wendy acts as a steganalyzer.

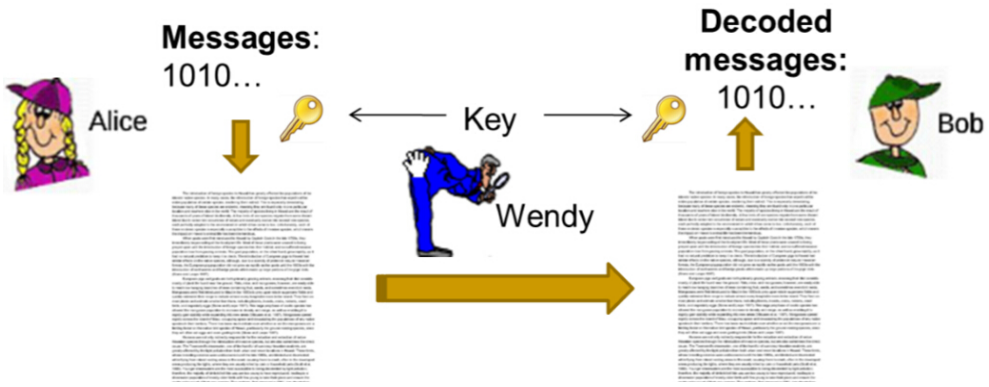


Figure 1.2: Prisoners escaping model of steganography

1.2 Motivation of Addressing Steganography and Steganalysis from Natural Image Structures

The rapid development of social media has resulted in a huge amount of image data in our daily lives. Meeker in her "state of the internet" report indicated that more than 3 billion images are uploaded to Facebook, Instagram, Flickr, Snapchat, and WhatsApp every day [1]. Fig.1.3 shows the total number of uploaded images for 5 hot social network platforms from 2005 to 2015. Facebook in a white paper also revealed that more than 250 billion images have been uploaded by its users [2]. Among all uploaded images, most of them are the natural images, for example the human beings, animals, buildings, environmental scenes, etc. Such huge amount of natural images provide steganographiers and steganalyzers almost infinite materials for applying image steganography and image steganalysis.

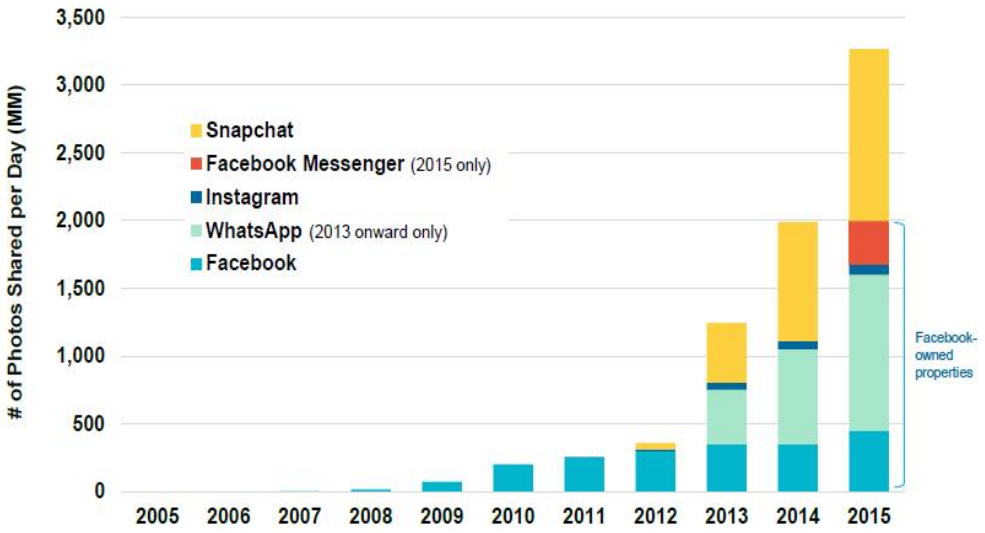


Figure 1.3: Daily Number of Photos Shared on Select Platforms, Global, 2005-2015 [116].

Natural images have their own properties that make them especially suitable for steganography and steganalysis. On one hand, natural images are highly non-random and have rich structures, including edges, textures, points, etc. These rich structures

[1] http://www.kpcb.com/internet-trends
[2] http//www.businessinsider.com/facebook-350-million-photos-each-day-2013-9

are often hard to be accurately modeled, which provides enough space for hiding secret messages. On the other hand, adjacent pixels in natural images are not independent, they have strong spatially local correlations. These local correlations would be changed when secret messages are embedded, making them different from natural ones. This change could provide steganalyzer information to discriminate natural cover images and their stego versions. However, there are few works that use these properties to design steganographic and steganalytic algorithms.

Because of the proliferation of natural images in the internet and their good properties, the primary focus of this thesis is to improve the performance of steganography and steganalysis based on the statistics of natural images. For steganography, we want to investigate what kinds of natural images that are suitable for message hiding and propose to improve the performance of steganography by cover image selection. For steganalysis, we propose to use convolutional neural networks to mine local correlations in natural images, thus improve the detection ability of steganalysis.

1.3 Proposed Framework

In the dissertation, we propose to improve the performance of steganography and steganalysis by mining structures of natural images. Fig.1.4 shows the research framework of our work. The techniques are designed and built from two aspects: 1) selecting suitable natural cover images to improve the undetectability of steganography; 2) modeling natural images with CNN to improve the detection ability of steganalysis:

1.3.1 Selecting Suitable Natural Cover Images to Improve the Undetectability of Steganography

Although extensive efforts focus on designing message embedding algorithms to avoid the stego images being distinguished from normal ones in the research of steganography, what properties that intrinsically determine the hiding ability of an image and

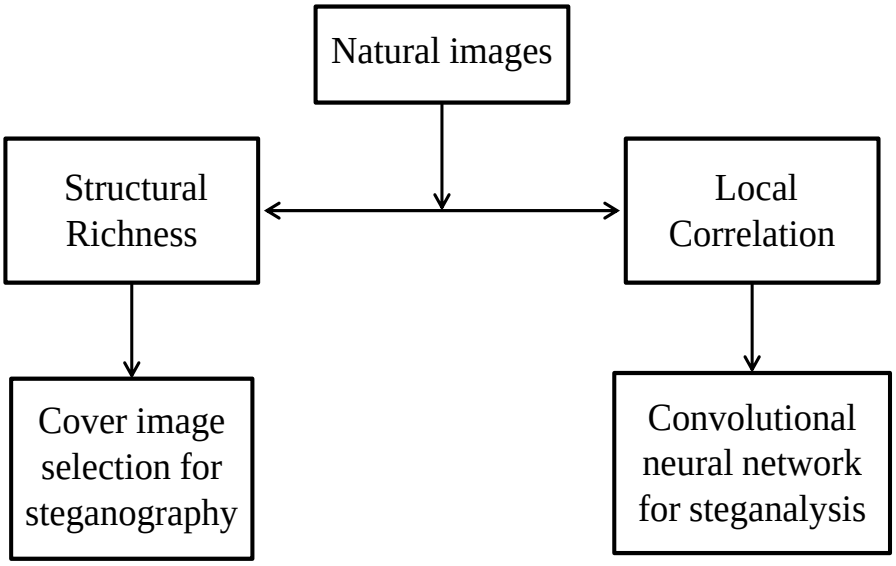


Figure 1.4: Proposed research framework.

make the steganography undetectable remain unclear. To handle the problem, this work proposes a new measure to analyze the hiding ability of cover images. Based on the information theoretic metric for steganography, the KL divergence, we derive the proposed measure between the cover image and the stego image. Unlike some existing measures that depend on the embedding operations or steganalytic methods, the proposed measure is only determined by the probability distribution of natural images. This advantage indicates that the measure is independent of any specific steganographic algorithms and steganalytic techniques. Another feature of the proposed measure is well founded by steganographic theory. We prove that, both for probability distributions of spatial domain images and compressed domain images, the proposed measure could bound the KL divergence. Consequently, the KL divergence is forced to decrease when the proposed measure becomes small, leading to a securer steganography. With the proposed measure, we further analyze the properties of cover images with good hiding ability. Our conclusion is that the security of a cover image relates to three factors: the entropy of the model to represent the image, the energy of varying pixels across the image, and the number of nonzero DCT coefficients to reconstruct the image. These properties enable us to select securer cover images on the internet for steganography.

1.3.2 Modeling Natural Images with CNN to Improve the Detection Ability of Steganalysis

Natural images have strong local correlations among image pixels. These correlations, however, are distorted when secret message are embedded. CNN models are powerful to capture various correlations in natural images. This work proposes to improve the detection ability of steganalysis with CNN. We propose a novel CNN model by designing two newly designed subnetworks: the adaptive content suppression subnetwork and the residual learning subnetwork. For the adaptive content suppression subnetwork, it aims to adaptively reduce the influence of image content and thus increase the Signal-to-Noise Ratio (SNR) between the stego signal generated by message embedding and the noise signal of image content. For the residual learning subnetwork, it is to preserve the stego signal when it propogates in the network, making the whole model capture the difference between cover images and stego images. Theoretical analysis indicates that residual learning can preserve and even improve the discrimination of cover images and stego images for the CNN model with any depths. Experimental results demonstrate that the proposed model can effectively detect the state of the art steganography when cover images and their stego images are paired in training and testing.

1.4 Organization of the Dissertation

The rest of this thesis is organized as follows.

- In Chapter 2, we introduce some background knowledge about steganography and steganalysis, including definitions and techniques. Several theoretical metrics of steganography and steganalysis are also described. We introduce some basics about natural images statistics at the end of this chapter.

- In Chapter 3, we improve steganographic security by selecting suitable natural

cover images. The KL divergence and two statistical models for natural images are used for the analysis purpose. Properties of cover image that intrinsically affect the undetectability of steganography are discussed in this chapter.

- In Chapter 4, we propose a unified CNN model for image steganalysis. By incorporating adaptive content suppression and residual learning, the proposed model can detect modern adaptive steganographic algorithm with better performances than previous methods when cover images and their stego images are paired in training and testing. We also discuss the limitation of a CNN model with batch normalization layers for image steganalysis.

- In Chapter 5, we give conclusions and possible future directions of this work.

Chapter 2

Literature Review

In this chapter, we review technical details of various steganographic techniques. According to the data embedding operations, we classify existing techniques into three main categories: LSB based steganography, quantization based steganography, and content adaptive steganography. Steganographic algorithms in the transform domain are introduced in the same section. Meanwhile, the counterpart of steganography, steganalysis, is also described in this chapter. In order to analyze steganography in theory, we introduce several validation metrics based on the information theoretical steganography. To overcome the computational difficulty of the theoretical security measure, we review several alternative methods to evaluate steganographic security. At the end of this chapter, we review some basics about the statistics of natural images and introduce two representative models for their description.

The rest of this chapter is organized as follows. In section 2.1, we review the technical details of steganographic algorithms and further analyze their pros. and cons. In section 2.2, we briefly review the techniques of steganalysis. In section 2.3, we review the theoretical foundation of steganography and introduce several existing methods to evaluate steganographic security. In section 2.4, we introduce some background knowledge about natural images.

2.1 Steganography

Since the first information hiding workshop was hold in 1996, techniques of steganographic algorithms has developed a lot over the past two decades. Here we chronologically list the key developing points of steganography:

- 1984: Simmons in [32] proposed the prisoner escaping problem, becoming the general formulation of modern steganography;
- 1985: Barrie Morgan and Mike Barney designed two c0(see zeros) systems [39] with the advent of personal computer, which were considered as two first steganographic algorithms in digital era;
- 1992: Charles Kurak and and John Mchungh proposed the least significant bit (LSB) [17] for hiding messages in digital images. The LSB method is fundamentally important for modern steganography for its practicability, and mostly important, it was the simplest algorithm satisfying all the three requirements of a secure stegosystem at that time;
- 1996: the first information hiding workshop opened, where many terminologies, such as cover signal, stego signal, message embedding, were defined. The workshop also clarified the differences between steganography, digital watermarking and fingerprinting [88];
- 1998: Cachin defined the security of a stegosystem in terms of the information theory [33], providing a solid theoretic basis for the research on steganography. We will introduce this definition and the information theoretic steganography at the end of this chapter;
- 1998: Kawaguchi proposed the first content adaptive steganography, the bit plane complexity steganography (BPCS) [28], which took the limitations of HVS to achieve high embedding capacity. Meanwhile, the matrix embedding steganography was introduced by Crandall [84];

- 2001: Quantization index modulation (QIM) steganography was proposed by Chen [12]. At the same year, Shi descried the reversible steganography [119], a new data hiding scheme requiring that the original data should be losslessly recovered;

- 2006: Younhee Kim proposed the modified matrix embedding steganography. This method extends the matrix embedding algorithm that more than one pixels/coefficients can be changed. This new method paves the way for the development of modern steganography, i.e. content adaptive steganography based on the channel coding. Following Kim's work, Fridrich proposed the framework of the wet paper coding [49].

- 2011: Tomas Filler proposed a novel method to minimize the additive distortion function with the Syndrome-Trellis Codes (STC) [96]. This method now becomes the best choice for steganographiers to embed secret messages once the distortion value for each pixel/coefficient is defined.

In recent years, steganography based on STC becomes the the mainstream for its efficiency and high undetectability. Based on this framework, the only work that a steganographier does is to develop novel distortion function. In the following sections, we review various steganographic algorithms in details.

2.1.1 LSB based Steganography

LSB steganography LSB steganography is one of extensively used technique capable of hiding large secret message in a cover image without introducing severe perceptible distortions [96]. In fact, LSB is the simplest case of Weber's law - it just takes the least modifications to the cover image. LSB directly replaces the least significant bits of randomly selected pixels in the cover image with the message bits [111]. The selection of image pixels is controlled by a secrete key to make the message unreadable

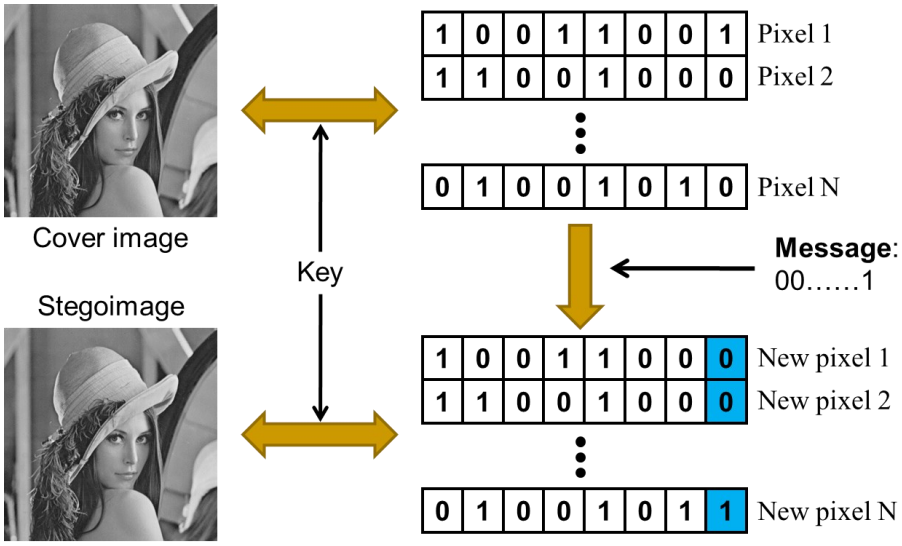


Figure 2.1: Schematic illustration to the LSB steganography

to some intenders. The basic description of the LSB steganography is shown as follows:

$$y_i = 2\lfloor x_i/2 \rfloor + m_i \tag{2.1}$$

where x_i, y_i, m_i represent the i-th message bit, the i-th selected pixel value before embedding, and that after embedding, respectively. In general, the messages embedded into the cover image are compressed and encrypted. In this case, the binary message bits are assumed to be approximate a uniform distribution, e.g. $P_{\mathbf{m}}(m = 0) = P_{\mathbf{m}}(m = 1) = 1/2$, where $P_{\mathbf{m}}(m = 0)$ and $P_{\mathbf{m}}(m = 1)$ represent the probabilities of binary message bit 0 and 1.

Since the least signification bits of a cover image are replaced by the message bits, whose distribution is an uniform distribution, the LSB technique enforces that pixels with adjacent values are equally distributed, called pair of value (PoV) phenomenon [10]. The PoV denotes that the histograms of two adjacent pixel values in the LSB stego image are equally high to each other. This phenomenon was utilized by many researcher to design steganlytic tools for message detection. For example, Westfeld in [10] proposed χ^2 statistics to analyze the existence of LSB embedding:

$$p = 1 - \frac{1}{2^{\frac{k-1}{2}}\Gamma\left(\frac{k-1}{2}\right)} \int_0^{\chi^2_{k-1}} e^{-\frac{x}{2}} x^{\frac{k-1}{2}-1} dx \tag{2.2}$$

where $\chi^2_{k-1} = \sum_{i=1}^{k} \frac{(O_{2n}-O_e)^2}{O_e}$, $O_e = \frac{O_{2n}+O_{2n+1}}{2}$, O_{2n} and O_{2n+1} denote the occurrence times of pixels whose values are equal to $2n$ and $2n+1$ respectively. $\Gamma(\cdot)$ denotes the Gamma function. In the Eq.(2.2), the value of p is roughly equal to the probability of message embedding. Experiments prove that the χ^2 statistic can attack LSB steganography very accurately if the message bits are embedded into a continuous region. Despite the χ^2 attack, several improved algorithms were proposed to overcome its limitation. For example, Fridirch's RS steganalysis [45] and weighted stego [44], Dumitrescu's simple pair analysis (SPA) [89] and recently hypothesis testing theory based method [83] can reliably attack the LSB steganography even though message bits are hidden in randomly selected locations.

LSB Matching Steganography In order to avoid the PoV phenomenon, Sharp [103] proposed an improved version of LSB steganography, the LSB matching (LSBM). Unlike the LSB steganography directly replaces the LSB with the message bits, LSBM works in this way: if the LSB matches the message bit, no operation is done, otherwise LSBM randomly adds +1 or -1 to the current pixel according to a secret key.

$$y_i = \begin{cases} x_i, & mod(x_i, 2) = m_i \\ x_i \pm 1, & mod(x_i, 2) \neq m_i \end{cases} \tag{2.3}$$

Compared with LSB method, LSBM does not generate the PoV phenomenon, thus achieve better security. Generalizing Sharp's work, Fridrich further proposed $\pm k$ steganography [47]. The method is similar to LSB matching but embeds more message bits at one pixel. In addition, she proposed the stochastic modulation based steganography [43]. Instead of hiding message bits into pixels according to uniform distribution,

this method embeds the message by adding a weak noise with arbitrary distribution. The detection accuracy is decreased when the sender embeds messages by simulating the noised generated by digital devices. Mielikainen in [57] utilized the redundancy between the message bits and the LSB of cover images to hide secret messages, achieving a higher embedding efficiency than LSBM.

Adaptive LSB steganography Currently, the developments to the LSB steganography fall in two directions. The first direction is to hide the messages bit by preserving the statistical properties of the cover image. Sallee's model based (MB) steganography [77] follows this idea. Before embedding, MB firstly estimates the conditional distribution between the deterministic variables (MSB, maximal significant bits) and the indeterministic variables (LSB). Then, the algorithm embeds the messages according to the estimated distribution. Another typical example is Fridrich's stochastic modulation based steganography [43], which embeds the message bits into the message by adding a noise signal with arbitrary distribution. Based on this method, the sender can embed the messages into cover images with a distribution similar to the natural noise. The second direction is the content adaptive LSB. For example, both Yang's method [18] and Luo's method [112] embeds the messages at the locations in which the difference between two neighborhood pixels is large. Since these locations are often at the edges, highly textured and cluttered regions, they are too complex to be modeled by off the shelf mathematical tools. Meanwhile, the distribution of LSB at these locations also approximates a uniform distribution. These factors greatly increase the security of the steganography.

2.1.2 Quantization based Steganography

Quantization is a frequently used technique to decrease the size of digital images for efficient transmission. Inevitably, information loss and artificial distortions are introduced during this process. The quantization based steganography is to embed the message

bits into the cover image by alternating the quantizer itself or the quantization table. Since quantization is an information reduction mapping, the security can be enhanced if the alternations to the cover images are lost in the quantization operation.

Quantization index modulation QIM is the first approach to embed the message in image quantization. In [12], Chen generalized the message embedding as a super-channel model. In the following figure, m is the original secret message, x is the cover signal, e is the encrypted secret message, s is the stego signal, y is the received signal, $\hat{m}$ represents the recover secret message.

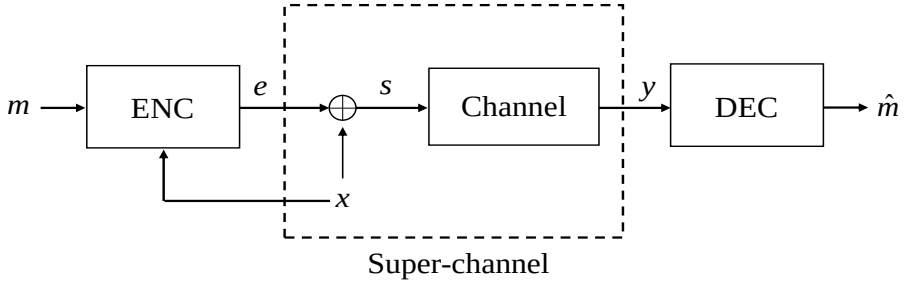


Figure 2.2: Equivalent super-channel model for information embedding. The composite signal is the sum of the host signal, which is the state of the super-channel, and a host-dependent distortion signal

Based on the channel model as Fig.2.2, one can take information-embedding problems as power-limited communication over a super-channel with a state that is known at the encoder. QIM embeds the message bit 0 or 1 by using different quantizers:

$$y_i = \mathbf{Q}_m(x_i) = \begin{cases} \Delta\lfloor x_i/\Delta + 1/2 \rfloor, \ if \ m = 0 \\ \Delta\lfloor x_i/\Delta \rfloor + \Delta/2, if \ m = 1 \end{cases} \tag{2.4}$$

where Δ denotes the quantization step. As proved by author from information theory, QIM methods are provably better than additive spread spectrum and generalized LSB against bounded perturbation and in-the-clear attacks. Therefore, they are also widely used in digital watermarking for its robustness against noise attacks. However, a fatal defect of QIM method is it makes the histogram of stego image sparser than the cover

image. This phenomenon become quantization step Δ is large. The remedy is to use a dither modulation during message embedding:

$$y_i = \mathbf{Q}_m(x_i + d_i) - d_i \tag{2.5}$$

where d_i is the dither signal uniformly distributed in $[-\Delta/4, \Delta/4]$ and determined by a key shared by senders and receivers. Compared to the original method, QIM with dither modulation successfully avoid the histogram sparsity phenomenon without decreasing the robustness or sacrificing advantages over the spread spectrum method and LSB steganography.

Quantized table modulation Unlike QIM embeds the message in the spatial domain, the quantization table modulation (QTM) [15] is to hide the message by changing the JPEG quantization table and then modify the middle-frequency DCT coefficients for message embedding.

$$\begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix} \Rightarrow \begin{bmatrix} 16 & 11 & 10 & 16 & 1 & 1 & 1 & 1 \\ 12 & 12 & 14 & 1 & 1 & 1 & 1 & 55 \\ 14 & 13 & 1 & 1 & 1 & 1 & 69 & 56 \\ 14 & 1 & 1 & 1 & 1 & 87 & 80 & 62 \\ 1 & 1 & 1 & 1 & 68 & 109 & 103 & 77 \\ 1 & 1 & 1 & 64 & 81 & 104 & 113 & 92 \\ 1 & 1 & 78 & 87 & 103 & 121 & 120 & 101 \\ 1 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

Figure 2.3: Schematic illustration to the QTM steganography

In the Fig.2.3, the table on the left side is the original JPEG quantization table, while the right side one is the modified quantization table. The reason for choosing middle-frequency DCT coefficients is that they are the tradeoffs to both achieve invisibility and security.

Some researchers have generalized the quantization based steganography in the framework of channel selection. The advantage of this framework is it can significantly increase the security by embedding message bits into pixels or DCT coefficients hard to

be discriminated from the quantization noise and the embedded bits when the image is quantized.

2.1.3 Content Adaptive Steganography

Currently, the content adaptive steganography becomes the mainstream of modern steganography. According to the development of this approach, it can be clearly classified into two stages. In the first stage, the content adaptive steganographies achieve high embedding capacity and security by embedding message bits into some busy regions such as textures or clutters. Their success lies in the fact that our visual system is insensitive to the changes in those busy regions. However, as the development of steganalysis, many tools can easily attack these approaches even though they are secure to HVS. Therefore, in the second stage, distortion minimization (DM) based steganography becomes popular for its high security to various steganalysis tools. Different from early content adaptive steganography, DM is to hide the message bits into positions secure to various detection algorithms.

BPCS. The first content adaptive steganography is BPCS proposed by Kawaguchi in [28]. Before message embedding, BPCS firstly defines a complexity measure for each bit plane of a region:

$$\alpha = \frac{k}{The\ maximum\ possible\ B-W\ changes\ in\ the\ region} \tag{2.6}$$

where k is the total length of black-and-white border in the binary region, which equals to the summation of the number of color-changes along the rows and columns in a region. In Fig.2.4, the α values of pure white or black regions are 0, while the regions as Wc and Bc give the largest value, $\alpha = 1$.

To embed secret message, BPCS searches the regions P with large α value and replaces the original pixel bits with the message bits. In order to achieve high em-

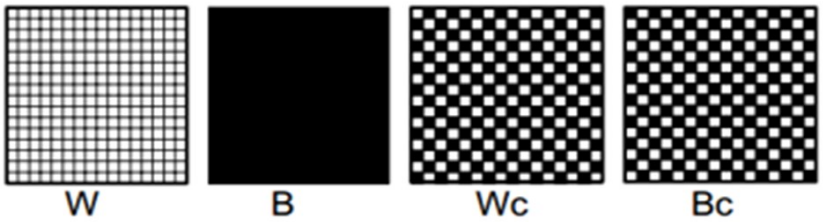


Figure 2.4: Low α regions and high α regions. In the figure, B and W denote the image with same pixel value(lowest and largest).

bedding capacity, the method transforms the informative regions, whose α values are small, with a conjugation operation:

$$P^* = P \oplus Wc \tag{2.7}$$

where $\oplus$ designates the bit-wise exclusive OR operation. By computation, the conjugation operation has following properties:

$$(P^*)^* = P \tag{2.8}$$

$$\alpha(P^*) = 1 - \alpha(P) \tag{2.9}$$

After this transformation, BPCS embeds the message bits into both MSB and LSB of regions with large α value. Experiments proved that BPCS can hide large amount of messages but leads to small visual artifacts.

Pixel value differencing. Pixel value differencing (PVD) proposed by Wu [24] is another typical content adaptive steganography using the limitation of HVS. The complexity in PVD is defined as the difference between two neighboring pixels. The larger the difference is, the higher the complexity it achieves. Generally, there are four steps for PVD steganography:

- Divide the image into non-overlapping two pixel blocks;
- Use a random key to visit all the blocks; compute the difference value and classify

it into a number of contiguous ranges;

- Replace the original difference value with the message bits;
- Pixel value inversion.

The absolute difference between two neighboring pixels could be any value in [0,255]. Before embedding, [0,255] is divided into several continuous intervals. A typical dividing scheme is [0,7], [8,23], [24,55], [56,87], ..., where the lengthes of the intervals are 8, 16, 16, 32, 64, 128, which implies 3, 4, 4, 5, 6 bits can be replaced by the messages. Then PVD computes the difference between two neighoring pixels, 50, 65 in the figure and the difference is 15. Checking the dived intervals, we find 15 fall into [8,23], indicating 4 message bits can be hidden into the selected pixels. Selecting four bits from the message, PVD replaces the original difference value with the message bits. The message bits embedded into two selected pixels according to the mapping function:

$$(\hat{g}_i, \hat{g_{i+1}}) = \begin{cases} (g_i - ceiling_m, g_{i+1} + floor_m), \ if \ mod(d,2) \neq 0 \\ (g_i - floor_m, g_{i+1} + ceiling_m), \ if \ mod(d,2) = 0 \end{cases} \tag{2.10}$$

where $\{(g_i, g_{i+1}), (\hat{g}_i, \hat{g}_{i+1})\}$, d, d' are the neighboring pixels, the difference values before and after message embedding, $m = d - d'$ and $celiling_m = \lceil m/2 \rceil$, $floor_m = \lfloor m/2 \rfloor$. The purpose of Eq.(2.10) is to adjust the difference value d to the value d' corresponding to binary messages.

Similar to PVD, Zhang in [113] posed a multiple-base notational system steganography. Both PVD and Zhang's method utilize the limitation of HVS, but Zhang's method embeds the messages into a busy region whose variance is large. Following the idea of PVD, Luo and Yang proposed two edge adaptive steganography by hiding the bits into highly different pixels with LSBM. Currently, the content adaptive steganography steps from HVS security to statistical security. The key to content adaptive approach is to find appropriate hiding positions and a reversible process to encode and decode

the messages. In the following section, we discuss this approach.

Channel coding based steganography For content adaptive steganography, both senders and receivers should have complete knowledge about the positions for message hiding. However, the channel coding based steganography can make receivers correctly decode the messages even though they do not have any knowledge about these embedding positions. Matrix embedding, proposed by Crandall in [33], is the first channel coding steganography to hide message bits by changing at most one bit.

Assume $\mathbf{x}$ is the LSBs of the pixels or quantized DCT coefficients, $\mathbf{H}$ is a binary matrix, $\mathbf{y}$ is the LSBs with hidden messages, $\mathbf{m}$ represents the message string. For matrix encoding, it hides the message string $\mathbf{m}$ by solving the following optimization problem:

$$\mathbf{y} = \begin{cases} \mathbf{x}, \ if \ \mathbf{Hx} = \mathbf{m} \\ \mathbf{x}', \ if \ \mathbf{Hx} \neq \mathbf{m} \end{cases} \tag{2.11}$$

the constraint is:

$$\mathbf{s.t.} \ \ dist_{Hamming}(\mathbf{x}, \mathbf{x}') = \sum_i (x_i \oplus x_i') \leq 1 \ \ and \ \ \mathbf{Hy} = \mathbf{m} \tag{2.12}$$

In many applications, the Hamming (7,3) binary matrix $\mathbf{H}$ in liner error-correcting codes (ECC) is used for encoding:

$$\mathbf{H} = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$$

Figure 2.5: The Hamming (7,3) ECC encoding matrix

Once the message string $\mathbf{m}$ is embedded into the cover image $\mathbf{x}$, a receiver can decode the message by applying a binary matrix multiplication, $\mathbf{m} = \mathbf{Hy}$. Except for Hamming

codes, there exist many other codes in channel coding theory, such as BCH codes [87], syndrome trellis code [96], etc.

The matrix embedding steganography requires at most one bit can be changed during message hiding. Kim in [115] relaxed this constraint and proposed modified matrix embedding (MME). In MME, $t \geq 1$ bits can be changed during embedding:

$$\mathbf{y} = \begin{cases} \mathbf{x}, \ if \ \mathbf{Hx} = \mathbf{m} \\ \mathbf{x}', \ if \ \mathbf{Hx} \neq \mathbf{m} \end{cases} \tag{2.13}$$

where the constraint is relaxed that t bits can be modified:

$$\mathbf{s.t.} \ \ dist_{Hamming}(\mathbf{x}, \mathbf{x}') = \sum_i (x_i \oplus x_i') \leq t \ \ and \ \ \mathbf{Hy} = \mathbf{m} \tag{2.14}$$

where $\oplus$ denotes the *xor* operator. For each bit, a distortion function is defined when it is changed:

$$d(x_i, y_i) = 1 - \delta(x_i - y_i) \tag{2.15}$$

where $\delta(\cdot)$ represents the delta function. Then MME can be formulated as a distortion minimization problem:

$$\mathbf{y} = min \ \mathbf{D}(\mathbf{x}, \mathbf{y}) \ \ \mathbf{s.t.} \ \ \mathbf{Hy} = \mathbf{m} \tag{2.16}$$

where the total distortion $\mathbf{D}(\mathbf{x}, \mathbf{y}) = \sum_i d(x_i, y_i)$. Current researches on channel coding based steganography focus on the design of $\mathbf{H}$, the design of distortion function $\mathbf{D}$, how to solve effectively the optimization problem as Eq.(2.16). The syndrome trellis coding (STC) proposed by Fridrich have successfully solved the optimization problem as Eq.(2.16) when the distortion is additive. STC liberates the researcher to focus on the design of distortion function, making the DM framework become mainstream.

2.1.4 Steganography in the Transform Domain

Unlike spatial domain steganography that directly hides message bits into image pixels, steganography in the transform domain changes the transformed coefficients to realize message hiding. These transformed coefficient can be the DCT coefficient for the JPEG image [50, 65, 66] or the wavelet coefficient for the JPEG2000 image [68, 78]. In general, the transform domain steganography can be divided as the following two main categories:

Random embedding approach This approach hides secret message bits into the randomly selected nonzero coefficients. The main advantage of this approach is that it can hide information efficiently. Jsteg [61] directly extends the LSB steganography to JPEG images , which changes the LSB of DCT coefficients to embed message bits. Outguess [75] is similar to Jsteg, but it adds some additional bits to adjust the histogram of DCT coefficients to avoid distortion. With matrix embedding, F5 [11] randomly select one nonzero coefficient for changing, achieving high embedding efficiency than Jsteg and Outguess. In order to eliminate the shrinkage phenomenon in F5, Fridrich proposed the non-shrinkage F5 (nsF5) [50] method based on the wet paper coding. Although the random embedding approach is efficient to hide message, they are easy to be attacked by detection algorithms.

Adaptive embedding approach Different from random embedding approach that hides message bits in a random manner, adaptive embedding methods hide messages into coefficients that are hard to be attacked. By heuristically defining a distortion function for each coefficient, the adaptive embedding approach uses STC to embed messages into coefficients with low distortion value. Following this idea, Holub in [107] proposed a universal distortion function based on the wavelet transform. The method forces that DCT coefficients to be embedded are the ones that lead to least changes in the wavelet domain. Guo in [65-66] proposed a distortion function based on the idea

of uniform embedding. The method defines the DCT coefficient with low distortion value if message embedding can result in a more uniform distribution. Compared with random embedding approach, the adaptive embedding approach is securer for message hiding, thus becomes the main research topic in the compressed domain steganography.

2.2 Steganalysis

The aim of steganography is to hide messages into cover signal to avoid being detected, while steganalysis from an opponent's perspective, is the art of deterring covert communications while avoiding affecting the innocent ones [80]. Its basic requirement is to determine whether a secret message is hidden in the testing medium. According to the knowledge they use, the technique can be roughly classified as specific method and universal method, which are introduced as follows.

2.2.1 Specific Steganalysis

Specific steganalysis fully utilizes full knowledge of a targeted steganographic algorithm to discriminate the cover image and the stego image. In general, there are two methods for specific steganalysis for the spatial domain images.

Embedding sensitive statistics based method This method constructs statistical quantities that are sensitive to a given message embedding algorithm for the discrimination of covers and stegos. By using the PoV phenomenon of LSB replacement embedding, the χ^2 steganalysis detects LSB steganography by calculating the χ^2 statistical quantity of digital images. In [45], a discrimination function in RS steganalysis, which is defined to capture the smoothness of a group of image pixels, divides image pixels into three groups. With the flipping operation, the RS method can find the number change of two groups and then determines whether the input image is a cover or a stego. The WS steganalysis [44] transforms each pixel into its weighted stego version by LSB flipping. The weighted average of image pixels is utilized to discriminate covers and stegos and further determine the size of embedded message.

The SPA steganalysis [89] extracts several multisets of sample pairs based on the finite state machine for hidden message detection and the message length estimation.

Hypothesis testing based method This method formulates the message detection to a hypothesis testing problem. The key of the method is to find a good probability model for natural images. Cogranne [82] used a zero-mean Gaussian distribution to model the noise during the acquisition. Based on the probability model, the paper proposed an Asymptotically Uniformly Most Powerful (AUMP) test to maximize the detection power for the LSB matching steganography. In [81], a new probability function is proposed to model the distribution of quantized DCT coefficients. Based on the hypothesis testing framework, the algorithm can reliably detect the Jsteg steganography.

In summary, the specific steganalysis is only effective for the targeted steganography. It cannot be generalized for other steganographic algorithm, which limits its application in practice.

2.2.2 Universal Steganalysis

In contrast to specific steganalysis needing strong prior knowledge, universal steganalysis detects hidden message in which no prior knowledge is provided [13]. Instead of using the knowledge about a specific steganographic method, they transform the steganalysis into a general binary classification problem.

Wavelet feature based method This method takes high order moments of wavelet coefficients as the feature vector for message detection. Farid in [35] proposed to use the regression vector between each wavelet coefficient and its neighborhood coefficients as the training feature. A Fisher linear discriminant (FLD) is learned based on the extracted feature, which is utilized to discriminate natural images and their stego versions. Following this work, Lyu [92] combined the higher order statistics of the wavelet coefficient and a SVM classifier to detect various steganographic algorithms.

In [70], Goljan *et al.* proposed to extract higher order absolute moment of wavelet coefficients, called the Wavelet Absolute Moment (WAM), to attack the LSB matching steganography.

Co-occurrence matrix based method In recent years, an increasing number of universal algorithms tend to use the co-occurrence feature because of its low extraction cost and high discriminability. For instance, Zou *et al.* in [27] extracted the Markov transition matrix from the error prediction image and trained a support vector machine for classifying the cover images and stego images. Pevny and Fridrich in [101] further developed the SPAM feature which showed promising performance for additive steganography. In the last four years, Fridrich and her group proposed several excellent steganalysis schemes such as rich model methods [46, 106] are developed based on various higher order co-occurrence matrices. Currenly, the rich model based steganalysis becomes an effective choice to attack both spatial domain steganography and transform domain steganography.

Convolutional neural network (CNN) based method Several pioneering works have been proposed to use deep CNN to attack content adaptive steganography. Unlike the rich model method that utilizes handcrafted features, CNN based methods directly learn effective features from input images to classify covers and stegos. In [94], Tan and Li presented a stacked convolutional auto-encoder to detect the presence of secret message. In this network, three processing units extract features from input images and a three-layer fully connected neural network maps the extracted features into their labels. For each processing unit, it contains a convolutional layer, a maximum pooling layer and a sigmoid activation layer. The network shows better performance than the traditional SPAM [101], but it is worse than the rich model method. Qian *et al.* in [117] proposed a different CNN architecture consisting of five convolutional layers, in which each layer is followed by an average pooling layer and a nonlinear activation layer. To better distinguish cover images and stego images, the paper proposed to

use Gaussian rather than sigmoid as the activation function. Even though Qian's network is inferior to the rich model method, the performance gap between CNN and the rich model has been narrowed from 14% (Tan and Li's network) to $2\% - 5\%$. To further improve the accuracy of CNN for steganalysis, Xu *et al.* [34] designed a new CNN model incorporating the domain knowledge of steganography and steganalysis. By taking absolute values to outputs of the first convolutional layer and applying the *tanh* activation function to the first two convolutional layers, the network improves the modeling ability to input images and prevents overfitting. Because of these modifications, Xu's network achieves competitive performances with the rich model method on S-UNIWARD and HILL. After trying numerous experiments for CNN with different structures, Pibre *et al.* [67] found a CNN model that first surpasses the rich model method on S-UNIWARD at 0.4 bit-per-pixel (bpp). Pibre's network has two convolutional layers but no pooling layers. This feature makes the model able to preserve the information generated by message embedding when the data goes through the whole network. The reported detection error rate to 0.4 bpp S-UNIWARD is 7.4%, which is greatly smaller than rich model's 20%. Compared with other universal steganalytic methods, the CNN based method shows excellent performances in attacking various content adaptive steganography. This makes the method becomes increasingly hot in recent two years.

2.2.3 Steganalysis for JPEG Images

Same to the spatial domain case, steganalysis for JPEG images can also be divided into specific methods and universal methods. Specific methods make full knowledge of the embedding details of the targeted JPEG steganography for accurate detection. The chi-square [10] steganalysis utilized the PoV character to attack the Jsteg steganography. For OutGuess, the message embedding increases the blockiness, which is utilized for hidden message detection [41]. By directly estimating cover-image histogram from the stego image, Fridrich [48] successfully attacked F5 steganography. Universal methods

extract steganalytical features from the JPEG images and then train a binary classifier to distinguish the cover images and the stego images. Different steganalytical features for JPEG images are used for detection, e.g. 548-dimentional PEV features [100], 144-dimensional LIU features [79] and 8000-dimentional DCTR features [105]. With the extracted features, a binary classifier, support vector machine or ensemble classifier, is trained and used to discriminate the covers and the stegos. Recently, Ker [2] and Li [30] proposed to use clustering rather than classification to identify the suspected steganographier. The advantage of universal method is that it requires less or no prior knowledge on the embedding details of steganography.

2.3 Validation Metrics

In this section, we introduce several validation metrics to evaluate the security of steganographic algorithms. We follow the model proposed by Cachin in [84], where Alice and Bob are the message sender and receiver, while Wendy is an passive adversary who has perfect read-only access to the public channel. Following the approach of information theory, the knowledge of covertext and stegotext are captured by probabilistic models and Wendy's task of detecting hidden messages is viewed as a problem of *hypothesis testing*.

During communication, Alice operates in one of two modes. In first case, Alice is inactive and sends an innocent message containing no hidden information, called covertext and denoted by C. it is generated according to a distribution P_C known to Eve. In the second case, Alice is active and sends stegotext S with distribution denoted by P_S. The stegotext is computed from an embedding function $\mathscr{F}$ and contains an embedded message E intended for Bob. The message is a random variable drawn from a message space ε.

Definition. Fix a covertext distribution C and a message space ε. A pair of algorithms $(\mathscr{F}, \mathscr{G})$is called a stegosystem if there exist random variables K and R

as described such that for all random variables E over ε with $H(E) > 0$, it holds $I(E; \hat{E}) > 0$. Where $\hat{E}$ is the random variable received by receivers, $H(E) > 0$ and $I(E; \hat{E}) > 0$ are entropy and mutual information:

$$H(X) = -\int_{x \in \mathscr{X}} P_X(x) log P_X(x) dx \tag{2.17}$$

$$I(X; Y) = H(X) - H(X|Y) \tag{2.18}$$

A stegosystem is called ϵ-secure (*against passive adversaries*) if:

$$D(P_C||P_S) = -\int P_C log \frac{P_C}{P_S} dP \leq \epsilon \tag{2.19}$$

where P_C is the probability distribution of covertext, P_S represents the probability distribution of stegotext, $D(P_C||P_S)$ is the KL divergence between two distributions.

Remarks

1. When $\epsilon = 0$, the system is called *perfectly secure*. In this case, Wendy cannot discriminate the P_C and P_S. This makes the observer have no information about the presence of hidden message.

2. The condition in the definition of a stegosystem, $I(E; \hat{E}) > 0$ implies that a stegosystem indeed transmits useful information to Bob;

3. The model assumes that the covertext in current transmission that is not known to Eve. He only has the knowledge of the distribution P_C.

4. The purpose of steganography is to design sophisticated data embedding algorithms to make $D(P_C||P_S)$ as small as possible.

The KL divergence provides fundamental security measure for steganography. However, it can hardly be applied in practice because of two reasons. For the first, for a

given steganographic algorithm, it is often hard to obtain accurate P_C and P_S for the cover signal and its corresponding stego signal. For the second, there is no close expression for $D(P_C||P_S)$, making the calculation to the KL divergence difficult. To handle this difficulty, there are four alternative methods to measure the security of steganographic algorithms:

Detection Error This approach utilizes a steganalytic algorithm to predict the label of many test images. In general, the detection algorithm will generate two types of errors. The first error is the false alarm P_{FA}, which is the error of mistaking a cover image as a stego image. The second error is the miss detection P_{MD}, which denotes the error of classifying a stego image to a cover image. To measure the security of steganography, the detection error P_E takes the minimal total error with equal priors for evaluation:

$$P_E = \min_{P_{FA}} \frac{P_{FA} + P_{MD}(P_{FA})}{2} \tag{2.20}$$

Large P_E indicates the steganographic algorithm is hard to be detected. Currently, the detection error is most frequently used method to evaluate the performance of steganography. The main limitation of this evaluation measure is that it cannot represent the true security of steganography. According to the information processing theorem, any extraction of features in steganalysis results in the lose of information. In addition, different classification algorithms in steganalysis also leads to different detection accuracies.

Receiver Operating Characteristic (ROC) curve Kharrazi in [74] proposed to use the ROC curve the evaluate the detectability of different steganographic algorithms. ROC actually draws the curve of the true positive (TP) rate against the false positive (FP) rate at different thresholds. For steganography, TP rate measures the case that a stego image is correctly classified as stego; while FP rate measures the case that a cover image is mistaken as stego. The area under ROC curve reflects the detectability

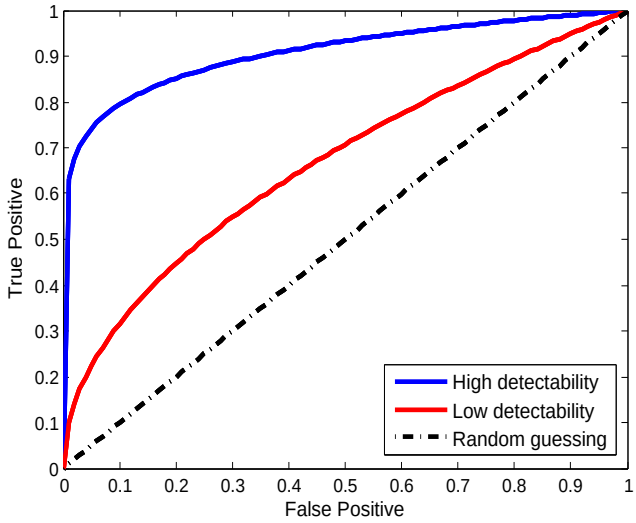


Figure 2.6: Schematic illustration to ROC curve. The area under the curve indicates the detectability of a steganographic algorithm. The smaller the area is, the harder the algorithm to be deected.

of steganographic algorithms: if the area is small, the algorithm is hard to be detected. Fig.2.6 demonstrates three cases of the ROC curve. Compared with detection error, ROC curve can provide more information when the detection algorithm varies.

Maximum Mean Discrepancy To overcome practical difficulties coming from estimating the KL divergence, Pevny in [99] proposed to use the mean maximum discrepancy (MMD) to benchmark steganography. MMD calculates the difference between the cover image and the stego image in a given feature space as:

$$\mathrm{MMD}\left[\mathcal{F}, \mathbf{X}, \mathbf{Y}\right] = \sup_{f \in \mathcal{F}} \left(\frac{1}{D} \sum_{i=1}^{D} f(x_i) - \sum_{i=1}^{D} f(y_i) \right) \tag{2.21}$$

where $\mathbf{X} = \{x_1, ..., x_D\}$, $\mathbf{Y} = \{y_1, ..., y_D\}$ represents D cover images and their corresponding stego images, $\mathcal{F}$ denotes the feature space. The selection of feature space heavily determines the accuracy of MMD. In practice, various steganalytic features are adopted as $\mathcal{F}$. Although MMD proves to be numerically stable even for high dimensional feature space, the feature extraction inevitably loses useful information. This defect limits its further application for the evaluation of steganographic security.

Steganographic Fisher Information Ker in [4] proposed to use the local quadratic term of the KL divergence, i.e. steganographic fisher information, to measure the security of steganographic algorithms. In the asymptotic behavior, the KL divergence can be expanded as:

$$D_{KL}\left(P(0)||P(\lambda)\right) \sim \frac{1}{2}I\lambda^2 + O(\lambda^3) \tag{2.22}$$

where λ denotes the payload size, I represents the Fisher information of the distribution $P(\lambda)$ around zero. Ker used I to measure the security of steganography. According to the paper, algorithms with low I are more secure than those with high I. The Fisher information is effective for measuring steganographic security. However, the estimation of I is too expensive and unstable to be applied for the security evaluation. In chapter 5, we have derived analytic expressions for the steganographic Fisher information, which is used to analyze what the properties of cover images that can affect steganographic security.

2.4 Natural Images

Natural images, often refer to photographs of our typical environment we live in, are the most popular image files on the internet. Fig.2.7 shows some typical examples of natural images and non-natural images (e.g. cartoon image, synthesized image, noise image). Compared to these non-natural ones, natural images seem more "acceptable" to our human beings. Many researches on the Human Visual System (HVS) [5, 9, 21] support this phenomenon that neural representations of our visual cortex have similar statistical regularities in natural images. In statistics, natural images have several distinctive properties that make them different from other kinds of images.

- Scale invariance. The property states that the statistical properties of an ensemble of natural images is independent of their sizes. This indicates that some key image statistics do not change even though their scales are changed. Several

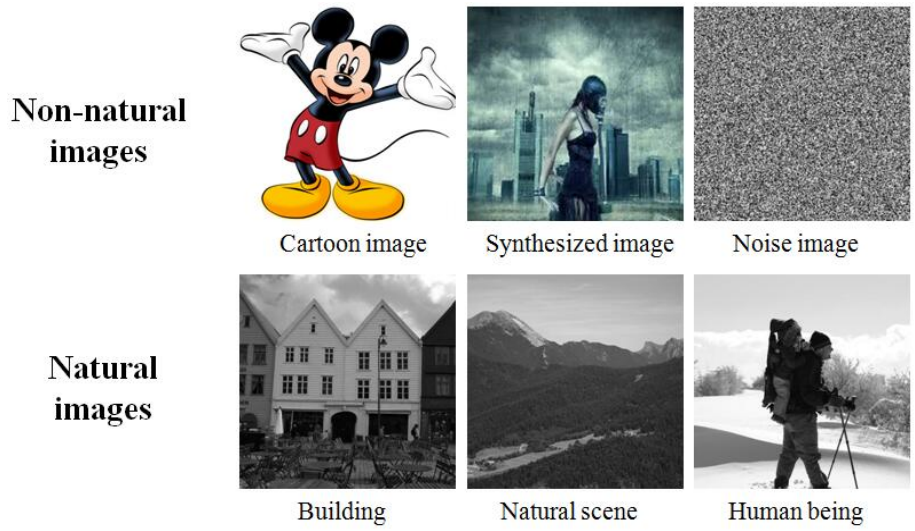


Figure 2.7: Demonstration of non-natural images and natural images.

works [8, 19, 20] found that the power spectrum of natural images approximately follow the power low $S(k) \propto 1/k^{2-\eta}$, where k denotes the spatial frequency, η is a small constant. The form of spectrum $S(k)$ remains same when spatial frequency k is changed, which provides a strong evidence for scale invariance.

- Structural richness. Natural images have many different kinds of structures. These structures could be the lower order histograms and correlations among neighborhood pixels, or some higher order regularities which should be represented by various image patterns [25, 118]. Many mathematical models have been proposed to describe the structures, such as the Gaussian Mixture Model [26], the Markov Random Field (MRF) [90], Field of Expert model [93] etc. These models, however, can not fully represent the rich and complex structures in natural images, thus provide space for applying image steganography.

- Local correlation. Adjacent pixels of a natural image are not independent, they are strongly correlated with each other. Many evidences [53, 116] show that natural images are highly redundant, mainly because of strong correlations among neighborhood pixels. For steganalysis, this local correlation can be used to find abnormal relationships caused by message embedding.

For many applications including image compression, denoising, steganography or steganalysis, we need models to describe the distribution of natural images. In the following part, we will introduce several representative examples for modeling the distribution of natural images.

2.4.1 Gaussian Mixture Model

Gaussian Mixture Model (GMM) [26] is a simple but effective probability model to describe the distribution of natural images in the spatial domain. In mathematics, this model can be viewed as a special case to the mixture model that:

$$p(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i N(\mathbf{x}; \boldsymbol{\mu}_i, \Sigma_i) \tag{2.23}$$

where $\mathbf{x}$ denotes a small patch in natural images. $(\alpha_i, \boldsymbol{\mu}_i, \Sigma_i)_{i=1}^{N}$ represent the parameters of the GMM, where $\alpha_i \geq 0$ and $\sum_{i=1}^{N} \alpha_i = 1$. This model is widely used in natural image modeling because GMM can approximate any continuous probability function with enough number of components. In addition, parameters of GMM can be efficiently learned by the Expectation Minimization (EM) algorithm. For natural images, Zoran in [26] found that GMM with a small number of components can compete any other state of the art models such as Principal Component Analysis (PCA), Independent Component Analysis (ICA) or the Gaussian Scale Mixture (GSM) [59]. This result demonstrates the effectiveness of GMM for modeling natural images.

2.4.2 Convolutional Neural Network

CNN has achieved a great success in many image related tasks [6, 62, 63], indicating its superior capacity to capture the structure of natural images. Fig.2.8 shows a typical example of CNN. In general, a CNN model contains three basic layers:

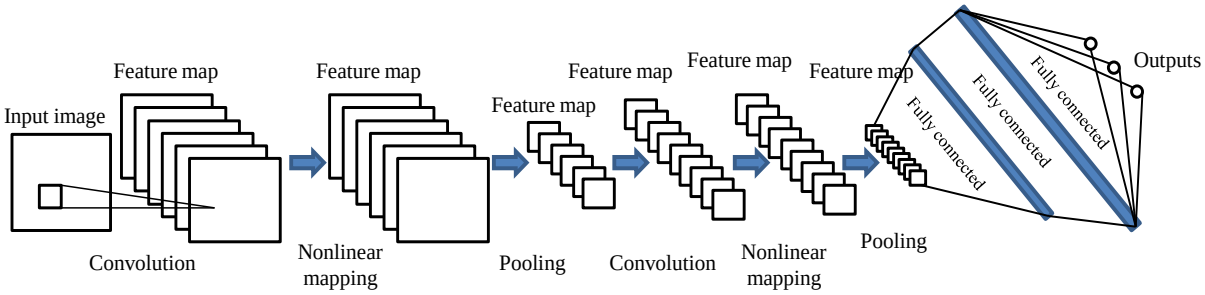


Figure 2.8: Schematic illustration of a typical CNN model. It contains basic building blocks, including convolution, nonlinear mapping, pooling, etc.

- Convolution layer. This layer is to use one or several filters with small size (3×3, 5×5 or 7×7) to convolve the input images, generating different feature maps for subsequent processing. These filters are not fixed but can be automatically learned by the back-propagation algorithm. Thus, well learned filters can extract different correlations in natural images for more accurate modeling.

- Nonlinear mapping layer/activation layer. This layer is to transform the input feature map through nonlinear functions, such as sigmoid, tanh, or ReLU. The nonlinear mapping layer is important, since a neural network with any number of layers is equal to the one with just one layer if there is no nonlinear mapping. In addition, nonlinear mapping makes the CNN extract more complex correlations in natural images.

- Pooling layer. This layer is to reduce dimensionality of input feature maps, making the extracted features compact. Furthermore,large distance correlations in natural images can be captured by pooling the feature map into a small size.

These basic operations indicate that a CNN model has powerful ability to model various correlations in natural images. Different from mathematical models such as GMM that explicitly give the analytical distribution, CNNs implicitly represent these correlations through various neural network architectures.

Chapter 3

Selecting Natural Cover Images for Steganography

3.1 Overview

Natural images have rich and complex structures. These structures are usually hard to be accurately modeled, providing enough space for message hiding. Existing work on steganography focus on designing data embedding algorithms to preserve the structures of natural images. In this work, we turn to in investigate what kinds of structures/images that make the steganography undetectable and improve undetectability of steganography by selecting suitable natural cover images. To address this problem, we propose a new measure based on the KL divergence between covers and stegos to evaluate the hiding ability of a natural images. Experiments on standard datasets validate that, under standard steganalytic methods, the cover image with good hiding ability can improve the performance of various steganographic algorithms. With the proposed measure, we further analyze the properties that intrinsically make stego images undetectable.

3.2 Background and Motivation

There are several interesting works reported that if an appropriate cover image is selected, it will be more difficult to detect the existence of secret image and thus the security of steganography can be largely improved. Sajedi [37] observed that complex cover images, which contains many noisy, textured and cluttered regions, are generally securer for data hiding than those smooth and flat images. Kermani *et al.* [72] validated that the texture information is of great importance in evaluating the hiding ability of the cover image. Kodovsky *et al.* [55] discussed how the texture, spatial frequency and the quality of cover images influence the steganographic security. Sajedi [37] proposed to use steganalytic features to evaluate the embedding capacity of a cover image. Kharrazi *et al.* [74] investigated whether the steganographic security can be improved by using different measures to select cover images, such as the image quality, the number of pixel changes, the mean square error, etc. Penvy *et al.* [99] presented the Maximum Mean Discrepancy measure, which is calculated in image feature spaces, to benchmark steganographic schemes. Although several kinds of image features have been reported to be related to the hiding ability of the corresponding image, what properties that intrinsically determine the hiding ability of an image and make the steganography undetectable remain unclear.

To handle this problem, we propose a new measure to evaluate the hiding ability of the cover image. The main characteristic of the proposed measure is that it is independent of data embedding algorithms, making the analysis for the steganographic security purely from the properties from cover images possible. In theory, we also prove that the proposed measure is an upper bound for the KL divergence both for the spatial domain images and the transformed domain images. This conclusion indicates that the undetectability of steganographic algorithms is improved when a cover image with small measure value is selected for message hiding. Based on the proposed measure, we have analyzed what properties that intrinsically determines the undetectability of

steganography.

The rest of the chapter is organized as follows. In section 3.3, we introduce the proposed measure to evaluate the hiding ability of cover images. Extensive experimental results are reported in section 3.4 to validate the effectiveness of the proposed measure and the properties of cover images that make steganography undetectable is discussed. The chapter is concluded in section 3.5.

3.3 General Framework

In this section, we focus on selecting appropriate cover images to hide secret messages. Fig.3.1 shows the process of cover images selection for steganography. We present a new measure to evaluate the hiding ability of each cover images. Furthermore, the relationship between the proposed measure and the KL divergence is also derived, both in spatial domain or in compressed domain.

Several important notations used in this chapter are listed in Tab.3.1.

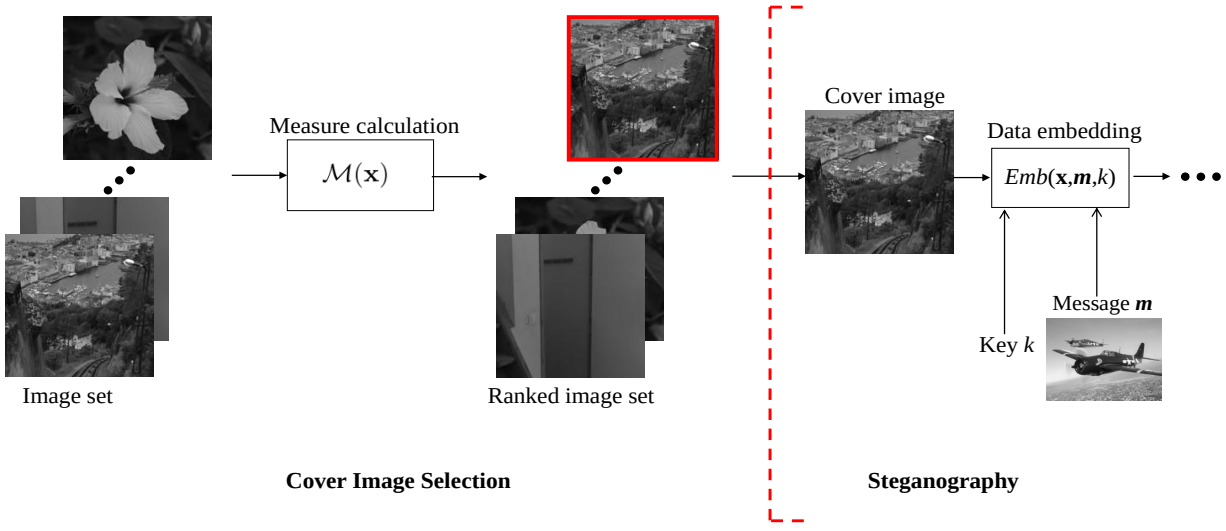


Figure 3.1: Schematic display of the process of cover image selection for steganography. Images with good hiding abilities are selected for hiding secret messages.

Table 3.1: Notations used in this chapter.

Notations	Descriptions
$\mathbf{x}$	The cover image
$\mathbf{y}$	The stego image
$\mathbf{z}$	The message vector
α	The message embedding rate
$\boldsymbol{I}_s$	The image patch in the spatial domain
I_c	The DCT coefficient in the compressed domain
a_i	The i-th coefficient of the Gaussian Mixture Model (GMM)
$\boldsymbol{\mu}_i$	The i-th mean vector in the GMM
Σ_i	The i-th covariance matrix in the GMM
σ	The scale of the Laplace distribution
$\mathcal{M}$	The proposed measure

3.3.1 Proposed Measure

The security of a stegosystem is defined as the KL divergence between the cover image and the stego image [84]:

$$D_{KL}(P||Q) = E_P\left[ln\frac{P}{Q}\right], \tag{3.1}$$

where P and Q are the probability distribution of the cover image $\mathbf{x}$ and the stego image $\mathbf{y}$ respectively, $E_P[\cdot]$ represents the expectation with respect to P. The KL divergence measures the discrepancy between two probability distributions. It is used as a theoretical model for analyzing many aspects relating to steganographic security. For example, Ker in [3] used the KL divergence to derive a *Q-factor* to benchmark binary steganalysis methods. With the KL divergence, Fridrich evaluated how image quantization [51], image scaling [54] affect the security of steganographic algorithms.

Even though the KL divergence is a theoretical model for security evaluation of a stegosystem, it cannot directly be used to evaluate the hiding ability of a cover image. This is because the KL divergence is not only determined by the distribution of cover images but also dependent on the data embedding algorithms. Moreover, direct estimation to the KL divergence is proved to be quite challenging [8].

The measure $\mathcal{M}$ we propose to evaluate the hiding ability of a cover image $\mathbf{x}$ is

calculated as following:

$$\mathcal{M}(\mathbf{x}) = tr\left(J(P(\mathbf{x}))\right), \tag{3.2}$$

where $tr(\cdot)$ represents the trace of a matrix, $J(P(\mathbf{x}))$ is the Fisher information matrix of the probability distribution $P(\mathbf{x})$ of cover image $\mathbf{x}$:

$$J(P(\mathbf{x})) = E_P\left[\left(\frac{\partial lnP(\mathbf{x})}{\partial \mathbf{x}}\right)\left(\frac{\partial lnP(\mathbf{x})}{\partial \mathbf{x}}\right)^T\right], \tag{3.3}$$

where A^T represents the transpose operator to the matrix A. In the next two subsections, we will prove that the following proposition holds for the distributions of spatial domain images and compressed domain images:

Proposition 1 *assume message embedding in steganography can be modeled as:*

$$\mathbf{y} = \mathbf{x} + \alpha\mathbf{z}, \tag{3.4}$$

where $\mathbf{y}$ *represents the stego image with distribution* $Q_\alpha(\mathbf{y})$. $\mathbf{z}$ *represents the message vector, which is mutually independent with* $\mathbf{x}$. α *represents the message embedding rate. The message vector* $\mathbf{z}$ *is assumed to be a random vector with Gaussian distribution* $\mathcal{N}(\mathbf{0}, \mathbf{I})$*, where* $\mathbf{I}$ *denotes the identity matrix. Then we have:*

$$D_{KL}(P(\mathbf{x}))||Q_\alpha(\mathbf{y}))) \leq c \cdot \mathcal{M}\alpha^2, \tag{3.5}$$

where c *is a constant.*

Following the general assumption in most of steganography works[52, 97, 98], the stego signal is modeled as the noise with Gaussian distribution in the analysis.

3.3.2 Proposition Proof for the Spatial Domain Images

In this subsection, we prove the proposition for spatial domain images. Among probability distributions, the GMM shows surprisingly strong performance in modeling the statistics of natural images [26]. Furthermore, any continuous distribution function can be approximated by the GMM [23]. Therefore, we use the GMM to approximate the probability distribution of a given image in the spatial domain:

$$p(\boldsymbol{I}_s) = \sum_{i=1}^{N} a_i \mathcal{N}(\boldsymbol{I}_s; \boldsymbol{\mu}_i, \Sigma_i), \tag{3.6}$$

where $\boldsymbol{I}_s$ denotes the patch of the cover image in the spatial domain, N represents the number of components in GMM, a_i is the i-th coefficient of the GMM model, where $a_i > 0$ and $\sum_{i=1}^{N} a_i = 1$. The Gaussian distribution $\mathcal{N}(\boldsymbol{I}_s; \boldsymbol{\mu}_i, \Sigma_i)$ is parameterized by the expectation $\boldsymbol{\mu_i}$ and the covariance matrix Σ_i:

$$\mathcal{N}(\boldsymbol{I}_s; \boldsymbol{\mu}_i, \Sigma_i) = \frac{exp\left(-\frac{1}{2}(\boldsymbol{I}_s - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (\boldsymbol{I}_s - \boldsymbol{\mu}_i)\right)}{(2\pi)^{d/2} |\Sigma_i|^{1/2}}, \tag{3.7}$$

where d is the dimension of $\boldsymbol{I}_s$, and $|\cdot|$ denotes the determinant of a matrix. In our paper, each patch $\boldsymbol{I}_s$ is centralized by by subtracting its mean $\overline{\boldsymbol{I}}_s$ before learning the parameters of the GMM. Therefore, we have $\boldsymbol{\mu}_i = \mathbf{0}$ and Eq.(3.7) becomes:

$$\mathcal{N}(\boldsymbol{I}_s; \mathbf{0}, \Sigma_i) = \frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} exp\left(-\frac{1}{2} \boldsymbol{I}_s^T \Sigma_i^{-1} \boldsymbol{I}_s\right), \tag{3.8}$$

For GMM, the formula of the proposed measure $\mathcal{M}$ is (see Appendix A):

$$\mathcal{M} = \int_{\boldsymbol{I}_s} \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} \gamma_i \gamma_j \boldsymbol{I}_s^T \Sigma_i^{-1} \Sigma_j^{-1} \boldsymbol{I}_s}{\sum_{k=1}^{N} \gamma_k} d\boldsymbol{I}_s, \tag{3.9}$$

where γ_i is defined as:

$$\gamma_i = a_i \mathcal{N}(\boldsymbol{I}_s; \mathbf{0}, \Sigma_i) \tag{3.10}$$

The KL divergence between the cover image with the probability $p(\boldsymbol{I}_s)$, and its stego version with the probability $q_\alpha(\boldsymbol{I}_s)$, can be written as:

$$D_{KL}\left(p(\boldsymbol{I}_s)||q_\alpha(\boldsymbol{I}_s)\right) = H\left(p(\boldsymbol{I}_s), q_\alpha(\boldsymbol{I}_s)\right) - H\left(p(\boldsymbol{I}_s)\right), \tag{3.11}$$

where $H\left(p(\boldsymbol{I}_s), q_\alpha(\boldsymbol{I}_s)\right)$ denotes the cross entropy between the cover image and its stego image:

$$H\left(p(\boldsymbol{I}_s), q_\alpha(\boldsymbol{I}_s)\right) = -\int_{\boldsymbol{I}_s} p(\boldsymbol{I}_s) ln\left(q_\alpha(\boldsymbol{I}_s)\right) d\boldsymbol{I}_s, \tag{3.12}$$

and $H\left(p(\boldsymbol{I}_s)\right)$ denotes the entropy of the cover image:

$$H\left(p(\boldsymbol{I}_s)\right) = -\int_{\boldsymbol{I}_s} p(\boldsymbol{I}_s) ln\left(p(\boldsymbol{I}_s)\right) d\boldsymbol{I}_s, \tag{3.13}$$

Eq.(3.11) could be rewritten as:

$$\begin{aligned} D_{KL}\left(p(\boldsymbol{I}_s)||q_\alpha(\boldsymbol{I}_s)\right) = {} & H\left(p(\boldsymbol{I}_s), q_\alpha(\boldsymbol{I}_s)\right) - H(q_\alpha(\boldsymbol{I}_s)) \\ & + H(q_\alpha(\boldsymbol{I}_s)) - H\left(p(\boldsymbol{I}_s)\right), \end{aligned} \tag{3.14}$$

when the embedding rate α is small, the first term on right side of Eq.(3.14) can be rewritten as (see Appendix B):

$$H\left(p(\boldsymbol{I}_s), q_\alpha(\boldsymbol{I}_s)\right) - H(q_\alpha(\boldsymbol{I}_s)) = \varepsilon\alpha^2 + o(\alpha^2), \tag{3.15}$$

where ε is:

$$\varepsilon = -\frac{\int_{\boldsymbol{I}_s}\left[\sum_{i=1}^{N}\gamma_i \boldsymbol{I}_s^T \Sigma_i^{-2} \boldsymbol{I}_s\right] ln\left(\sum_{i=1}^{N}\gamma_i\right) d\boldsymbol{I}_s}{2}, \tag{3.16}$$

For the second term on the right side of Eq.(3.14), it can be rewritten as (see Appendix C):

$$H\left(q_\alpha(\boldsymbol{I}_s)\right) - H(p(\boldsymbol{I}_s)) = \mathcal{M}\alpha^2 + o(\alpha^2), \tag{3.17}$$

we can further prove that (see Appendix D):

$$2\varepsilon \leq \widetilde{K}_s \cdot \mathcal{M}, \tag{3.18}$$

where $\widetilde{K}_s$ is a constant depending on the number of GMM components N, the maximum value of γ, and the eigenvalue of all inverse covariance matrices $\{\Sigma_i^{-1}\}_{i=1}^N$ (see Appendix C). With Eq.(3.14), Eq.(3.15) and Eq.(3.17), we can obtain that:

$$D_{KL}\left(p(\boldsymbol{I}_s)||q_\alpha(\boldsymbol{I}_s)\right) = (\mathcal{M}+\varepsilon)\alpha^2 + o(\alpha^2), \tag{3.19}$$

Combining Eq.(3.18) and Eq.(3.19), then:

$$D_{KL}\left(p(\boldsymbol{I}_s)||q_\alpha(\boldsymbol{I}_s)\right) \leq \left(1+\frac{\widetilde{K}_s}{2}\right)\cdot \mathcal{M}\alpha^2 + o(\alpha^2), \tag{3.20}$$

Since $\mathcal{M}$ is positive, it follows that $o(\alpha^2) \leq \mathcal{M}\alpha^2$ when α is small. Then Eq.(3.20) can be rewritten as:

$$\begin{aligned} D_{KL}\left(p(\boldsymbol{I}_s)||q_\alpha(\boldsymbol{I}_s)\right) &\leq \left(1+\frac{\widetilde{K}_s}{2}\right)\cdot \mathcal{M}\alpha^2 + \mathcal{M}\alpha^2 \\ &= \left(2+\frac{\widetilde{K}_s}{2}\right)\cdot \mathcal{M}\alpha^2 \end{aligned} \tag{3.21}$$

The proposition for the GMM model is proved.

3.3.3 Proposition Proof for the Compressed Domain Images

Many works show that DCT coefficients are best approximated by the Laplacian distributions [86]. It becomes the dominant choice to model the distribution of DCT coefficients which balances simplicity of the model and fidelity to the empirical data [29]. In this subsection, we derive the proposed measure and prove its theoretical

effectiveness based on the Laplacian distribution.

The probability density function of a Laplacian distribution for DCT coefficients can be written as:

$$p(I_c) = \frac{1}{2\sigma} exp\left(-\frac{|I_c|}{\sigma}\right), \tag{3.22}$$

where I_c denotes the value of a DCT coefficient, σ is a positive parameter. When adding a Laplace distributed DCT coefficient I_c with a Gaussian distributed random variable $\mathcal{N}(0, \alpha^2)$, the coefficient then follows a new distribution $q_c(I_c)$, called the Normal-Laplace distribution [60]:

$$\begin{aligned} q_\alpha(I_c) = \frac{1}{2\sigma} exp\left(\frac{\alpha^2}{2\sigma^2}\right) \Bigg[& e^{I_c/\sigma} \Phi\left(-\frac{\sigma I_c + \alpha^2}{\sigma\alpha}\right) \\ & + e^{-I_c/\sigma} \Phi\left(\frac{\sigma I_c - \alpha^2}{\sigma\alpha}\right)\Bigg], \end{aligned} \tag{3.23}$$

where $\Phi(x)$ represents the cumulative distribution function (cdf) of a standard normal distribution:

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-t^2/2} dt, \tag{3.24}$$

For Eq.(3.23), it can be approximated as the following equation when α is small (see Appendix E):

$$\begin{aligned} q_\alpha(I_c) &\approx \frac{1}{2\sigma} exp\left(\frac{\alpha^2}{2\sigma^2} - \frac{|I_c|}{\sigma}\right) \\ &= p(I_c) exp\left(\frac{\alpha^2}{2\sigma^2}\right), \end{aligned} \tag{3.25}$$

For the Laplace distribution, $\mathcal{M}$ has a close expression:

$$\mathcal{M} = \int_{-\infty}^{+\infty} \left(\frac{\partial ln(p(I_c))}{\partial I_c}\right)^2 p(I_c) dI_c = \frac{1}{\sigma^2}, \tag{3.26}$$

Similar to the Eq.(3.11), the KL divergence between $p(I_c)$ and $q_\alpha(I_c)$ can be written

as:

$$D_{KL}(p(I_c)||q_\alpha(I_c)) = H(p(I_c), q_\alpha(I_c)) - H(p(I_c)), \tag{3.27}$$

where $H(p(I_c), q_\alpha(I_c))$ denotes the cross entropy between $p(I_c)$ and $q_\alpha(I_c)$:

$$H(p(I_c), q_\alpha(I_c)) = -\int_{-\infty}^{+\infty} p(I_c) ln\left(q_\alpha(I_c)\right) dI_c, \tag{3.28}$$

Substituting Eq.(3.22) and Eq.(3.25) into Eq.(3.28), we can obtain:

$$\begin{aligned} H(p(I_c), q_\alpha(I_c)) &= -\int_{-\infty}^{+\infty} p(I_c) ln\left(p(I_c)\right) dI_c \\ &= -\int_{-\infty}^{+\infty} p(I_c) ln\left(p(I_c) exp\left(\frac{\alpha^2}{2\sigma^2}\right)\right) dI_c \\ &= H(p(I_c)) - \frac{\alpha^2}{2\sigma^2}\int_{-\infty}^{+\infty} p(I_c) dI_c \\ &= H(p(I_c)) - \frac{\alpha^2}{2\sigma^2}, \end{aligned} \tag{3.29}$$

Thus, the $D_{KL}(p(I_c)||q_\alpha(I_c))$ can be simplified as:

$$\begin{aligned} D_{KL}(p(I_c)||q_\alpha(I_c)) &= H(p(I_c), q_\alpha(I_c)) - H(p(I_c)) \\ &\leq |H(p(I_c), q_\alpha(I_c)) - H(p(I_c))| \\ &= \frac{\alpha^2}{2\sigma^2}, \end{aligned} \tag{3.30}$$

Since $\mathcal{M} = 1/\sigma^2$, then:

$$D_{KL}(p(I_c)||q_\alpha(I_c)) \leq \mathcal{M}\alpha^2, \tag{3.31}$$

Figure 3.2: Sample images in BOSSbase ver 1.01.

Figure 3.3: Sample images in MIR Flickr.

3.4 Experiments

This section presents three experiments conducted to assess the effectiveness of the proposed measure. In the first experiment, we numerically validate the Eq.(3.5) for the GMM model and the Laplace distribution. The effectiveness of the proposed measure is demonstrated for the spatial domain images and compressed domain images in the second and the third experiments respectively.

Two datasets are used in our experiment. The first dataset is the BOSSbase ver 1.01 dataset [76], which is a standard dataset for evaluating steganographic algorithms. The dataset consists of 10,000 grayscale natural images with the size of 512×512. Fig.3.2 shows several sample images of the dataset.

In order to apply the proposed measure to the real images in the social network, we choose the MIR Flickr [71] as the second dataset. The dataset is a collection of 25,000 JPEG images from the Flickr website which are redistributable for research purposes and represent a real community of users in the image content. Fig.3.3 demonstrates several sample images in the MIR Flickr dataset.

For parameter setting of GMM, we set N, i.e. the number of components, as 100. The covariance matrices $\{\Sigma_i\}_{i=1}^N$ and the coefficients $\{a_i\}_{i=1}^N$ in GMM are learned by the efficient online Expectation Minimization (EM) method [85]. 10,000 centralized image patches with the size of 5×5 are uniformly sampled from a given image for

learning the parameters of the GMM model. Since the proposed measure cannot be calculated directly, we use its empirical estimation $\widetilde{\mathcal{M}}$ to approximate the measure $\mathcal{M}$:

$$\widetilde{\mathcal{M}} = \frac{1}{K}\sum_{k=1}^{K}\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}\gamma_i^k\gamma_j^k}{\left(\sum_{i=1}^{N}\gamma_i^k\right)^2}(\boldsymbol{I}_s^k)^T\Sigma_i^{-1}\Sigma_j^{-1}\boldsymbol{I}_s^k, \tag{3.32}$$

where $\boldsymbol{I}_s^k$ denotes the k-th patch sampled from Eq.(3.7) using the method introduced in [29], γ_i^k is defined as:

$$\gamma_i^k = \frac{a_i}{(2\pi)^{d/2}|\Sigma_i|^{1/2}}exp\left(-\frac{(\boldsymbol{I}_s^k)^T\Sigma_i^{-1}\boldsymbol{I}_s^k}{2}\right), \tag{3.33}$$

The number of random samples K used for measure estimation is set to 10,000. **Algorithm 1** gives the procedure to estimate the proposed measure for spatial domain images.

Algorithm 1: Measure estimation for spatial images

Input : Cover image $\boldsymbol{I}_s$

Output: Estimated measure $\widetilde{\mathcal{M}}$

1 Decompose cover image $\boldsymbol{I}_s$ into 10,000 patches with the size of 5×5

2 Use the online EM algorithm to estimate the parameters of GMM, and gives the distribution of $\boldsymbol{I}_s$

$$p(\boldsymbol{I}_s) = \sum_{i=1}^{N} a_i\mathcal{N}(\boldsymbol{I}_s; \boldsymbol{\mu}_i, \Sigma_i)$$

3 Calculate the measure for $\boldsymbol{I}_s$ according to:

$$\widetilde{\mathcal{M}} = \frac{1}{K}\sum_{k=1}^{K}\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}\gamma_i^k\gamma_j^k}{\left(\sum_{i=1}^{N}\gamma_i^k\right)^2}(\boldsymbol{I}_s^k)^T\Sigma_i^{-1}\Sigma_j^{-1}\boldsymbol{I}_s^k$$

For Laplace distribution, the parameter σ is estimated by $\widetilde{\sigma}$:

$$\widetilde{\sigma} = \frac{1}{N}\sum_{i}^{N}|I_{ci} - \mu_{I_c}|, \tag{3.34}$$

where I_{ci} represents the i-th DCT coefficient, μ_{I_c} is the expectation of $\{I_{ci}\}_{i=1}^{N}$. Since

only AC components are used for steganography, all DC components are set to 0. Based on the estimated $\widetilde{\sigma}$, the measure for JPEG images is calculated as:

$$\widetilde{\mathcal{M}} = \frac{1}{\widetilde{\sigma}^2}, \tag{3.35}$$

Algorithm 2 gives the procedure to estimate the proposed measure for JPEG images.

Algorithm 2: Measure estimation for JPEG images

Input : JPEG cover image I_c
Output: Estimated measure $\widetilde{\mathcal{M}}$

1 Get the DCT coefficients of the cover image I_c
2 Estimate the parameter Laplace distribution using the formula:

$$\widetilde{\sigma} = \frac{1}{N}\sum_{i}^{N} |I_{ci} - \mu_{I_c}|$$

3 Calculate the measure for the JPEG image I_c as:

$$\widetilde{\mathcal{M}} = \frac{1}{\widetilde{\sigma}^2}$$

3.4.1 Demonstration of Theoretical Results

In this experiment, we demonstrate the derived inequalities on the BOSSbase and the MIR Flickr. For each dataset, five images ranging from visually complex to simple are selected for demonstration. The measure $\mathcal{M}$ is calculated according to Eq.(3.32) and Eq.(3.35) for spatial domain images and compressed domain images respectively. For stego images, we firstly generate random messages according to Gaussian distribution $\mathcal{N}(0, \alpha^2\mathbf{I})$. Then these messages are added to the pixels or the DCT coefficients of cover images, which generated stego images following the assumption in the proposition.

The distributions of five BOSSbase cover images and five MIR Flickr images are estimated by the GMM model. For the BOSSbase images, the patch size is set to 5×5;

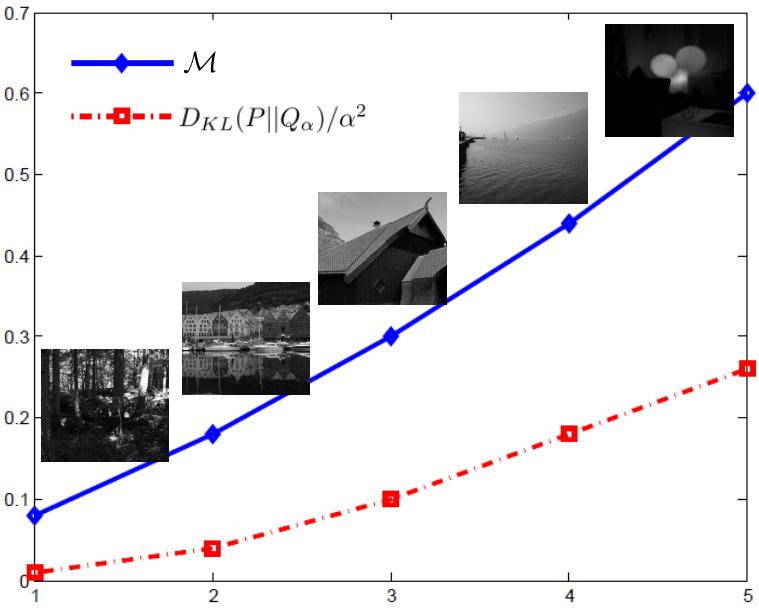


Figure 3.4: The comparison of the scores of the proposed measure and the KL divergence for the BOSS dataset. The solid blue curves reflect the value of proposed measure $\mathcal{M}$, and dotted red curve show the value of $D_{KL}(P||Q_\alpha)/\alpha^2$.

while the patch size of MIR Flickr images is set to 8×8 in order to make the setting consistent with the size of a basic compression block of JPEG. Assume a cover image $\mathbf{x}$ ($\mathbf{x}$ can be a spatial domain image or JPEG image) among ten selected images has a distribution:

$$p(\mathbf{x}) = \sum_{i=1}^{N} a_i \mathcal{N}(\mathbf{x}; \mathbf{0}, \Sigma_i), \tag{3.36}$$

According to the result in Appendix B, the distribution of its stego image is:

$$q_\alpha(\mathbf{x}) = \sum_{i=1}^{N} a_i \mathcal{N}(\mathbf{x}; \mathbf{0}, \Sigma_i + \alpha^2 \mathbf{I}), \tag{3.37}$$

Since the KL divergence between $p(\mathbf{x})$ and $q_\alpha(\mathbf{x})$ cannot be calculated directly, we use the empirical estimation to approximate the $D_{KL}(p(\mathbf{x})||q_\alpha(\mathbf{x}))$:

$$\hat{D}_{KL}(p(\mathbf{x})||q_\alpha(\mathbf{x})) = \frac{1}{L} \sum_{i=1}^{L} ln \left(\frac{p(\mathbf{x}_i)}{q_\alpha(\mathbf{x}_i)} \right), \tag{3.38}$$

where L denotes the number of samples used for estimation, $\mathbf{x}_i$ is the i-th patch sampled

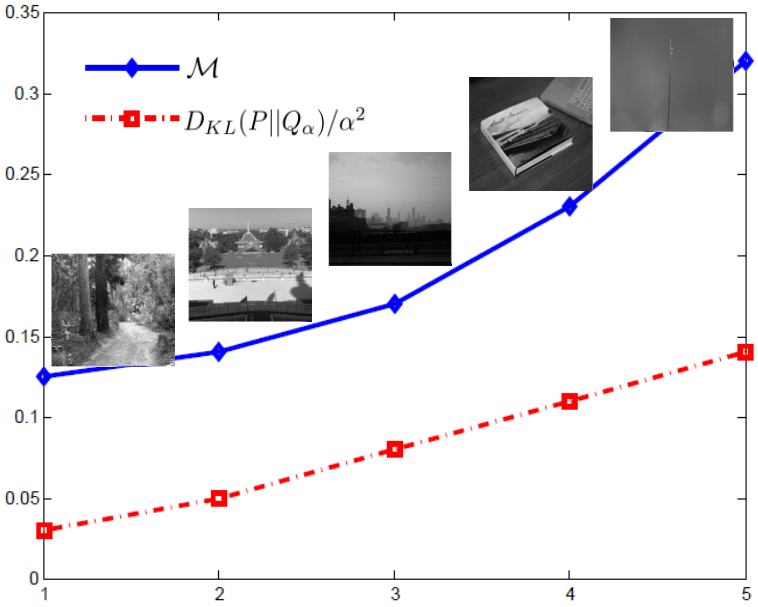


Figure 3.5: The comparison of the scores of the proposed measure and the KL divergence for the MIR-Flickr dataset. The solid blue curves reflect the value of proposed measure $\mathcal{M}$, and dotted red curve show the value of $D_{KL}(P||Q_\alpha)/\alpha^2$.

from $p(\mathbf{x})$. In the experiment, α is chosen as a small value, 0.1. The final estimated $\hat{D}_{KL}(p(\mathbf{x})||q_\alpha(\mathbf{x}))$ is the averaging result of ten times running of Eq.(3.38).

Experimental results are shown as Fig.3.4 and Fig.3.5. It is easy to find that, $\mathcal{M}$ is always larger than $D_{KL}(p||q_\alpha)/\alpha^2$, validating the correctness of the proved inequalities. Additionally, a cover image with small $\mathcal{M}$ leads to a small KL divergence, which further demonstrates the effectiveness of the proposed measure.

3.4.2 Cover Image Selection for Steganography in Spatial Domain

In this experiment, we conduct the proposed measure on the BOSSbase. For steganography, four state-of-the-art algorithms are used for performance evaluation: Least Significant Bit Matching revisiting (LSBM-r) [58], Edge Adaptive steganography (EA)[112], Highly Undetectable steGanOgraphy (HUGO) [102] and the Spatial UNIversal WAvelet Relative Distortion (S-UNIWARD) [107]. For steganalysis, the Spatial Rich Model (S-

RM) based steganalysis is selected for its excellent performances in attacking many steganographic algorithms. In our implementation, 5,000 randomly selected images in BOSSbase are used for training SRM based ensemble classifier [56] and the rest 5,000 images are for testing. The security performance is evaluated by the detection error P_E:

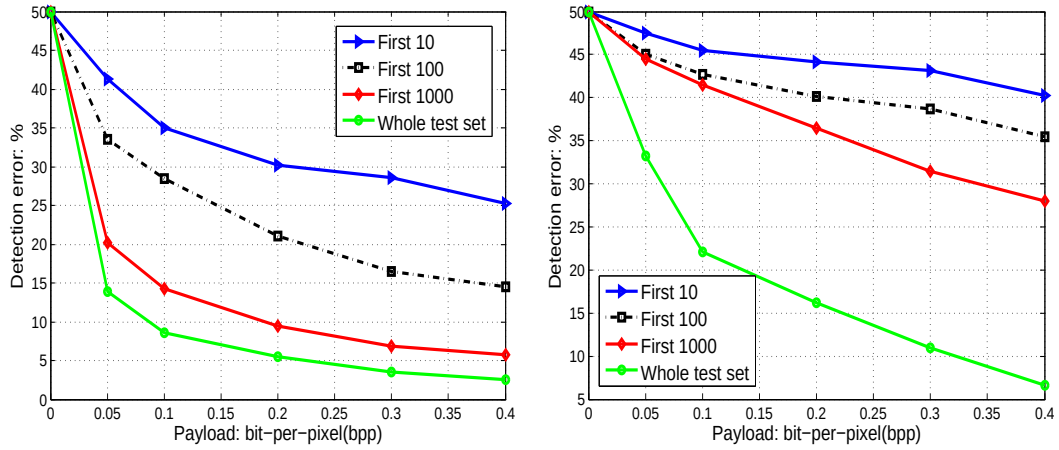
$$P_E = \frac{1}{2}(P_{MD} + P_{FA}) \tag{3.39}$$

where P_{MD} is the miss detection probability and P_{FA} represents the false alert probability. We use this evaluation standard because it is widely used in modern steganalysis [54].

Before evaluation, the proposed measure $\mathcal{M}$ for each image in the test set is calculated according to Eq.(3.32). Then all these images are sorted in an ascending manner. To prove the effectiveness of the proposed measure, we select the first r cover images with high hiding ability, where r is chosen as 10, 100, 1,000 and 5,000 (whole test set). The prediction error is the average of ten times running based on Eq.(3.39).

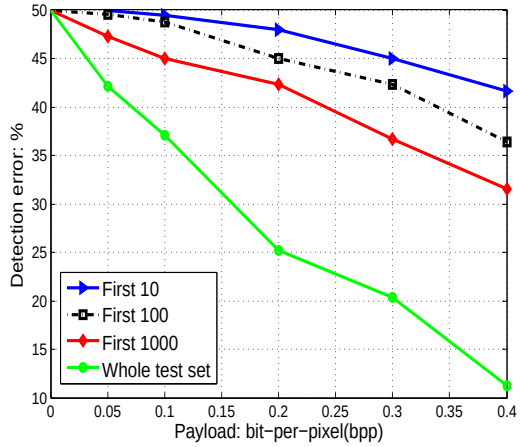
For the experiment, we evaluate security performances of four different steganographic algorithms based on the SRM steganalysis. The purpose is to investigate how the detection error P_E changes if top secure images are selected as the covers. Higher detection error indicates securer cover images and the vice versa. The experiment is conducted on different payloads, where the payload is defined as the division between the length of hidden messages and the dimension of the cover image, bit-per-pixel (bpp). Random binary messages are embedded to the cover images according to different steganographic algorithms. We follow the general settings to the payload in image steganography. Fig.3.6 shows the detection error curves. Experimental results show that, when the first 10 secure images are selected as covers, the detection errors are high for four steganographic algorithms at five different payloads.

Observing experimental results as Fig.3.6, we can find that the detection error in-

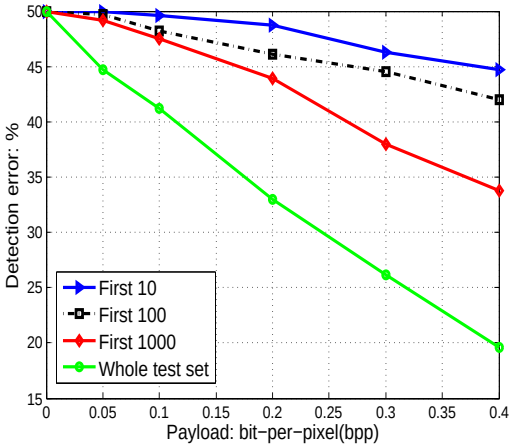


(a) LSBM-r

(b) EA



(c) HUGO



(d) S-UNIWARD

Figure 3.6: Average detection errors P_E for LSBM-r, EA, HUGO and S-UNIWARD. Four different settings are investigated: first 10 images, first 100 images, first 1,000 images, the whole test dataset. Here, first r represent r highest ranked images according to the proposed measure.

creases monotonously as the measure value decreases, for all steganographic algorithms. This fact proves that the proposed measure successfully reflects the security level of a cover image. To further verify the effectiveness of the measure, we use a parametric model to approximate the curve of detection error:

$$P_E = 0.5 - p^{\alpha} \tag{3.40}$$

where α is the parameter, $\alpha \geq 0$. p represents the payload and in our experiment,

$0 \leq p \leq 0.4$. By taking derivatives on both sides of Eq.(5.40), we get:

$$dP_E = -\alpha p^{\alpha-1} dp \tag{3.41}$$

Table 3.2: Estimated α for four steganographic algorithms. The dataset with 5000 represents the whole test set.

Algorithms	10	100	1000	5000
LSBM-r	0.788	0.606	0.559	0.542
EA	1.361	1.214	1.007	0.694
HUGO	2.410	1.785	1.429	0.974
S-UNIWARD	3.609	2.231	2.002	1.714

According to this equation, the detection error drops slowly as the increase of p when α is large and p is near to zero. Therefore, a good steganographic algorithm should have an error curve with large α to make it immune to data embedding. The fitted parameters to the curve are listed in Tab.3.2. Results as this table show that α of four steganographic algorithms systematically increases to be a larger one if more secure images are chosen as covers.

We also compare the proposed measure with several other cover image selection methods. Three measures, the mean square error based cover selection (MSE-sel), number of pixel changes based cover selection (Change-sel) and the local prediction error based cover selection (Local-sel), are chosen for comparison. Details about these algorithms are introduced in [73]. We choose these methods because they achieve promising performances in improving steganographic security. Tab.3.3 shows detection errors of four steganographic algorithms for the first 10 selected images at 0.2 bpp payload. The results prove that the proposed measure outperforms all other three measures. Mostly important, the detection error is near to random guessing if messages are hidden in the images selected by the proposed measure, which is meaningful for practical application.

Table 3.3: Detection errors for other measures: MSE-sel, Change-sel and Local-sel. All schemes select the first 10 secure images according to their measures. The payload is chosen as 0.2 bpp.

Algorithms	MSE-sel	Change-sel	Local-sel	$\mathcal{M}$
LSBM-r	10.5%	12.8%	25.4%	**30.0%**
EA	21.0%	19.4%	38.0%	**44.7%**
HUGO	35.5%	31.7%	43.8%	**48.2%**
S-UNIWARD	33.6%	35.9%	45.2%	**49.3%**

3.4.3 Cover Image Selection for Steganography in Compressed Domain

In this experiment, we conduct the proposed measure for MIR Flickr JPEG images. Three steganographic algorithms are used for evaluation: JPEG steganography [61] (Jsteg), nsF5 steganography [50] and JPEG UNInversal WAvelet Relative Distortion (J-UNIWARD) steganography [107]. The payload is set to 0.1 bit per AC coefficient (bpac), where random binary messages are embedded to the JPEG images according to three steganographic algorithms. For steganalysis, the recently proposed Discrete Cosine Transform Residual (DCTR)[105] based steganalysis is used for detection. Similarly to the experiment in spatial domain, Eq.(5.39) is chosen for performance evaluation. 20000 images from MIR Flickr are used for training DCTR based classifier and the rest images are used for testing. Since GMM can approximate any probability distribution, we also use the GMM model to approximate the probability distribution of DCT coefficients. The parameter setting follows the first experiment for MIR Flickr images. The GMM based measure and Laplace distribution based measure are compared in this experiments.

Tab.3.4 shows the results, the performance of three algorithms is obviously improved if good hiding ability images are selected for steganography. In addition, the Laplace based measure outperforms the GMM based measure, which demonstrates that Laplace

Table 3.4: Detection errors for Jsteg, nsF5 and J-UNIWARD with 0.1 bpac, MIR Flickr dataset.

Methods	**Mode**	10	100	1000	5000
Jsteg	Laplace	38%	32.6%	28.6%	25.4%
	GMM	36%	31.5%	28.5%	25.4%
nsF5	Laplace	36.5%	31.5%	29.1%	21.5%
	GMM	33%	30.4%	28.3%	21.5%
J-UNIWARD	Laplace	50%	48%	46.5%	44.8%
	GMM	47.5%	46%	45.4%	44.8%

distribution is more suitable than the GMM model to calculate the proposed measure for JPEG images.

In feature space, images with excellent hiding ability are hard to be discriminated from their stego versions. In order to observe discriminability between cover images and their stegos in feature space, we extract SRM features of 500 best images and 500 worst images. Then Principle Component Analysis (PCA) is used to project high dimensional SRM features into 2 dimensional vectors. The results are shown as Fig.3.7. Obviously, SRM features of best cover images and their stegos are mixed with each other, while they can be easily discriminated for the worst case.

As we known, the larger the number of secret bits to be embedded, the easier the stego-image is detected by steganalysis algorithms. The maximal secure payload [52] that refers to the hiding capacity is obtained when the stego-image degradation becomes detectable. Thus, the maximal secure payload can be used as a criterion to evaluate the hiding ability of cover images.

To prove the effectiveness of the proposed measure, we select 100 cover images with top hiding ability, medium hiding ability and low hiding ability images, respectively. To find the secure payload of three kinds of images, we increase the size of hidden messages from 0 to a maximum value on the constraint that the detection rate is less than 60%. This maximum value is regarded as the secure payload. The classifier used for detection is trained based on 1,000 randomly selected images and their corresponding

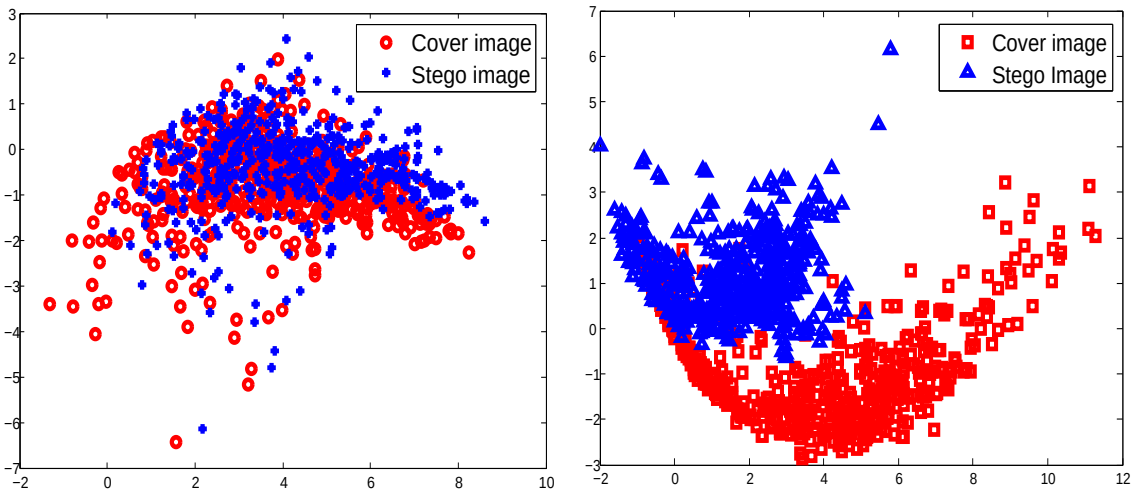


Figure 3.7: 2-D representations for SRM features from cover image and their stego image. Each cover image is embedded by random message using HUGO, with payload 0.4 bpp. SRM features of first 500 cover images and last 500 cover images, ranked by the measure $\mathcal{M}$, are extracted and projected onto 2-D. (a) Visualization on 2-D principal component plane for SRM features of first 500 cover images; (b) Visualization on 2-D principal component plane for SRM features of last 500 cover images.

stego images.

Table 3.5: The average value of maximal secure payload for three steganographic algorithms.

Methods	Top 100	Medium 100	Low 100
Jsteg	0.063 bpAC	0.029 bpAC	0.010 bpAC
nsF5	0.073 bpAC	0.035 bpAC	0.014 bpAC
J-UNIWARD	0.203 bpAC	0.115 bpAC	0.042 bpAC

On MIR Flickr image dataset, the average value of maximal secure payload capacities according to three popular steganographic algorithms for the images with high, middle, and low hiding ability are shown in Tab.3.5. From this table, we can find the maximum secure payloads are improved if good hiding ability images are selected as cover images, which also evidence the effectiveness of the proposed measure.

3.4.4 What Makes the Stego Image Undetectable ?

To investigate what properties of cover images determine steganographic security in spatial domain, in this section, we introduce two variable factors: entropy variable factor and energy variable factor. Assume the image is modeled by the GMM as Eq. (3.6), the entropy variable factor S is defined as Eq.(3.42), and the energy variable factor E is defined as Eq.(3.43):

$$S = -\sum_{i=1}^{L} a_i ln(a_i) \tag{3.42}$$

$$E = \sum_{i=1}^{L} a_i tr\left(\Sigma_i\right) \tag{3.43}$$

In Fig.3.8 and Fig.3.9, we demonstrate several sample images with different values of these two variable factors, including: entropy variable factor and energy variable factor. From Fig.3.8, we can find the image with low value of entropy variable factor is less clustered, less textured, and highly redundant than the image with high value of the same factor. This result is consistent with the definition of entropy variable factor in Eq.(3.42), which is used to measure the disorder of the coefficient in the GMM model. To the image with high value of energy variable factor, it is obviously that the image is highly cluttered and highly textured than the image with low value of the same factor. Furthermore, the intensity of different patches in high value image changes more than that of the other images. By taking the definition of energy variable factor in Eq.(3.43) into consideration, we believe that the energy variable factor, compared with the entropy variable factor, is more relevant with the diversity of the intensity in different patches.

To further investigate the effect of these two variable factors on the proposed measure, we calculate the Pearson correlations and corresponding p-values. From the results in Tab.3.6, both factors show a significant medium negative correlation with

Table 3.6: Pearson correlations and partial correlations between the proposed measure and two variable factors investigated on an image-by-image basis.

Methods	Pearson correlation		Partial correlation	
Terms	S	E	S	E
r	-0.3686	-0.5157	-0.1077	-0.4184
p	< 0.001	< 0.001	< 0.001	< 0.001

the proposed measure, which are in line with our expectations. We find there exists a positive correlation between two variable factors. Accordingly, we also compute the partial correlations between one variable factor and the proposed measure with another factor as a control variable. Partial correlation coefficients and corresponding p-values are also listed in Tab.3.6. All of the variables showed significant correlations, though not always strong ones. It ensures that both entropy and energy variable factors are effective to make the spatial domain stego image undetectable.

(a) High S (b) Low S

Figure 3.8: Sample images with low and high values of entropy variable factor S.

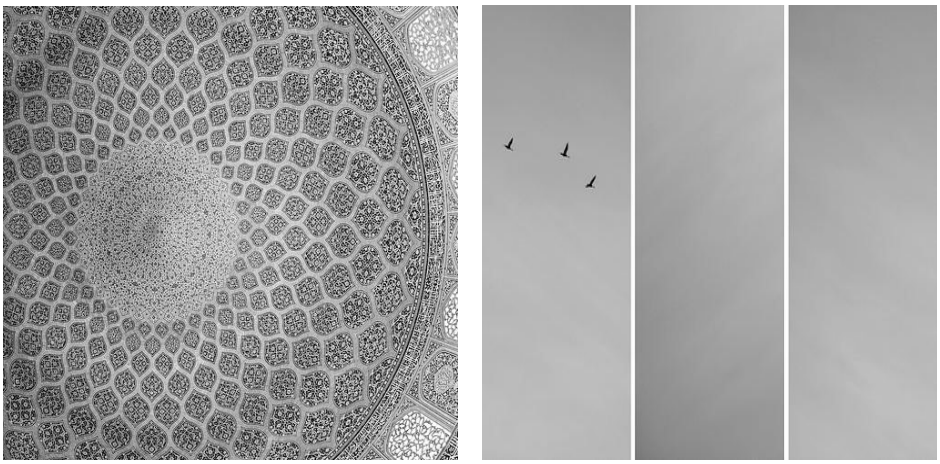
For the compressed domain images, another variable factor based on the DCT coefficients is introduced here. The nonzero DCT coefficient ratio is defined as the ratio between the number of non-zero DCT coefficients to all number of DCT coefficients:

$$C = \frac{n_1}{n_0} \tag{3.44}$$

(a) High E (b) Low E

Figure 3.9: Sample images with low and high values of energy variable factor E.

where n_1 denotes the number of non-zero DCT coefficients, n_0 is the total number of DCT coefficients.



(a) High C (b) Low C

Figure 3.10: Sample images with different values of nonzero DCT coefficient ratio C.

Fig.3.10 demonstrates two sample images with low or high values for the nonzero DCT coefficient ratio. The image with high C also possesses high value on the proposed measure in compressed domain, and vice versa. To provide a sense of C for the proposed measure, we also correlate image-by-image C with the proposed measure. The correlation is significant with a negative correlation coefficient of -0.663 ($p < 0.001$). To any image, when the number of non-zero DCT coefficients increases, the variance of

the corresponding Laplacian distribution will also increase, which is the inverse of our proposed measure. Therefore, the observed negative correlationship between C and the proposed measure is easy to understand. Overall, these results can support that the nonzero DCT coefficient ratio can be used to explain what makes the compressed domain stego image undetectable.

Fig 3.11 demonstrates several sample images evaluated by the proposed measure, including high hiding ability, middle hiding ability and low hiding ability images.

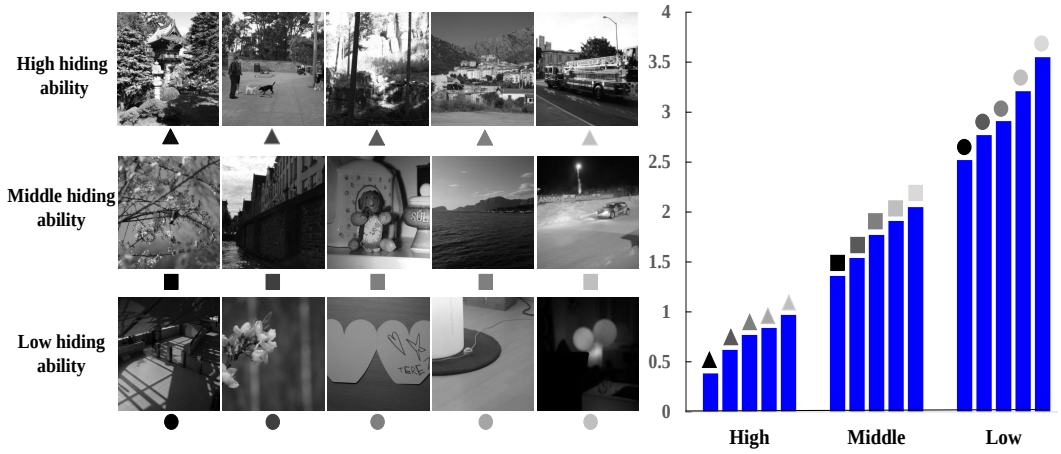


Figure 3.11: Demonstration for the images with high, middle and low hiding ability.

3.5 Summary

This chapter aims to improve steganography security by selecting the cover images with good hiding ability. We propose a novel measure based on information theoretic model of steganography, the KL divergence. We demonstrate the effectiveness of the proposed measure by testing on various steganography and steganalysis techniques in both spatial domain and compressed domain. We conclude that:

- The cover images selected by the proposed measure improve the performance of steganography techniques obviously.
- The proposed measure outperforms other existing cover image selection techniques.
- The cover images selected by the proposed measure have the common statistic character: for spatial domain images, the entropy of the GMM coefficients is high; for transform domain images, the number of nonzero DCT coefficients is high. These observations explain why the cover images with complex texture, cluttered visual content, and low spatial redundancy, are recognized as the images with good hiding ability by the previous works. It also indicates that the proposed model could be considered as the generalization of the existing hiding ability measure.

Appendix A

According to Eq.(3.3), the Fisher information matrix of GMM model is:

$$J(p(\boldsymbol{I}_s)) = E_{p(\boldsymbol{I}_s)}\left[\left(\frac{\partial ln\ p(\boldsymbol{I}_s)}{\partial \boldsymbol{I}_s}\right)\left(\frac{\partial ln\ p(\boldsymbol{I}_s)}{\partial \boldsymbol{I}_s}\right)^T\right], \tag{3.45}$$

Based on the Eq.(3.6), we can derive that:

$$\frac{\partial ln\ p(\boldsymbol{I}_s)}{\partial \boldsymbol{I}_s} = \frac{1}{\sum_{i=1}^{N}\gamma_i}\sum_{i=1}^{N}\frac{\partial \gamma_i}{\partial \boldsymbol{I}_s}, \tag{3.46}$$

where $\gamma_i = a_i\mathcal{N}(\boldsymbol{I}_s; \boldsymbol{\mu}_i, \Sigma_i)$. For γ_i, we can derive that:

$$\frac{\partial \gamma_i}{\partial \boldsymbol{I}_s} == -\gamma_i\Sigma_i^{-1}\boldsymbol{I}_s, \tag{3.47}$$

Combining Eq.(3.46) and Eq.(3.47), Eq.(3.45) can be written as:

$$J(p(\boldsymbol{I}_s)) = E_{p(\boldsymbol{I}_s)}\left[\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}\gamma_i\gamma_j\Sigma_i^{-1}\boldsymbol{I}_s\boldsymbol{I}_s^T\Sigma_j^{-1}}{\left(\sum_{i=1}^{N}\gamma_i\right)^2}\right], \tag{3.48}$$

Since $\mathcal{M}$ is the trace of $J(p(\boldsymbol{I}_s))$, we can conclude that:

$$\mathcal{M} = E_{p(\boldsymbol{I}_s)}\left[\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}\gamma_i\gamma_j\boldsymbol{I}_s^T\Sigma_i^{-1}\Sigma_j^{-1}\boldsymbol{I}_s}{\left(\sum_{i=1}^{N}\gamma_i\right)^2}\right], \tag{3.49}$$

Rewrite Eq.(3.49) into integral form, Eq.(3.9) can be derived.

Appendix B

For GMM, $q_\alpha(\boldsymbol{I}_s)$is the result of convolving $p(\boldsymbol{I}_s)$ and the distribution of message $\mathbf{z}$. It is easy to find that $q_\alpha(\boldsymbol{I}_s)$ has the expression:

$$q_\alpha(\boldsymbol{I}_s) = \sum_{i=1}^{N} a_i \mathcal{N}(\boldsymbol{I}_s; \mathbf{0}, \Sigma_i + \alpha^2 \mathbf{I}), \tag{3.50}$$

With Eq.(3.50) and the distribution of $p(\boldsymbol{I}_s)$, the difference between $H(q_\alpha(\boldsymbol{I}_s))$ and $H(p(\boldsymbol{I}_s), q_\alpha(\boldsymbol{I}_s))$ can be written as:

$$\begin{aligned} &H(p(\boldsymbol{I}_s), q_\alpha(\boldsymbol{I}_s)) - H(q_\alpha(\boldsymbol{I}_s)) \\ &\quad = \int_{\boldsymbol{I}_s} \left(\sum_{i=1}^{N} \gamma_i - \gamma_i(\alpha) \right) ln \left(\sum_{i=1}^{N} \gamma_i(\alpha) \right) d\boldsymbol{I}_s, \end{aligned} \tag{3.51}$$

where $\gamma_i = a_i \mathcal{N}(\boldsymbol{I}_s; \mathbf{0}, \Sigma_i)$, while $\gamma_i(\alpha)$ is defined as:

$$\gamma_i(\alpha) = a_i \mathcal{N}(\boldsymbol{I}_s; \mathbf{0}, \Sigma_i + \alpha^2 \mathbf{I}), \tag{3.52}$$

when α is small, $\gamma_i - \gamma_i(\alpha)$ can be expanded as:

$$\begin{aligned} \gamma_i - \gamma_i(\alpha) &= a_i \mathcal{N}(\boldsymbol{I}_s; \mathbf{0}, \Sigma_i) - a_i \mathcal{N}(\boldsymbol{I}_s; \mathbf{0}, \Sigma_i + \alpha^2 \mathbf{I}) \\ &= \frac{-a_i}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} \left(\eta_i(\alpha) - \eta_i \right) + o(\alpha^2), \end{aligned} \tag{3.53}$$

where η_i and $\eta_i(\alpha)$ are:

$$\eta_i = exp\left(-\frac{\boldsymbol{I}_s^T \Sigma_i^{-1} \boldsymbol{I}_s}{2} \right), \tag{3.54}$$

$$\eta(\alpha)_i = exp\left(-\frac{\boldsymbol{I}_s^T \left(\Sigma_i + \alpha^2 \mathbf{I} \right)^{-1} \boldsymbol{I}_s}{2} \right), \tag{3.55}$$

For $\eta(\alpha)_i$, we use the Taylor expansion:

$$\begin{aligned}
\eta(\alpha)_i &= exp\left(-\frac{\boldsymbol{I}_s^T\left(\Sigma_i+\alpha^2\mathbf{I}\right)^{-1}\boldsymbol{I}_s}{2}\right) \\
&= exp\left(-\frac{\boldsymbol{I}_s^T\left(\Sigma_i^{-1}-\alpha^2\Sigma_i^{-2}\right)\boldsymbol{I}_s}{2}\right)+o(\alpha^2) \\
&= exp\left(-\frac{\boldsymbol{I}_s^T\Sigma_i^{-1}\boldsymbol{I}_s}{2}\right)exp\left(\frac{\alpha^2\boldsymbol{I}_s^T\Sigma_i^{-2}\boldsymbol{I}_s}{2}\right)+o(\alpha^2) \\
&= exp\left(-\frac{\boldsymbol{I}_s^T\Sigma_i^{-1}\boldsymbol{I}_s}{2}\right)\left(1+\frac{\alpha^2}{2}\boldsymbol{I}_s^T\Sigma_i^{-2}\boldsymbol{I}_s\right)+o(\alpha^2) \\
&= \eta_i\left(1+\frac{\alpha^2}{2}\boldsymbol{I}_s^T\Sigma_i^{-2}\boldsymbol{I}_s\right)+o(\alpha^2),
\end{aligned} \tag{3.56}$$

In this derivation, two approximations are used:

$$\left(\boldsymbol{A}+\alpha^2\boldsymbol{B}\right)^{-1}=\boldsymbol{A}^{-1}-\alpha^2\boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{A}^{-1}+o(\alpha^2), \tag{3.57}$$

$$e^x=1+x+o(x), \tag{3.58}$$

where $\boldsymbol{A}$ and $\boldsymbol{B}$ are two invertible matrices, x is a small value. Combining Eq.(5.51), Eq.(5.53) and Eq.(5.56), we have:

$$\begin{aligned}
&H(p(\boldsymbol{I}_s),q_\alpha(\boldsymbol{I}_s))-H(q_\alpha(\boldsymbol{I}_s)) \\
&=-\frac{\alpha^2}{2}\int_{\boldsymbol{I}_s}\left(\sum_{i=1}^{N}\gamma_i\boldsymbol{I}_s^T\Sigma_i^{-2}\boldsymbol{I}_s\right)ln\left(\sum_{i=1}^{N}\gamma_i(\alpha)\right)d\boldsymbol{I}_s+o(\alpha_2) \\
&=-\frac{\alpha^2}{2}\int_{\boldsymbol{I}_s}\left(\sum_{i=1}^{N}\gamma_i\boldsymbol{I}_s^T\Sigma_i^{-2}\boldsymbol{I}_s\right)ln\left(\sum_{i=1}^{N}\gamma_i\right)d\boldsymbol{I}_s+o(\alpha^2),
\end{aligned} \tag{3.59}$$

Thus the expression of ε is:

$$\varepsilon=-\frac{1}{2}\int_{\boldsymbol{I}_s}\left(\sum_{i=1}^{N}\gamma_i\boldsymbol{I}_s^T\Sigma_i^{-2}\boldsymbol{I}_s\right)ln\left(\sum_{i=1}^{N}\gamma_i\right)d\boldsymbol{I}_s, \tag{3.60}$$

Appendix C

For the difference between $H(q_\alpha(\boldsymbol{I}_s))$ and $H(p(\boldsymbol{I}_s))$, we consider the following limit H':

$$H' = \lim_{\alpha \to 0} \frac{H(q_\alpha(\boldsymbol{I}_s)) - H(p(\boldsymbol{I}_s))}{\alpha^2}, \tag{3.61}$$

According to our assumption, the message vector has a Gaussian distribution and the stego image is the result of adding Gaussian distributed message onto the cover image. Based on the De Bruijn's Identity [99] in information theory, we can conclude that:

$$\lim_{\alpha \to 0} \frac{H(q_\alpha(\boldsymbol{I}_s)) - H(p(\boldsymbol{I}_s))}{\alpha^2} = tr\left(J(p(\boldsymbol{I}_s))\right) \equiv \mathcal{M}, \tag{3.62}$$

Thus,

$$H' = \lim_{\alpha \to 0} \frac{H(q_\alpha(\boldsymbol{I}_s)) - H(p(\boldsymbol{I}_s))}{\alpha^2} = \mathcal{M}, \tag{3.63}$$

With the Taylor theorem, the difference between $H(q_\alpha(\boldsymbol{I}_s))$ and $H(p(\boldsymbol{I}_s))$ can be expanded as:

$$H(q_\alpha(\boldsymbol{I}_s)) - H(p(\boldsymbol{I}_s)) = \mathcal{M}\alpha^2 + o(\alpha^2), \tag{3.64}$$

Appendix D

Before proving the conclusion, we assume that the probability distribution of the cover image $p(\boldsymbol{I}_s)$ is continuous and bounded. Then, we have:

$$0 < \max_{\boldsymbol{I}_s, i} \gamma_i \leq R, \tag{3.65}$$

where R is a finite positive value. Since $p(\boldsymbol{I}_s)$ is bounded, the eigenvalue of any Σ_i^{-1} is bounded. Further, the pixel value of natural images cannot be infinitely large, thus the eigenvalue of any Σ_i^{-1} cannot approach to infinity. Based these observations, we conclude that:

$$0 < \sigma_{min} < \sigma_{max} \leq \nu \sigma_{min}, \tag{3.66}$$

where σ_{min} and σ_{max} denotes the maximum and minimum eigenvalue among all covariance matrices $\{\Sigma_i^{-1}\}_{i=1}^N$ respectively, ν is a finite constant larger than 1.

For ε, it can be rewritten as the following equation:

$$\begin{aligned} 2\varepsilon &= -\int_{\boldsymbol{I}_s} \left(\sum_{i=1}^{N} \gamma_i \boldsymbol{I}_s^T \Sigma_i^{-2} \boldsymbol{I}_s \right) ln \left(\sum_{i=1}^{N} \gamma_i \right) d\boldsymbol{I}_s \\ &= -\int_{\boldsymbol{I}_s} \frac{\left(\sum_{i=1}^{N} \gamma_i \boldsymbol{I}_s^T \Sigma_i^{-2} \boldsymbol{I}_s \right) \left(\sum_{j=1}^{N} \gamma_j \right) ln \left(\sum_{i=1}^{N} \gamma_i \right)}{\left(\sum_{j=1}^{N} \gamma_j \right)} d\boldsymbol{I}_s, \end{aligned} \tag{3.67}$$

For any x, the inequality $x \leq |x|$ always holds, then:

$$\begin{aligned} &2\varepsilon \leq |2\varepsilon| \\ &= \int_{\boldsymbol{I}_s} \frac{\left(\sum_{i=1}^{N} \gamma_i \boldsymbol{I}_s^T \Sigma_i^{-2} \boldsymbol{I}_s \right) \left(\sum_{j=1}^{N} \gamma_j \right) ln \left(\sum_{i=1}^{N} \gamma_i \right)}{\left(\sum_{j=1}^{N} \gamma_j \right)} d\boldsymbol{I}_s, \end{aligned} \tag{3.68}$$

Since $xlnx \leq x^2$ for any $x > 0$, we have:

$$2\varepsilon \leq \int_{\boldsymbol{I}_s} \frac{\left(\sum_{i=1}^{N} \gamma_i \boldsymbol{I}_s^T \Sigma_i^{-2} \boldsymbol{I}_s\right) \left(\sum_{j=1}^{N} \gamma_j\right)^2}{\left(\sum_{j=1}^{N} \gamma_j\right)} d\boldsymbol{I}_s, \tag{3.69}$$

With the Holder's inequality:

$$\begin{aligned} 2\varepsilon \leq & \int_{\boldsymbol{I}_s} \frac{\left(\sum_{i=1}^{N} \gamma_i \boldsymbol{I}_s^T \Sigma_i^{-2} \boldsymbol{I}_s\right) \left(\sum_{j=1}^{N} \gamma_j\right)}{\left(\sum_{j=1}^{N} \gamma_j\right)} d\boldsymbol{I}_s \\ & \cdot \max \left(\sum_{i=1}^{N} \gamma_i\right) \\ & \leq NR \int_{\boldsymbol{I}_s} \frac{\left(\sum_{i=1}^{N} \gamma_i \boldsymbol{I}_s^T \Sigma_i^{-2} \boldsymbol{I}_s\right) \left(\sum_{j=1}^{N} \gamma_j\right)}{\left(\sum_{j=1}^{N} \gamma_j\right)} d\boldsymbol{I}_s \\ & = NR \int_{\boldsymbol{I}_s} \frac{\left(\sum_{i=1}^{N} \sum_{j=1}^{N} \gamma_i \gamma_j \boldsymbol{I}_s^T \Sigma_i^{-2} \boldsymbol{I}_s\right)}{\left(\sum_{j=1}^{N} \gamma_j\right)} d\boldsymbol{I}_s, \end{aligned} \tag{3.70}$$

For the proposed measure $\mathcal{M}$, its expression is:

$$\mathcal{M} = \int_{\boldsymbol{I}_s} \frac{\left(\sum_{i=1}^{N} \sum_{j=1}^{N} \gamma_i \gamma_j \boldsymbol{I}_s^T \Sigma_i^{-1} \Sigma_j^{-1} \boldsymbol{I}_s\right)}{\left(\sum_{j=1}^{N} \gamma_j\right)} d\boldsymbol{I}_s, \tag{3.71}$$

If we choose a constant K_s as:

$$K_s \geq \frac{\sigma_{max}}{\sigma_{min}} \geq 1, \tag{3.72}$$

We consider the following difference:

$$D = K_s \sum_{i,j=1}^{N} \gamma_i \gamma_j \boldsymbol{I}_s^T \Sigma_i^{-1} \Sigma_j^{-1} \boldsymbol{I}_s - \sum_{i,j=1}^{N} \gamma_i \gamma_j \boldsymbol{I}_s^T \Sigma_i^{-2} \boldsymbol{I}_s, \tag{3.73}$$

Eq.(3.73) contains two terms T_1 and T_2:

$$T_1 = (K_s - 1) \sum_{i=1}^{N} \gamma_i^2 \boldsymbol{I}_s^T \Sigma_i^{-2} \boldsymbol{I}_s, \tag{3.74}$$

$$T_2 = \sum_{i=1}^{N} \sum_{j \neq i}^{N} \gamma_i \gamma_j \boldsymbol{I}_s^T \Sigma_i^{-1} \left(K_s \Sigma_j^{-1} - \Sigma_i^{-1} \right) \boldsymbol{I}_s, \tag{3.75}$$

Since $K_s > 1$ and Σ_i^{-1} is symmetric positive definite matrix, then for any nonzero vector $\boldsymbol{I}_s$:

$$T_1 = (K_s - 1) \sum_{i=1}^{N} \gamma_i^2 \boldsymbol{I}_s^T \Sigma_i^{-2} \boldsymbol{I}_s > 0, \tag{3.76}$$

For T_2, it is easy to verify that, for any nonzero vector $\boldsymbol{I}_s$:

$$\gamma_i \gamma_j \boldsymbol{I}_s^T \Sigma_i^{-1} \left(K_s \Sigma_j^{-1} - \Sigma_i^{-1} \right) \boldsymbol{I}_s > 0, \tag{3.77}$$

This conclusion is true because for any nonzero vector $\boldsymbol{I}_s$, the following inequality holds:

$$\frac{K_s \boldsymbol{I}_s^T \Sigma_j^{-1} \boldsymbol{I}_s}{\boldsymbol{I}_s^T \Sigma_i^{-1} \boldsymbol{I}_s} \geq \frac{K_s \sigma_{min}^j ||U_j \boldsymbol{I}_s||^2}{\sigma_{max}^i ||U_i \boldsymbol{I}_s||^2} = \frac{K_s \sigma_{min}^j}{\sigma_{max}^i} > 1, \tag{3.78}$$

where σ_{min}^j and σ_{max}^i denotes the minimum and the maximum eigenvalue of Σ_j^{-1} and Σ_i^{-1} respectively, U_i and U_j are the orthogonal matrices whose columns are eigenvectors of Σ_j^{-1} and Σ_i^{-1}. Eq.(3.78) indicates that $K_s \Sigma_j^{-1} - \Sigma_i^{-1}$ is a positive definite matrix. Since Σ_i^{-1} is positive definite, γ_i and γ_j are positive values, then it is easy to prove that, for any nonzero vector $\boldsymbol{I}_s$:

$$T_2 = \sum_{i=1}^{N} \sum_{j \neq i}^{N} \gamma_i \gamma_j \boldsymbol{I}_s^T \Sigma_i^{-1} \left(K_s \Sigma_j^{-1} - \Sigma_i^{-1} \right) \boldsymbol{I}_s > 0, \tag{3.79}$$

Thus D is a positive value. Then we conclude that:

$$2\varepsilon \leq (K_s\mathcal{M}) \cdot NR = \widetilde{K}_s \cdot \mathcal{M}, \tag{3.80}$$

where $\widetilde{K}_s = K_s NR$. The conclusion is proved.

Appendix E

There is an analytical relationship between the cumulative distribution function $\Phi(x)$ of normal distribution and the error function $erf(x)$:

$$\Phi(x) = \frac{1}{2}\left[1 + erf\left(\frac{x}{\sqrt{2}}\right)\right], \tag{3.81}$$

For $erf(x)$, it can be expanded as an infinite Burmann series [100]:

$$erf(x) = \frac{2}{\sqrt{\pi}}\left(sgn(x)\sqrt{1-e^{-x^2}}\right)\sum_{k=0}^{\infty} c_k e^{-kx^2}, \tag{3.82}$$

where c_k are real coefficients and $c_0 = \sqrt{\pi}/2$, $sgn(x)$ denotes the sign function. When x is very large, the series can be approximated as:

$$erf(x) \approx \frac{2}{\sqrt{\pi}}\left(sgn(x)\cdot 1\right)c_0 = sgn(x), \tag{3.83}$$

Thus, $\Phi(x)$ can be approximated:

$$\Phi(x) \approx \frac{1}{2}\left[1 + sgn\left(\frac{x}{\sqrt{2}}\right)\right] = \frac{1}{2}\left[1 + sgn(x)\right], \tag{3.84}$$

For the Normal-Laplace distribution $q_c(I_c)$, its expression is:

$$\begin{aligned} q_c(I_c) = \frac{1}{2\sigma} exp\left(\frac{\alpha^2}{2\sigma^2}\right)\Bigg[e^{I_c/\sigma}\Phi\left(-\frac{\sigma I_c + \alpha^2}{\sigma\alpha}\right) \\ + e^{-I_c/\sigma}\Phi\left(\frac{\sigma I_c - \alpha^2}{\sigma\alpha}\right)\Bigg], \end{aligned} \tag{3.85}$$

when α is small compared to I_c, then:

$$\Phi\left(-\frac{\sigma I_c + \alpha^2}{\sigma\alpha}\right) \approx \Phi\left(-\frac{I_c}{\alpha}\right) \approx \frac{1}{2}\left[1 + sgn(-I_c)\right], \tag{3.86}$$

$$\Phi\left(\frac{\sigma I_c - \alpha^2}{\sigma\alpha}\right) \approx \Phi\left(\frac{I_c}{\alpha}\right) \approx \frac{1}{2}\left[1 + sgn(I_c)\right], \tag{3.87}$$

Combining Eq.(3.85), Eq.(3.86) and Eq.(3.87), the conclusion is proved:

$$\begin{aligned} q_c(I_c) &\approx \frac{1}{2\sigma} exp\left(\frac{\alpha^2}{2\sigma^2}\right) exp\left(-\frac{|I_c|}{\sigma}\right) \\ &= \frac{1}{2\sigma} exp\left(\frac{\alpha^2}{2\sigma^2} - \frac{|I_c|}{\sigma}\right), \end{aligned} \tag{3.88}$$

Chapter 4

Modeling Natural Images with Convolutional Neural Network for Steganalysis

4.1 Overview

Natural images show strong correlations in adjacent pixels. As analyzed in section 2.4.2, these local correlations can be effectively modeled by CNNs. Because of this advantage, we propose to use CNN to address image steganalysis in this chapter. In order to reliably detect steganography, we design a novel CNN model for steganalysis from two aspects. For the first, different from existing CNN based steganalytic algorithms that use a predefined highpass kernel to preprocess input images, we integrate the highpass filtering operation into the proposed network by building a content suppression subnetwork. Highpass kernels in this subnetwork are adaptively updated in the network training, allowing more powerful discriminative features come into the subsequent network than that of CNN models with a predefined kernel. For the second,

we propose a novel subnetwork to actively preserve and further strengthen the weak stego signal generated by secret messages based on residual learning, making the whole network capture the difference between cover images and stego images. Theoretically, we prove that the residual learning can preserve the weak stego signal for the deep model with any depths. Extensive experiments demonstrate that, when cover images and stego images are paired in training and testing, the proposed network can detect the state of the art steganography with much performances than previous methods. We further discuss the proposed method in general cases and analyze the limitation of batch normalization for image steganalysis.

4.2 Background and Motivation

Designing effective features that are sensitive to message embedding is key to steganalysis. Traditional methods use handcrafted features to detect steganography. However, the feature design is a difficult task which needs the domain knowledge of steganography and steganalysis. Recently, several interesting works have been proposed to detect steganography based on deep convolutional neural network models. Compared with traditional methods that extract handcrafted features, CNN based steganalysis directly learns effective features using various network architectures for discriminating cover images and stego images. Tan and Li [94] first proposed to detect the presence of secret messages based on a deep stacked convolutional auto-encoder network. Qian *et al.* in [117] proposed a model for steganalysis using the standard CNN architecture with Gaussian activation function. Xu *et al.* [34] designed a new CNN structure with *tanh* activation function and absolute operation after the first convolutional layer. Pibre *et al.* [67] presented a novel CNN model featured that the network is greater in height (the number of kernels in each convolutional layer) than in depth and no pooling is involved. Couchot *et al.* [42] proposed a CNN model for steganalysis by using convolutional kernels with very large size.

Preprocessing input images is a crucial step for steganalysis. The purpose of preprocessing is to suppress image content so that the Signal-to-Noise-Ratio (SNR) [1] is largely increased. A limitation of existing CNN based steganalysis is that they only use a predefined highpass kernel to preprocess the input image. This feature would limit the subsequent network to capture effective features to discriminate cover images and stego images, which is harmful for detecting steganography.

Although steganalysis can be formulated as a classification problem, detecting cover images and stego images is actually different from the classical binary classification. Steganalysis is special in the fact that it is to classify cover images and those stego images that are the results of adding weak stego signal into the cover images. This characteristic requires that CNN models for steganalysis should preserve the weak signals when input stego images propagate the network. Qian *et al.* [52] utilized the Gaussian non-linearity rather than the Sigmoid function to better preserve the discriminability between cover images and stego images. Xu *et al.* [34] used the absolute value of feature maps in early layers to improve modeling the difference of cover images and stego images in the subsequent network for steganalysis. To avoid the discriminative information lost in a very deep network, Pibre [67] and Couchot [42] proposed novel CNN architectures with small number of layers for steganalysis. These techniques can partially handle the difficulty of steganalysis in a CNN model, but how to preserve and even strengthen the discriminability of cover images and stego images, especially for a deep network, is a problem waiting to be solved.

To address these difficulties, this paper proposes a unified CNN for steganalysis. On one hand, unlike previous methods that separately preprocess the input image and extract features for classification, we integrate the highpass filtering operation into the proposed network by building a content suppression subnetwork. The highpass kernels in the subnetwork is adaptively updated in the network training, allowing more powerful discriminative features come into the subsequent network than that

[1] Here, "signal" is the weak stego signal generated by message embedding, "noise" denotes the content of a cover image.

of CNN models with a predefined kernel. To the best of our knowledge, this is the first CNN model that unifies image preprocessing and feature learning in a whole network for steganalysis. On the other hand, we propose a novel learning scheme to actively preserve the weak stego signal generated by secret message by incorporating residual learning [62] in our network. This learning scheme has demonstrated superior performance than previous CNN based steganalytic methods. In theory, we have proved that shortcut connections in residual learning can effectively preserve the weak stego signal for the network with any depths.

The rest of this paper is organized as follows. In section 4.3, we introduce the content suppression for image steganalysis. In section 4.4, we describe the advantage of CNN model and explain the rationality of residual learning for image steganalysis. In section 4.5, we introduce the proposed network model. In section 4.6, we validate the effectiveness of the proposed model on several states of the art steganographic algorithms. In section 4.7, we discuss several characteristics of the proposed network. The paper is finally closed with the conclusion in section 4.8.

4.3 Adaptive Content Suppression

Steganographic algorithms embed secret messages into cover images by modifying their pixels slightly, i.e. they change each pixel by ± 1. Under this case, it is hard to find a statistical model to capture the difference between cover images and stego images because the stego signal generated by secret messages is too weak. To address this difficulty, instead of modeling natural images directly, modern methods turn to extract the noise component of images by filtering original images with various highpass kernels [49].

Traditional steganalytic algorithms suppress image content using predefined high-pass kernels before feature extraction. Lyu and Farid [92] proposed to get the noise component by decomposing images with wavelet-like kernels. Pevny *et al.* [101] ex-

tracted the SPAM feature along four axes by using pixel difference kernels. To better capture various dependencies among pixels, Fridrich *et al.* [46] proposed the rich model steganalysis which unites a large number of predefined submodels. For the CNN based steganalysis, the KV kernel [34, 94, 117] is used to preprocess input images.

However, these predefined kernels may not be optimal for content suppression. To demonstrate this case, we calculate the following distance:

$$d = ||\mathbf{k} * I_c - \mathbf{k} * I_s||^2/(R \cdot C) \tag{4.1}$$

where $\mathbf{k}$ is the kernel for content suppression, I_c and I_s represent the cover image and the stego image respectively. R and C denote the row and the column of the image. Tab.4.1 gives average d calculated by the predefined KV kernel and the adaptively learned kernel of our model, based on 10,000 BOSSbase images [76] and S-UNIWARD steganography [107] at 0.4 bit-per-pixel (bpp) payload. We can find that the adaptively learned kernel has a large d than that of the predefined kernel. This indicates that the proposed content suppression subnetwork can extract better discriminative information than a predefined kernel for image steganalysis. In section 4.6, we will use experiments to demonstrate that adaptive content suppression can obviously improve the detection accuracy of the proposed model for detecting steganography.

Table 4.1: Average d with different kernels.

Kernel type	Predefined	Adaptive
d	0.3048	0.3074

4.4 Convolutional Neural Network for Steganalysis

In this section, we introduce the convolutional neural network for image steganalysis. Firstly, we explain why CNN models are suitable for image steganalysis. Then, we point out the difficulty of training a deep CNN model for discriminating cover images

and stego images. Finally, we verify that residual learning can effectively overcome this difficulty.

4.4.1 Advantages of Using CNN for Image Steganalysis

In recent years, CNN has achieved a great success for many image related tasks. A series of breakthroughs have been made for discriminative learning, including image classification [6, 62, 63], image denoising [108], and image super-resolution [16]. In addition, several recent works show that CNN models are successfully applied for generative learning, including real image generation [38], image rendering [64] and texture synthesis [31]. These successes indicate that CNN can not only extract effective features for discriminating different images but also provide a good description for representing real images. All the evidences show that CNN can well describe the distribution of natural images. These results motivate us to use CNN for image steganalysis, since its purpose is to discriminate the "natural" images (cover images) against the "unnatural" images contaminated by embedded secret messages (stego images). The following three characteristics of CNN models further demonstrate that they are suitable for the task of image steganalysis:

- Convolutional kernels in CNN models can exploit the strong spatially local correlation present in input images. This local correlation among image pixels is distorted when secret messages are embedded, making it different from the correlation in natural images. The difference between natural images and distorted images can be effectively captured by CNN models;

- The convolution operation is actually to sum image pixels in a local region, which would accumulate the weak stego signal of this region to be a large value. This may lead to stego images be more easily detected against cover images;

- Nonlinear mappings in CNN models make them able to extract rich features for

classifying cover images and stego images. These features, which are automatically learned by updating the network, can hardly be designed by hand.

Although CNN models are suitable for image steganalysis, in the following part, we will show that training a deep CNN model to classify cover images and stego images is difficult.

4.4.2 Difficulty of Training a Deep CNN for Image Steganalysis

Training deep convolutional neural networks is difficult. A deep CNN model may suffer the *feature diminishing* problem [31] when the input data propagates the network forwardly. It may also suffer the *gradient vanishing* problem when the error signal is back-propagated [114]. For image steganalysis, the *feature diminishing* becomes a major problem. This indicates that the weak stego signal added to a cover image would be attenuated as it travels the whole CNN model, making the later network hardly capture effective features to discriminate cover images and stego images. To illustrate this phenomenon, we perform a mathematical derivation as follows. Assume we have a cover image $\mathbf{x}$ and its stego version $\mathbf{y}$, where the stego image $\mathbf{y}$ can be represented as:

$$\mathbf{y} = \mathbf{x} + \mathbf{s} \tag{4.2}$$

where $\mathbf{s}$ denotes the stego signal generated by message embedding. Generally, CNN can be abstracted as a typical model as Fig.4.1(a). In this abstracted form, the mapping $H_i(\mathbf{x})$ $(i = 1, 2, \cdots, n)$ could be a convolutional layer, a nonlinear activation layer, a pooling layer or their combinations. By feeding $\mathbf{x}$ and $\mathbf{y}$ into a typical CNN model, we obtain their outputs:

$$Z_t^n(\mathbf{x}) = H_n(\cdots H_2(H_1(\mathbf{x}))) \tag{4.3}$$

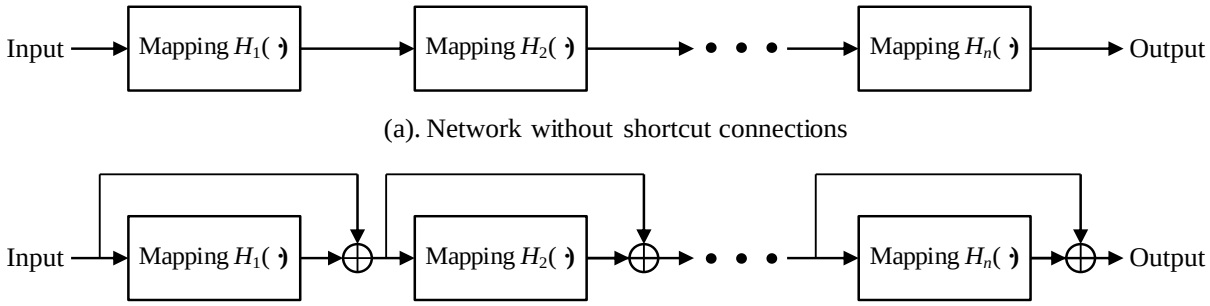


Figure 4.1: Network without and with shortcut connections. (a). A typical CNN model can be abstracted as a network with cascaded building blocks. (b). A residual learning network can be abstracted as a network with cascaded building blocks, where each building block has a shortcut connecting its input and output.

$$Z_t^n(\mathbf{y}) = H_n(\cdots H_2(H_1(\mathbf{y}))) \tag{4.4}$$

Compared with the cover image $\mathbf{x}$, the stego signal $\mathbf{s}$ is often very weak. Therefore, we iteratively use Taylor expansion for $Z_t^n(\mathbf{y})$ and get the following result:

$$Z_t^n(\mathbf{y}) = Z_t^n(\mathbf{x}) + \left(\prod_{i=1}^{n} F^{(i)}(\mathbf{x})\right)\mathbf{s} + O(||\mathbf{s}||^2) \tag{4.5}$$

where $O(||\mathbf{s}||^2)$ is the expansion remainder, $F^{(i)}(\mathbf{x})$ is:

$$F^{(i)}(\mathbf{x}) = \begin{cases} H_i'(\mathbf{x}), & i = 1 \\ H_i'(\cdots H_1(\mathbf{x})), & i < 1 \leq n \end{cases} \tag{4.6}$$

In above equation, $H_i'(\mathbf{x})$ denotes the derivative of the mapping $H_i(\mathbf{x})$:

$$H_i'(\mathbf{x}) = \frac{\partial H_i(\mathbf{x})}{\partial \mathbf{x}} \tag{4.7}$$

In a CNN model, each element f_i in the derivative matrix $F^{(i)}(\mathbf{x})$ satisfies the following inequality:

$$|f_i| \leq 1 \tag{4.8}$$

where p, q represents the row index and the column index of $F^{(i)}(\mathbf{x})$ respectively. We explain this result for each basic operation in a CNN model:

- For a convolutional layer, f_i actually relates to the sum of image pixels multiplied by weights in a convolutional kernel. To ensure the stability of learning CNN models, existing methods initialize convolutional kernels with small weights (e.g. Gaussian random values with zero mean and 0.01 standard derivation) and set learning rate for parameter updating to a small value. These settings ensure that elements in convolutional layers are small during the learning phase. In addition, the size of a convolutional kernel in CNN models is often small. Consequently, f_i of a convolutional layer is small and Eq.(8) can be satisfied in most cases;

- For a nonlinear activation layer, f_i in $F^{(i)}(\mathbf{x})$ is ensured to be smaller than 1. This is because for existing activation functions, e.g. the sigmoid, tanh or ReLU, their slopes are smaller than 1 anywhere.

- For a pooling layer, either the average pooling or the maximum pooling does not increase the absolute value of each element in a feature map $H_i(\mathbf{x})$, thus Eq.(4.8) is satisfied.

With the property as the Eq.(4.8), $\prod_{i=1}^{n} F^{(i)}(\mathbf{x})$ decays exponentially as n increases. This will make the difference between $\mathbf{x}$ and $\mathbf{y}$ very small for large n. Under this case, the CNN model can hardly discriminate cover images and stego images. The result explains why CNN models that can better detect steganography have very small depths [34, 67]. However, deep neural networks have more powerful representation ability than the shallow ones [69].

An exception of Eq.(4.8) is the batch normalization [91] in CNN models. Recently, this operation becomes an effective technique in CNN models to improve their convergence speed. For a batch normalization layer, $F^{(i)}(\mathbf{x})$ is a diagonal matrix in which each diagonal element is equal to the inverse of the variance of input data in the corresponding dimension. Therefore, $|f_i|$ could be larger than 1 if the variance of the data in

a mini-batch is small. Nevertheless, for a CNN model with batch normalization layers, it can not pledge that $\prod_{i=1}^{n} F^{(i)}(\mathbf{x})$ does not decay in deep layers since the value $|f_i|$ is determined by the variance of input data.

4.4.3 Rationality of Residual Learning for image Steganalysis

Residual learning was originally proposed by He *et al.* in [62]. The main feature of residual learning is that a shortcut path connects the input and the output of $H_i(\mathbf{x})$ in a CNN model. According to the introduction in [62], this shortcut connection can enforce a CNN model to fit the residual part of a function to be approximated, making the network be optimized easily. For image steganalysis, a network with shortcut connections (the model as Fig.4.1(b) shows) can effectively overcome the feature diminishing phenomenon. Same to the analysis as Fig.4.1(a), we feed the cover image $\mathbf{x}$ and its stego image $\mathbf{y}$ into the network as Fig.4.1(b) and obtain their outputs:

$$Z_s^n(\mathbf{x}) = R_n(\cdots R_2(R_1(\mathbf{x}))) \tag{4.9}$$

$$Z_s^n(\mathbf{y}) = R_n(\cdots R_2(R_1(\mathbf{y}))) \tag{4.10}$$

where $R_i(\mathbf{x})$ denotes:

$$R_i(\mathbf{x}) = H_i(\mathbf{x}) + \mathbf{x},\ 1 \leq i \leq n \tag{4.11}$$

Similarly, we perform the Taylor expansion for Eq.(9):

$$Z_s^n(\mathbf{y}) = Z_s^n(\mathbf{x}) + \left[\prod_{i=1}^{n}\left[1 + H_i'(F_R^{(i)}(\mathbf{x}))\right]\right]\mathbf{s} + O(||\mathbf{s}||^2) \tag{4.12}$$

where $F_R^{(i)}(\mathbf{x})$ is:

$$F_R^{(i)}(\mathbf{x}) = \begin{cases} \mathbf{x}, & i = 1 \\ R_{i-1}(\mathbf{x}) & i = 2 \\ R_{i-1}(\cdots R_1(\mathbf{x})), & i > 2 \end{cases} \tag{4.13}$$

Unlike the case as Fig.4.1(a), the coefficient matrix of the stego signal $\mathbf{s}$, $\prod_{i=1}^{n}\left[1 + H_i'(F_R^{(i)}(\mathbf{x}))\right]$, does not exponentially decays as the depth increases. To better understand the advantage of the network with shortcut connections, we factorize $Z_s^n(\mathbf{x})$ and $Z_s^n(\mathbf{y})$ when n is 2. For $Z_s^n(\mathbf{x})$, we have:

$$Z_s^2(\mathbf{x}) = R_2(R_1(\mathbf{x})) = \mathbf{x} + H_1(\mathbf{x}) + H_2(H_1(\mathbf{x}) + \mathbf{x}) \tag{4.14}$$

For $Z_s^n(\mathbf{y})$, we iteratively use Taylor expansion and obtain:

$$\begin{aligned} Z_s^2(\mathbf{y}) &= R_2(R_1(\mathbf{y})) \\ &= R_2(R_1(\mathbf{x})) + [R_2'(R_1(\mathbf{x})) \cdot R_1'(\mathbf{x})]\,\mathbf{s} + O(||\mathbf{s}||^2) \\ &= \mathbf{x} + H_1(\mathbf{x}) + H_2(H_1(\mathbf{x}) + \mathbf{x}) + \mathbf{s} + [H_1'(\mathbf{x})]\,\mathbf{s} \\ &+ [H_2'(H_1(\mathbf{x}) + \mathbf{x})]\,\mathbf{s} + [H_2'(H_1(\mathbf{x}) + \mathbf{x}) \cdot H_1'(\mathbf{x})]\,\mathbf{s} \\ &+ O(||\mathbf{s}||^2) \end{aligned} \tag{4.15}$$

We compared the factorization Eq.(4.15) with Eq.(4.5), and find two advantages of the network with shortcut connections:

- The coefficient matrix for the stego signal $\mathbf{s}$ does not decay as the network's depth increases. Unlike the Eq.(4.5), the coefficient matrices for $\mathbf{s}$ in Eq.(4.15),

i.e. $\mathbf{s}$, $[H_1'(\mathbf{x})]\,\mathbf{s}$, $[H_2'(H_1(\mathbf{x})+\mathbf{x})]\,\mathbf{s}$, are always kept to be no larger than order one, independent of network's depth. Actually, this is a general case for any depth. This property ensures that the stego signal $\mathbf{s}$ does not decays as the network's depth increases;

- The difference between cover images and stego images does not decay as the network's depth increases. By checking Eq.(4.15), we find that each term of $\mathbf{x}$ is accompanied with a corresponding term of $\mathbf{s}$, i.e. $\mathbf{s}$ for $\mathbf{x}$, $[H_1'(\mathbf{x})]\,\mathbf{s}$ for $H_1(\mathbf{x})$, and $[H_2'(H_1(\mathbf{x})+\mathbf{x})]\,\mathbf{s}$ for $H_2(H_1(\mathbf{x})+\mathbf{x})$. This property ensures a non-decaying SNR between the stego signal $\mathbf{s}$ and the image content $\mathbf{x}$ when the depth n is large, which is beneficial for discriminating cover images and stego images.

To summarize, we have explained the rationality of residual learning for image steganalysis. We will introduce the proposed model in the following section.

4.5 Proposed Network Model

4.5.1 Network Architecture

Fig.4.2 shows the overall architecture of the proposed network model. The network contains the content suppression sub-network, the residual learning sub-network and the classification sub-network. Each sub-network has its own role in information processing of the overall model, which are introduced in the following parts.

The content suppression sub-network is to extract the noise component of input cover/stego images. Three 5×5 kernels rather than one are used to filter the input image, aiming to capture more dependencies among pixels. To pledge that the sub-network indeed extracts noise components, each of three kernels is initialized by a

highpass kernel, i.e. the KV kernel:

$$KV = \frac{1}{12}\begin{pmatrix} -1 & 2 & -2 & 2 & -1 \\ 2 & -6 & 8 & -6 & 2 \\ -2 & 8 & -12 & 8 & -2 \\ 2 & -6 & 8 & -6 & 2 \\ -1 & 2 & -2 & 2 & -1 \end{pmatrix} \tag{4.16}$$

The residual learning subnetwork is to extract effective features for steganalysis. In the residual learning sub-network, 64 filters with the size of 7×7 are used to convolve the the noise components generated by the content suppression sub-network. Following the convolutional layer is a batch normalization layer, a ReLU activation layer [109] and a max pooling layer[2]. Then, the network uses two kinds of blocks to process the data: the residual learning (ResL) block and the dimension increasing block. A ResL block consists of two convolutional layers, each of which is also followed by a batch normalization layer and a ReLU layer (i.e. two cascaded "Conv+BN+ReLU" blocks). The size of convolutional kernels in the block is 3×3 and the number of kernels is equal to the number of input feature map (details about the block are described in [62]). A shortcut path connects the input and the output of the block, acting as the identical mapping. For a dimension increasing block, the only difference to a ResL block is that the number of feature maps is doubled and each feature map is down-sampled for the output. In general, there are several ResL blocks before a dimensional increasing block in the residual learning sub-network. We use Fig.3 to represent feature maps followed by several ResL blocks to make the figure of overall network compact. For economical considerations [62], a bottleneck version for residual learning and dimension increasing is developed for very deep networks. Different from the non-bottleneck version with two convolutional layers, a bottleneck version has three convolutional layers [62].

In general, there are several ResL blocks before a dimensional increasing block in the

[2]The batch normalization and ReLU are not shown in the figure.

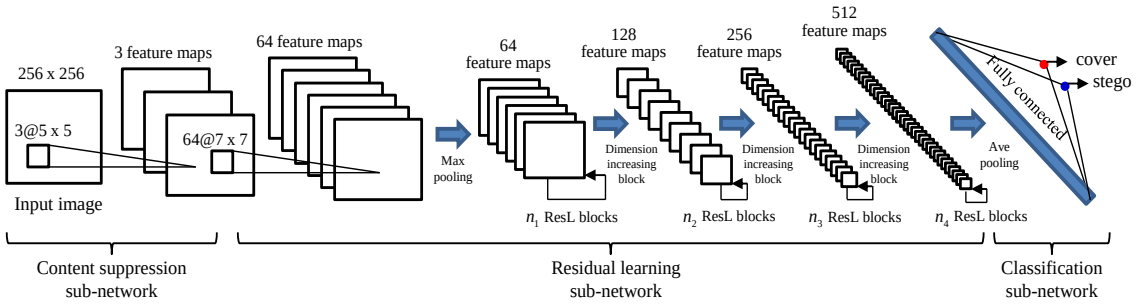


Figure 4.2: The proposed network for image steganalysis. In the content suppression sub-network, three kernels initialized by a KV filter is used to extract the noise component of input images. In the residual learning sub-network, the residual learning (ResL) block and dimension increasing block are used to extract effective features for discriminating cover/stego images. The classification sub-networks maps features into binary labels. $p@q \times q$ denotes p convolutional kernels with the size of $q \times q$.

residual learning stage. We use Fig.4.3 to represent feature maps followed by several ResL blocks to make the figure of overall network compact.

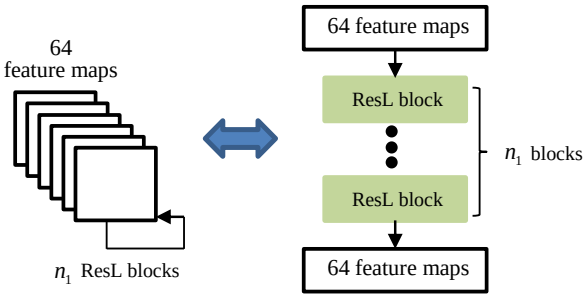


Figure 4.3: Feature maps followed by several ResL blocks.

A classification sub-network maps extracted features into binary labels. Two output nodes, which corresponds to the label of cover and stego, are fully connected to the feature map that are averaged pooled in the residual learning sub-network.

4.5.2 Network Training

Parameters of the proposed network are learned by minimizing the softmax loss function:

$$L(\mathbf{x_i}, \theta) = -\sum_{k=1}^{K} 1\{y_i = k\} \cdot log\left(\frac{e^{o_{i,k}(\mathbf{x_i},\theta)}}{\sum_{k=1}^{K} e^{o_{i,k}(\mathbf{x_i},\theta)}}\right) \tag{4.17}$$

where θ denotes the parameters of the network, including weight matrices $\mathbf{W}$ and the bias vectors $\mathbf{b}$. K is the number of labels, where $K = 2$ in our model. y_i is the label of $\mathbf{x}_i$, $1\{\cdot\}$ is the indicator function. $o_{i,k}(\mathbf{x_i}, \theta)$ represents the output of the network for the sample $\mathbf{x}_i$. $\mathbf{W}$ and $\mathbf{b}$ of the network parameter θ are updated by the mini-batch stochastic gradient descending (SGD):

$$\mathbf{W}(t+1) = \mathbf{W}(t) - \alpha \frac{1}{N} \sum_{i \in B} \frac{\partial L(\mathbf{x}_i, \theta)}{\partial \mathbf{W}} \tag{4.18}$$

$$\mathbf{b}(t+1) = \mathbf{b}(t) - \alpha \frac{1}{N} \sum_{i \in B} \frac{\partial L(\mathbf{x}_i, \theta)}{\partial \mathbf{b}} \tag{4.19}$$

where N is the size of a mini-batch B, α is the learning rate.

4.6 Experiments

This section is to validate the effectiveness of the proposed network model. The dataset used for validation is the BOSSbase 1.01 [76], which is a standard database for evaluating steganography and steganalysis. The original BOSSbase contains 10,000 natural images with the size of 512×512. Following the setting in [34, 67], each image in the dataset is cropped into 4 non-overlapping 256×256 in our experiments. Therefore, we have a cropped BOSSbase with 40,000 images.

In the residual learning sub-network and the classification sub-network, weight matrices $\mathbf{W}$ are initialized by a zero mean Gaussian distribution with the standard derivation 0.01 and biases vectors $\mathbf{b}$ are initialized to zeros. The momentum and the weight decay in two sub-networks are set to 0.9 and 0.0001 respectively. Following the setting in [62], the learning rate α of $\mathbf{W}$ and $\mathbf{b}$ starts from 0.001 and is divided 10 every 50 training epoches. The purpose of dividing the learning rate is to make the network escape the error plateaus. For the content suppression sub-network, all three highpass kernels are initialized as the KV kernel. The learning rate of this subnetwork is set as

follows:

$$\alpha_c(t) = \begin{cases} \alpha_0, \ p > 0.1 \text{ bpp} \\ \alpha_0/t, \ p \leq 0.1 \text{ bpp} \end{cases} \tag{4.20}$$

where α_0 is a predefined value. In our experiment, α_0 is set to 0.0001. p denotes the payload for message embedding, t represents the number of training epoch. We will explain this setting in the discussion section. The size of mini-batch SGD, N, is set to 10, which indicates that 10 paired cover images and their stego versions are for training and testing. In following experiments, we use the batch mean and batch variance as parameters for the batch normalization layer, both in training phase and testing phase. The number of training epoch is set to 200. All experiments are tested on the Nvidia Tesla K80 platform.

4.6.1 Demonstration of Adaptive Content Suppression

In this experiment, we investigate the effect of adaptive content suppression for image steganalysis. The proposed model is compared with and a baseline network that has fixed highpass kernels in the content suppression sub-network. To make the result comparable, highpass kernels in the baseline network are set to the KV filter. Same to the first experiment, $[n_1, n_2, n_3, n_4]$ of both networks are set to $[2, 2, 1, 1]$. We use S-UNIWARD stegnography [107] with the payload 0.4 bit-per-pixel (bpp) for validation. 30,000 randomly selected cover images from the cropped BOSSbase and their stegos are used as the training set. The rest 10,000 images and their stegos are used for testing.

Fig.4.4 shows the training error curve or testing error curve for the rich model steganalysis, the baseline network and the proposed network. In this figure, we find that the proposed network outperforms the baseline network. The performance improvement is indicated in three folds. For the first, both the training error and the testing error of our network are smaller than the baseline network. For the second, our network converges in a faster speed than the baseline network. For the third, the gap

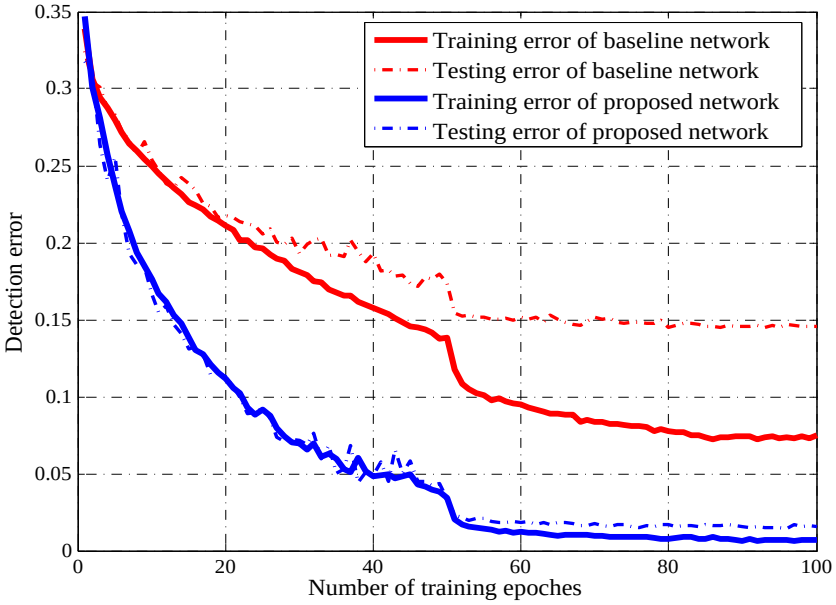


Figure 4.4: Performance comparisons for the rich model method, the proposed network and the baseline network on the S-UNIWARD steganography at 0.4 bpp. This figure only shows the training error and the testing error of the first 100 training epoches. The finally converged detection error rates for the baseline network and the proposed network are 3.16% and 1.47% respectively.

between the training error and the testing error of our network is much smaller than that of the baseline network. All these results demonstrate that an adaptively learned content suppression sub-network can improve the performance obviously.

4.6.2 Performance Comparisons with Prior Arts

We conduct a comprehensive experiment to demonstrate the effectiveness of the proposed network. We compare the proposed network with the classical Spatial Rich Model based steganalysis (SRM) [46] and its select-channel-aware version, the maxSRMd2 steganalysis [95]. SRM based steganalysis first extracts many handcrafted features that are sensitive to message embedding and combine them into a long feature vector for classification. An ensemble classifier [56] is trained based on the extracted features and is used for predicting the label of an input image. The maxSRMd2 steganalysis is similar to the SRM method but pay more attention on image pixels with high embedding probabilities. This steganalytic method is specifically designed for adap-

Table 4.2: Detection error rates of SRM, maxSRMd2 and the proposed network on four steganographic algorithms.

Steganography	Detection algorithm	0.1 bpp	0.2 bpp	0.3 bpp	0.4 bpp	0.5 bpp
WOW	SRM + ensemble	40.15%	31.33%	24.76%	20.08%	15.76%
	maxSRMd2 + ensemble	30.12%	22.84%	17.98%	15.20%	12.93%
	The proposed network	**11.54%**	**3.59%**	**1.42%**	**0.92%**	**0.67%**
S-UNIWARD	SRM + ensemble	40.38%	32.54%	25.51%	20.70%	16.21%
	maxSRMd2 + ensemble	35.63%	28.04%	22.35%	18.84%	15.13%
	The proposed network	**12.61%**	**4.08%**	**2.13%**	**1.47%**	**1.15%**
HILL	SRM + ensemble	43.71%	36.47%	29.39%	23.57%	20.22%
	maxSRMd2 + ensemble	37.26%	30.17%	25.71%	21.63%	17.45%
	The proposed network	**12.94%**	**5.65%**	**3.37%**	**2.17%**	**1.31%**
MiPOD	SRM + ensemble	40.68%	33.25%	26.12%	22.26%	18.38%
	maxSRMd2 + ensemble	37.51%	29.26%	24.19%	20.38%	16.39%
	The proposed network	**11.28%**	**4.41%**	**2.42%**	**1.26%**	**0.62%**

tive steganography. Four states of the art steganographic algorithms, including the Wavelet Obtained Weights steganography (WOW) [104], S-UNIWARD, the HIgh-pass Low-pass Low-pass steganography (HILL) [14] and the Minimizing the Power of Optimal Detector steganography (MiPOD) [110], are used for validation. Same to the setting in previous two experiments, 30,000 randomly selected images and their stegos are for training the model, the rest 10,000 and their stegos are for testing. Tab.4.2 gives the detection error rates of the proposed network. The results demonstrate that our network achieves much lower detection error rates than the rich model based steganalysis over all settings.

4.7 Discussions

4.7.1 Rationality of the Proposed Network When Training Images and Testing Images are Paired

In this section, we will demonstrate that the basic building unit "Conv+BN+ReLU" is suitable for paired image steganalysis. We also provide experimental evidence to verify

the fact that the combination of the unit "Conv+BN+ReLU" and residual learning can effectively overcome feature diminishing phenomenon.

The configuration "Conv+BN+ReLU" is a standard block in residual network. This standard setting is effective for image steganalysis in paired case. To illustrate this claim, we provide mathematical analysis here. Assume we feed the network with a cover image $\mathbf{x}$ and its stego image $\mathbf{y}$, where $\mathbf{y} = \mathbf{x} + \mathbf{s}$. For the block "Conv+BN+ReLU", the outputs of the cover image and the stego image are:

$$\mathbf{x}^{op} = \mathrm{ReLU}\left(\frac{\mathbf{W}\mathbf{x} - \mu}{\sigma}\right) = \max\left(\frac{\mathbf{W}\mathbf{x} - \mu}{\sigma}, 0\right) \tag{4.21}$$

$$\mathbf{y}^{op} = \mathrm{ReLU}\left(\frac{\mathbf{W}\mathbf{y} - \mu}{\sigma}\right) = \max\left(\frac{\mathbf{W}(\mathbf{x}+\mathbf{s}) - \mu}{\sigma}, 0\right) \tag{4.22}$$

where $\mathbf{x}^{op}$ and $\mathbf{y}^{op}$ represent the output of cover image and stego image in paired case respectively. μ denotes the mean value of all pixels in $\mathbf{x}$ and $\mathbf{y}$, σ represents its variance. For simplicity, we have omitted the bias term and the scaling term in the batch normalization layer. For μ, we write it as the follows:

$$\mu = \frac{1}{2}E\left[\mathbf{W}\mathbf{x} + \mathbf{W}\mathbf{y}\right] = E\left[\mathbf{W}\mathbf{x}\right] + \frac{1}{2}E\left[\mathbf{W}\mathbf{s}\right] \tag{4.23}$$

where $E[\cdot]$ denotes the expectation operator. For Eq.(4.21) and Eq.(4.22), we consider the expectation of batch normalization layer's outputs:

$$E\left[\frac{\mathbf{W}\mathbf{x} - \mu}{\sigma}\right] = E\left[\frac{\mathbf{W}\mathbf{x} - E\left[\mathbf{W}\mathbf{x}\right] + \frac{1}{2}E\left[\mathbf{W}\mathbf{s}\right]}{\sigma}\right] = -\frac{E\left[\mathbf{W}\mathbf{s}\right]}{2\sigma} \tag{4.24}$$

$$E\left[\frac{\mathbf{W}\mathbf{y} - \mu}{\sigma}\right] = E\left[\frac{\mathbf{W}\mathbf{x} + \mathbf{W}\mathbf{s} - E\left[\mathbf{W}\mathbf{x}\right] + \frac{1}{2}E\left[\mathbf{W}\mathbf{s}\right]}{\sigma}\right] = \frac{E\left[\mathbf{W}\mathbf{s}\right]}{2\sigma} \tag{4.25}$$

Either the sign of $E\left[\mathbf{W}\mathbf{s}\right]$ is positive or negative, on average, the cover image $\mathbf{x}$ and stego image $\mathbf{y}$ distributed across 0. This property makes elements in the cover feature

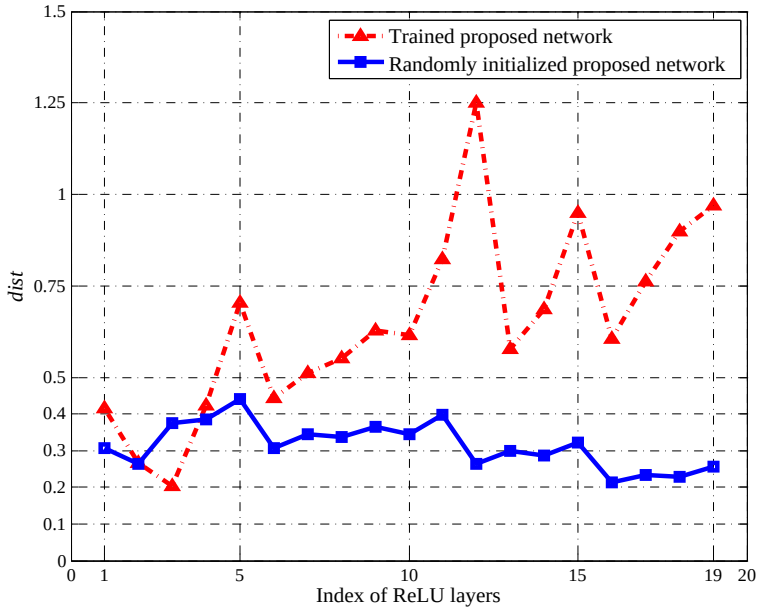


Figure 4.5: Feature map difference between $dist_i$ cover images and stego images at different layers.

map and elements in the stego feature map be easily separated after the ReLU layer. For a mini-batch with several cover images and their stegos, the above analysis is also applicable.

In order to demonstrate that the residual learning combined with the block "Conv+BN+ReLU" can overcome the feature diminishing, we take the following experiment. For each layer in the proposed network, we calculate the Euclidean distance between the feature maps of cover images and stego images:

$$dist_i = \frac{1}{M_i D_i} \sum_j ||f_i(\mathbf{x}_j) - f_i(\mathbf{y}_j)|| \tag{4.26}$$

where M_i and D_i denote the number of feature maps and its dimension at the i-th layer. $f_i(\mathbf{x}_j)$ and $f_i(\mathbf{y}_j)$ represents the feature map of the j-th cover image $\mathbf{x}_j$ and its stego image $\mathbf{y}_j$. Eq.(4.26) actually measures the disciminability between cover images and stego: a large distance indicates that $\mathbf{x}$ and $\mathbf{y}$ can be easily classified while a small distance indicates they are hard to be classified. We calculate $dist_i$ for each layer of a randomly initialized proposed network and a trained proposed network based on 40,000

cover images and their S-UNIWARD stegos. The network with the best configuration, i.e. $[n_1, n_2, n_3, n_4]$ is set to $[2, 2, 1, 1]$ and the total number of convolutional layers is 20, is used for demonstration. Fig.4.5 reports the distance for the output in each ReLU layer in the trained network and the randomly initialized network. The distance is calculated on the cropped BOSSbase dataset and the S-UNIWARD steganography at 0.4 bpp. The figure has shows two interesting results. First, $dist_i$ has almost no changes between the first layer and the last layer for a randomly initialized network. The result is consistent with mathematical analysis in section 4.4. Second, although vibrates $dist_i$ through the network, the trained network enlarges the distance as the network goes deeper. This result indicates that the well trained network can not only overcome the feature diminishing phenomenon but even improve the discriminability between cover images and stego images, when the cover image and the stego image are paired.

4.7.2 Performance Analysis When Testing Images are not Paired

In previous, we have theoretically prove the of rationality of residual learning for steganalysis when cover image $\mathbf{x}$ and its stego $\mathbf{y}$ are paired into the network. However, in our experiment, we find the detection error rate is greatly increased if the test images are not paired. The result is shown as Tab.4.3. In this section, we will provide mathematical analysis and experimental evidence to explain this result.

Table 4.3: Detection error rates for paired case and unpaired case.

Steganography	Paired case	Unpaired case
S-UNIWARD at 0.4 bpp	1.47%	27.61%

Batch Normalization for Natural Image Classification

Batch normalization is a standard technique that is widely used in image classification CNN models. Training a deep neural network model is often difficult not only because

of the gradient vanishing/exploding but also because the distribution of data changes in different layers, which is called the "internal covariate shift" phenomenon. Batch normalization is such a technique that can relieve this phenomenon, by introducing several simple operations to the input data:

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} I_i \tag{4.27}$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m} \left(I_i - \mu_{\mathcal{B}}\right)^2 \tag{4.28}$$

$$\hat{I}_i = \frac{I_i - E\left[I_i\right]}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \tag{4.29}$$

$$I_i^o = \gamma \hat{I}_i + \beta \tag{4.30}$$

where $\mathcal{B} = \{I_{1,\ldots m}\}$ denotes the input data in a mini-batch, $\mu_{\mathcal{B}}$ and $\sigma_{\mathcal{B}}$ represent the mean and variance of the mini-batch $\mathcal{B}$ respectively. ϵ is a small constant to avoid zero dividing, γ and β are the parameters. With these operations, the output data I_i^o in the mini-batch is distributed with fixed mean and variance at any depth after the batch normalization. Thus, deviations to the mean and variance can be eliminated by the batch normalization, which makes the network overcome the "internal covariate shift".

Batch Normalization for Image Steganalysis

For image steganalysis, the batch normalization plays a different role as it plays in natural image classification. Based on Eq.(4.24) and Eq.(4.25), we find that batch normalization actually forces the cover image and stego image to be distributed into opposite side of the batch mean in paired learning case. Thus, the batch normalization not only normalize the input data, but also can discriminate cover images and their stegos to some extent. However, the model would fail to predict the label of unpaired input images in the testing phase. To analyze this phenomenon, we first substitute μ

in Eq.(4.21) and Eq.(4.22) with Eq.(4.23) and then rewrite them as follows:

$$\mathbf{x}^{op} = \frac{\left[\mathbf{W}\mathbf{x} - E\left(\mathbf{W}\mathbf{x} + \frac{1}{2}\mathbf{W}\mathbf{s}\right)\right]}{\sigma} \circ \mathcal{H}\left[\frac{\mathbf{W}\mathbf{x} - E(\mathbf{W}\mathbf{x} + \frac{1}{2}\mathbf{W}\mathbf{s})}{\sigma}\right] \tag{4.31}$$

$$\mathbf{y}^{op} = \frac{\left[\mathbf{W}(\mathbf{x}+\mathbf{s}) - E\left(\mathbf{W}\mathbf{x} + \frac{1}{2}\mathbf{W}\mathbf{s}\right)\right]}{\sigma} \circ \mathcal{H}\left[\frac{\mathbf{W}(\mathbf{x}+\mathbf{s}) - E(\mathbf{W}\mathbf{x} + \frac{1}{2}\mathbf{W}\mathbf{s})}{\sigma}\right] \tag{4.32}$$

where $\circ$ represents the pointwise product, $\mathcal{H}(\cdot)$ is Heaviside step function:

$$\mathcal{H}(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{4.33}$$

For the unpaired case, outputs of the "Conv+BN+ReLU" block are not same to above equations. Assume the block is fed with a cover image $\mathbf{x}$ and a stego image $\mathbf{y}'$, where $\mathbf{y}' = \mathbf{x}' + \mathbf{s}$ and $\mathbf{x}' \neq \mathbf{x}$. For the "Conv+BN+ReLU" block, outputs of $\mathbf{x}$ and $\mathbf{y}'$ are:

$$\mathbf{x}^{ou} = \frac{\left[\mathbf{W}\mathbf{x} - \frac{1}{2}E\left(\mathbf{W}(\mathbf{x}+\mathbf{x}'+\mathbf{s})\right)\right]}{\sigma'} \circ \mathcal{H}\left[\frac{\mathbf{W}\mathbf{x} - \frac{1}{2}E\left(\mathbf{W}(\mathbf{x}+\mathbf{x}'+\mathbf{s})\right)}{\sigma'}\right] \tag{4.34}$$

$$\mathbf{y}^{ou} = \frac{\left[\mathbf{W}(\mathbf{x}'+\mathbf{s}) - \frac{1}{2}E\left(\mathbf{W}(\mathbf{x}+\mathbf{x}'+\mathbf{s})\right)\right]}{\sigma'} \circ \mathcal{H}\left[\frac{\mathbf{W}(\mathbf{x}'+\mathbf{s}) - \frac{1}{2}E\left(\mathbf{W}(\mathbf{x}+\mathbf{x}'+\mathbf{s})\right)}{\sigma'}\right] \tag{4.35}$$

where $\mathbf{x}^{ou}$ and $\mathbf{y}^{ou}$ represent the output of cover image and stego image in unpaired case respectively, σ' represent the variance of $\mathbf{W}\mathbf{x} + \mathbf{W}\mathbf{y}'$. The expected outputs of of $\mathbf{x}$ and $\mathbf{y}'$ after the batch normalization layer are:

$$E\left[\frac{\mathbf{W}\mathbf{x} - \frac{1}{2}E\left(\mathbf{W}(\mathbf{x}+\mathbf{x}'+\mathbf{s})\right)}{\sigma'}\right] = \frac{1}{2\sigma'}\left[E\left(\mathbf{W}(\mathbf{x}-\mathbf{x}')\right) - \frac{1}{2}E\left(\mathbf{W}\mathbf{s}\right)\right] \tag{4.36}$$

$$E\left[\frac{\mathbf{W}(\mathbf{x}'+\mathbf{s}) - \frac{1}{2}E\left(\mathbf{W}(\mathbf{x}+\mathbf{x}'+\mathbf{s})\right)}{\sigma'}\right] = \frac{1}{2\sigma'}\left[E\left(\mathbf{W}(\mathbf{x}'-\mathbf{x})\right) + \frac{1}{2}E\left(\mathbf{W}\mathbf{s}\right)\right] \tag{4.37}$$

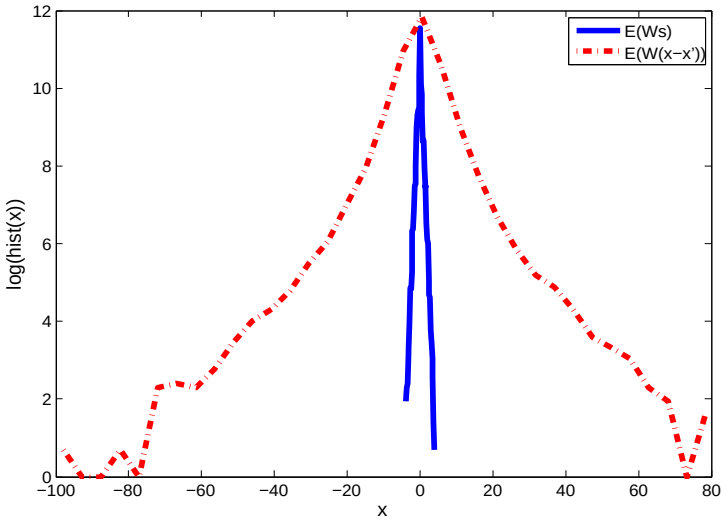


Figure 4.6: Histogram of elements in $\mathbf{Ws}$ and $\mathbf{W}(\mathbf{x}-\mathbf{x}')$. The feature map after the content suppression subnetwork is extracted and the steganographic algorithm S-UNIWARD at payload 0.4 bpp is used for demonstration.

Compared with the paired case, on average, the expected output of batch normalization layer in Eq.(4.36) and Eq.(4.37) not only depends $E\left[\mathbf{Ws}\right]$ but also $E\left[\mathbf{W}(\mathbf{x}-\mathbf{x}')\right]$. Fig.4.6 has shown the distribution of elements in $\mathbf{W}(\mathbf{x}-\mathbf{x}')$ and $\mathbf{Ws}$. We find that the amplitude of secret message $\mathbf{Ws}$ is smaller than $\mathbf{W}(\mathbf{x}-\mathbf{x}')$. Consequently, the output of the "Conv+BN+ReLU" block is dominated by cover images $\mathbf{x}$ rather than the secret message $\mathbf{s}$ in unpaired case. This characteristic leads to two direct results: (1). the path pattern selected by ReLU layer is largely determined by cover images; (2). the amplitude of the "Conv+BN+ReLU" output is largely determined by cover images. Both results make the feature map generated by the unpaired case significantly different from the feature map generated by the paired case.

For a network with many "Conv+BN+ReLU" blocks, the difference $\mathbf{W}(\mathbf{x}-\mathbf{x}')$ propagates through the whole network and finally makes the prediction incorrect. Actually, the output of cover image and stego image after several "Conv+BN+ReLU" blocks can be in an iterated form. For paired case:

$$\mathbf{x}_{n+1}^{op} = \frac{\mathbf{f}_{n+1}^{pc}}{\sigma_{n+1}} \circ \mathcal{H}\left[\frac{\mathbf{f}_{n+1}^{pc}}{\sigma_{n+1}}\right] \tag{4.38}$$

$$\mathbf{y}_{n+1}^{op} = \frac{\mathbf{f}_{n+1}^{ps}}{\sigma_{n+1}} \circ \mathcal{H}\left[\frac{\mathbf{f}_{n+1}^{ps}}{\sigma_{n+1}}\right] \tag{4.39}$$

$$\mathbf{f}_{n+1}^{pc} = \mathbf{W}_{n+1}\mathbf{x}_n^{op} - \frac{1}{2}E\left(\mathbf{W}_{n+1}\mathbf{x}_n^{op} + \mathbf{W}_{n+1}\mathbf{y}_n^{op}\right) \tag{4.40}$$

$$\mathbf{f}_{n+1}^{ps} = \mathbf{W}_{n+1}\mathbf{y}_n^{op} - \frac{1}{2}E\left(\mathbf{W}_{n+1}\mathbf{x}_n^{op} + \mathbf{W}_{n+1}\mathbf{y}_n^{op}\right) \tag{4.41}$$

where $\mathbf{x}_n^{op}$ and $\mathbf{y}_n^{op}$ represent the output of cover image and stego image after $n-$th "Conv+BN+ReLU" blocks in paired case, $\mathbf{W}_n$ denotes the convolution kernel and σ_n is the variance. For unpaired case:

$$\mathbf{x}_{n+1}^{ou} = \frac{\mathbf{f}_{n+1}^{uc}}{\sigma'_{n+1}} \circ \mathcal{H}\left(\frac{\mathbf{f}_{n+1}^{uc}}{\sigma'_{n+1}}\right) \tag{4.42}$$

$$\mathbf{y}_{n+1}^{ou} = \frac{\mathbf{f}_{n+1}^{us}}{\sigma'_{n+1}} \circ \mathcal{H}\left(\frac{\mathbf{f}_{n+1}^{us}}{\sigma'_{n+1}}\right) \tag{4.43}$$

$$\mathbf{f}_{n+1}^{uc} = \mathbf{W}_{n+1}\mathbf{x}_n^{ou} - \frac{1}{2}E\left[\mathbf{W}_{n+1}\mathbf{x}_n^{ou} + \mathbf{W}_{n+1}\mathbf{y}_n^{ou}\right] \tag{4.44}$$

$$\mathbf{f}_{n+1}^{us} = \mathbf{W}_{n+1}\mathbf{y}^{ou} - \frac{1}{2}E\left[\mathbf{W}_{n+1}\mathbf{x}_n^{ou} + \mathbf{W}_{n+1}\mathbf{y}_n^{ou}\right] \tag{4.45}$$

where $\mathbf{x}_n^{ou}$ and $\mathbf{y}_n^{ou}$ represent the output of cover image and stego image after $n-$th "Conv+BN+ReLU" blocks in unpaired case.

We expand the Eq.(4.38) and Eq.(4.39) after two "Conv+BN+ReLU" blocks:

$$\begin{aligned}\mathbf{x}_2^{op} =& \frac{\mathbf{W}_2(\mathbf{W}_1\mathbf{x} - E\left[\mathbf{W}_1\mathbf{x} + \frac{1}{2}\mathbf{W}_1\mathbf{s}\right])}{\sigma_2\sigma_1} \circ \mathcal{H}\left(\frac{\mathbf{f}_1^{pc}}{\sigma_1}\right) \circ \mathcal{H}\left(\frac{\mathbf{f}_2^{pc}}{\sigma_2}\right) \\ & - E\left(\frac{\mathbf{W}_2(\mathbf{W}_1\mathbf{x} - E\left[\mathbf{W}_1\mathbf{x} + \frac{1}{2}\mathbf{W}_1\mathbf{s}\right])}{\sigma_2\sigma_1} \circ \mathcal{H}\left(\frac{\mathbf{f}_1^{pc}}{\sigma_1}\right)\right) \circ \mathcal{H}\left(\frac{\mathbf{f}_2^{pc}}{\sigma_2}\right)\end{aligned} \tag{4.46}$$

$$\begin{aligned}\mathbf{x}_2^{ou} =& \frac{\mathbf{W}_2(\mathbf{W}_1\mathbf{x} - \frac{1}{2}E\left[\mathbf{W}_1(\mathbf{x} + \mathbf{x}' + \mathbf{s})\right])}{\sigma_2\sigma_1} \circ \mathcal{H}\left(\frac{\mathbf{f}_1^{uc}}{\sigma_1}\right) \circ \mathcal{H}\left(\frac{\mathbf{f}_2^{uc}}{\sigma_2}\right) \\ & - E\left(\frac{\mathbf{W}_2(\mathbf{W}_1\mathbf{x} - \frac{1}{2}E\left[\mathbf{W}_1(\mathbf{x} + \mathbf{x}' + \mathbf{s})\right])}{\sigma_2\sigma_1} \circ \mathcal{H}\left(\frac{\mathbf{f}_1^{uc}}{\sigma_1}\right)\right) \circ \mathcal{H}\left(\frac{\mathbf{f}_2^{uc}}{\sigma_2}\right)\end{aligned} \tag{4.47}$$

Observing the expanded equation, we find that $\mathbf{x}_2^{ou}$ is different from $\mathbf{x}_2^{op}$ in three as-

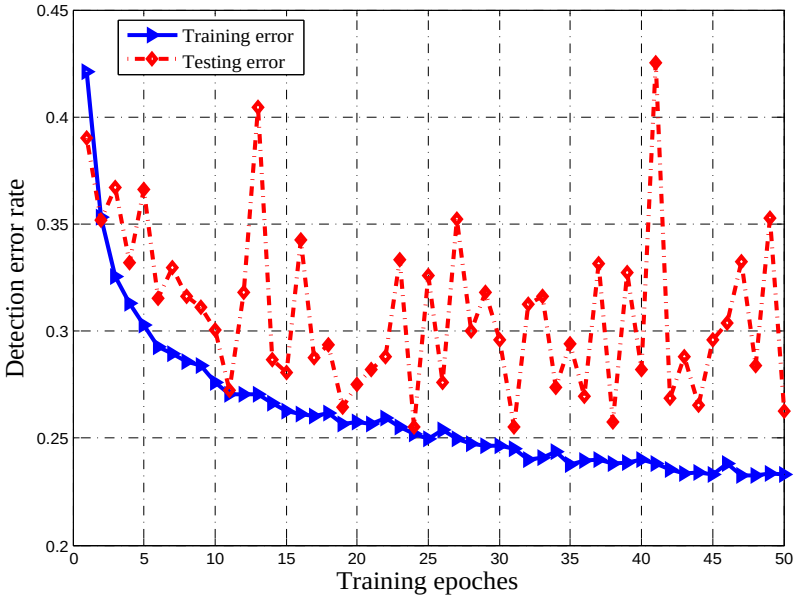


Figure 4.7: Testing error vibrates a lot if fixed parameters are used in batch normalization layers. The proposed model with 20 convolutional layers is used for demonstration. The tested steganographic algorithm is S-UNIWARD at payload 0.4 bpp.

pects: (1). the term $\mathbf{W}(\mathbf{x}' - \mathbf{x})$ exists in $\mathbf{x}_2^{ou}$, it does not decay as the number of "Conv+BN+ReLU" block increases; (2). the feature map $\mathbf{f}_1^{uc}$ is different from $\mathbf{f}_1^{pc}$, which also make $\mathbf{f}_2^{uc}$ different from $\mathbf{f}_2^{pc}$; (3). $\mathcal{H}(\mathbf{f}_i^{uc})$ is different from $\mathcal{H}(\mathbf{f}_i^{pc})$, $i \in \{1, 2\}$. Furthermore, the product of several $\mathcal{H}(\mathbf{f}_i^{uc})$ accumulates the difference and finally make the output completely different from the output in the paired case. Similar result can be found for $\mathbf{y}_2^{op}$ and $\mathbf{y}_2^{ou}$. Therefore, we can conclude that, for a network with many "Conv+BN+ReLU" blocks, it would give a poor detection result for the unpaired testing samples, when batch parameters are used in the batch normalization layer.

The performance of a network with batch normalization layers vibrates greatly if fixed μ and σ are used in the testing phase. The phenomenon is depicted in Fig.4.7. For the fixed parameter case, the output of a cover image $\mathbf{x}$ after two "Conv+BN+ReLU" blocks is:

$$\mathbf{x}_2^o = \frac{\mathbf{W}_2(\mathbf{W}_1\mathbf{x} - \mu_1)}{\sigma_1\sigma_2} \circ \mathcal{H}\left(\frac{\mathbf{f}_1^o}{\sigma_1}\right) \circ \mathcal{H}\left(\frac{\mathbf{f}_2^o}{\sigma_2}\right) - \frac{\mu_2}{\sigma_2} \circ \mathcal{H}\left(\frac{\mathbf{f}_2^o}{\sigma_2}\right) \tag{4.48}$$

where $\mu_1, \mu_2, \sigma_1, \sigma_2$ are fixed parameters of the batch normalization layer, $\mathbf{f}_1^o$ and $\mathbf{f}_2^o$ are defined as:

$$\mathbf{f}_1^o = \mathbf{W}_1\mathbf{x} - \mu_1 \tag{4.49}$$

$$\mathbf{f}_2^o = \mathbf{W}_2\mathbf{x}_1^o - \mu_2 \tag{4.50}$$

Actually, the discrimination of a cover image and a stego image depends on the stego sigal $\mathbf{s}$. However, $\mathbf{s}$ is generally small. Thus, inaccurate estimation to parameters $\mu_1, \mu_2, \sigma_1, \sigma_2$ will modify the output value $\mathbf{x}_1^o$, $\mathbf{f}_i^o$, and $\mathcal{H}(\mathbf{f}_i^o/\sigma_i)$. These accumulated modifications may surpass the stego sigal $\mathbf{s}$, and finally make an incorrect prediction.

4.8 Summary

This chapter introduced a unified convolutional neural network for image steganalysis. The proposed network has two improvements over previous CNN based steganalytic methods. On one hand, our network unifies image preprocessing and feature learning in a whole model. On the other hand, we proposed a novel subnetwork to actively preserve the weak stego signal based on residual learning. Experimental results and theoretical analysis have shown that the network has following main contributions.

- Adaptive content suppression can improve the detection accuracy. We analyzed that this content suppression subnetwork can also increase the network's convergence speed.

- Residual learning can effectively overcome the feature diminishing phenomenon in image steganalysis. In theory, we prove that either the weak stego signal or the difference between cover images and stego images does not decay as the depth of the network increases. Experimental results show that a well trained network can further enlarge the difference, thus obviously improve the detection accuracy to modern adaptive steganography.

- We analyzed the rationality of the proposed network when training images and test images are paired. We explained why there is a great performance loss when the test image are unpaired.

Current network shows promising performances on detecting spatial domain steganography when images are paired. We also analyzed the limitation of batch normalization for image steganalysis. In future works, we will develop CNN models to detect stego images without the batch normalization layer and further extend them into the compressed domain images.

Chapter 5

Conclusions and Future Work

This dissertation presents a series of studies to improve the performance of steganography and steganalysis based on natural image structures. In the chapter, we summarize the work presented in the thesis and discuss the future work.

5.1 Conclusions

Image steganography and steganalysis attract increasing interests because of its great potentials in military and commercial applications. Due to huge amounts of digital images in the internet, we are motivated to improve the performance of steganography and steganalysis by exploring the structures of natural images. Following this idea, the dissertation explores natural image structures for steganography and steganalysis from following two aspects:

1) Improving the undetectability of steganography by selecting suitable natural cover images. By taking the structural richness of natural images, this work aims to investigate what properties that make stego images undetectable and select suitable cover images to improve the undetectability of steganography. Based on statistical models of natural images, theoretically, we have derived a measure, which proves to be an upper bound of the Kullback-Leibler divergence between cover images and stego images. This measure, which is only determined by the distribution of images, is

used to analyze what properties of cover images that intrinsically affect steganographic security. With this measure, we conclude that the undetectability of the stego image relates to three factors: the entropy of the statistical model to represent the image, the energy of varying pixels across the image, and the number of nonzero DCT coefficients to reconstruct the image.

2) Improving the detection ability of steganalysis by modeling natural images with convolutional neural networks. Based on the property that CNNs have superior ability to capture correlations in natural images, this work proposed a novel CNN model for image steganalysis. By unifying image preprocessing and feature learning in a whole network, the model can adaptively suppress the image content so that the signal-to-noise ratio is increased. By incorporating residual learning in a novel subnetwork, the model can preserve the weak stego signal generated by message embedding at any depth. With these two improvements, the proposed can learn effective features for steganalysis when cover images and stego images are paired in training and testing.

5.2 Future Work

Though we have made progresses on using natural image structures for steganography and steganalysis, it is still far from perfect. A lot of work can be done to further improve the performance of steganography and steganalysis. In future, we can extend the work from the following two directions:

5.2.1 Design New Steganographic Algorithms based on Convolutional Neural Networks

Modern methods usually formulate steganography as a distortion minimization problem. To hide secret messages in images, they first define a distortion value for each image pixel/coefficient to represent its detectability. Then, hiding messages with least detectability is transformed to a problem of finding pixels/coefficients with minimal

distortions. All existing distortions in modern steganography are defined by hand. However, designing an effective distortion function proves to be a difficult task which need strong domain knowledge of steganography and steganalysis. To address this difficulty, in future, we could use a neural network model to automatically learn the distortion function that make existing steganalysis disable. The main advantage of this approach is that the difficulty of designing of distortion function is significantly reduced. In addition, the function space is enlarged that complex relationships among pixels/coefficients can be utilized to define the distortion function.

5.2.2 Develop New Convolutional Neural Networks for Image Steganalysis without Batch Normalization Layers

For image steganalysis, a CNN model with batch normalization layers is easily overfitted and sensitive the variation of parameters. These phenomenons are mainly due to the nature of the task: image steganalysis is to discriminate the cover image and the stego image which is the addition of weak stego signal and the cover image. The addition nature makes CNN models with batch normalization layers easily capture the difference between cover images and stego images in paired training, but fail to discriminate them when testing images are not paired. Weak stego signals make the difference between covers and stegos be very small, thus a slight variation to parameters of batch normalization layers would result in a great performance loss. To address these difficulties, in future, we will propose new CNN model without batch normalization layers for image steganalysis and further extend it to compressed domain images.

Bibliography

[1] A, Cheddad, J, Condell, K, Curran, and P. M. Kevitt. Digital image steganography: Survey and analysis of current methods. *Signal Processing*, 90(3): 727-752, 2010.

[2] A. D. Ker and T. Pevny. The steganographer is the outlier: Realistic large-scale steganalysis. *IEEE Transactions on Information Forensics and Security*, 9(9):1424-1435, 2014.

[3] A. D. Ker. Batch steganography and pooled steganalysis. In *Proceedings of 8th Information Hiding Workshop*, vol.4437, pp. 265-281, 2006.

[4] A. D. Ker. Estimating steganographic Fisher Information in real images. In *Proceeding of 11th Information Hiding Workshop*, vol. 5806, pp. 73-88, 2008.

[5] A. Hyvarinen, J. Hurri, and P. O. Hoyer. *Natural image statistics: a probabilistic approach to early computational vision*, Springer Press, 2009.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[7] A. Senior, G. Heigold, M. A. Ranzato, and K. Yang. An empirical study of learning rates in deep networks for speech recognition. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 22, pp. 6724-6728, 2013.

[8] A. Srivastava, A. B. Lee, E. P. Simoncelli, and S.-C. Zhu. On advances in statistical modeling of natural images. *Journal of Mathematical Imaging and Vision*, 18(1): 17-33, 2003.

[9] A. Torralba and A. Oliva. Statistics of natural image categories. *Network: Computation in Neural Systems*, vol. 14, pp. 391-412, 2003.

[10] A. Westfeld and A. Pltzmann. Attacks on Steganographic Systems. *Third International Workshop on Information Hiding*, pp. 61-76, 2000.

[11] A. Westfeld. F5-a steganographic algorithm high capacity despite better steganalysis. In *Proceedings of Fourth International Workshop on Information Hiding*, vol. 2137, pp. 289-302, 2001.

[12] B. Chen, and G. W. Wornell. Quantization index modulation: a class of provably good methods for digital watermarking and information embedding. *IEEE Transactions on Information Theory*, vol. 47, pp. 1423-1443, 2001.

[13] B. Li, J. He, J. Huang, and Y. Q. Shi. A survey on image steganography and steganalysis. *Journal of Information Hiding and Multimedia Signal Processing*, 2(2): 142-172, 2011.

[14] B. Li, M. Wang, J. Huang, and X. Li. A new cost function for spatial image steganography. *IEEE International Conference on Image Processing (ICIP)*, pp. 4206-4210, 2014.

[15] C. Chang, T. Chen and L. Chung. A steganographic method based upon JPEG and quantization table modification. *Information Science*, vol.141, pp. 123-138, 2002.

[16] C. Dong, C. C. Loy, K. He, and X. Tang. Image uper-resolution using deep convolutional networks. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014.

[17] C. Kurak and J. M. Hugh. A cautionary note on image downgrading. *Computer Security Applications Conference*, pp.153-159, 1992.

[18] C. Yang, C. Weng, S. Wang, H. Sun. Adaptive data hiding in edge areas of images with spatial LSB domain systems. *IEEE Transactions on Information Forensics and Security*, vol.3, pp. 488-497, 2008.

[19] D. L. Ruderman and W. Bialek. Statistics of natural images: scaling in the woods. *Physical Review Letters*, 73(6): 814-817, 1994.

[20] D. L. Ruderman. Origins of scaling in natural images. *Vision Research*, vol. 37, pp. 3385-3398, 1997.

[21] D. L. Ruderman. The statistics of natural images. *Network: Computation in Neural Systems*, vol. 5, pp. 517-548, 1994.

[22] D. P. Palomar and S. Verdu. Gradient of mutual information in linear vector Gaussian channels. *IEEE Transactions on Information Theory*, 52(1): 141-154, 2006.

[23] D. Reynolds. Gaussian mixture models. *Encyclopedia of Biometric Recognition*, pp. 659-663, 2009.

[24] D. Wu and W. Tsai. A steganographic method for images by pixel-value differencing. *Pattern Recognition Letters*, vol.24, pp. 1613-1626, 2003.

[25] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. *IEEE International Conference on Computer Vision (ICCV)*, 2011.

[26] D. Zoran and Y. Weiss. Natural images, Gaussian Mixtures and dead leaves. *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[27] D. Zou. Steganalysis based on markov model of thresholded prediction-error Image. *IEEE International Conference on Multimedia and Expo*, pp. 1365-1368, 2006.

[28] E. Kawaguchi and R. O. Eason. Principle and applications of BPCS steganography. *Multimedia Systems and Applications*, vol. 3528, pp.464-473, *SPIE*, 1998.

[29] E. Y. Lam and J. W. Goodman. A mathematical analysis of the DCT coefficient distributions for images. *IEEE Transactions on Image Processing*, 9(10): 1661-1666, 2000.

[30] F. Li, K. Wu, J. Lei, M. Wen, Z. Bi, and C. Gu. Steganalysis over large-scale social networks with high-order joint features and clustering ensembles. *IEEE Transactions on Information Forensics and Security*, 11(2): 344-357, 2016.

[31] G. Huang, Y. Sun, Z. Liuy, D. Sedra, and K. Q. Weinberger. Deep networks with stochastic depth. *arXiv:1603.09382v3*, 2016.

[32] G. J. Simmons. The prisoners problem and the subliminal channel. In *Advances in Cryptology: Proceedings of Crypto*, Plenum Press, pp. 51-67, 1984.

[33] G. J. Simmons. The prisoners problem and the subliminal channel. In *Advances in Cryptology: Proceedings of Crypto*, Plenum Press, pp. 51-67, 1984.

[34] G. Xu, H. Wu, and Y. Q. Shi. Structural design of convolutional neural networks for steganalysis. *IEEE Signal Processing Letters*, vol. 23, pp. 708-712, 2016.

[35] H. Farid. Detecting steganographic messages in digital images. *Technical Report 2001-412*, Department of Computer Science, Dartmouth College, 2001.

[36] H. M. Schopf and P. H. Supancic. On Burmann's theorem and its application to problems of linear and nonlinear heat transfer and diffusion. *The Mathematica Journal*, 2014.

[37] H. Sajedi and M. Jamzad. Secure cover selection steganography. *Advances in Information Security and Assurance*, vol. 5576, pp. 317-326, 2009.

[38] H. Su, R. Qi, Y. Li, and L. J. Guibas. Render for CNN: viewpoint estimation in images using CNNs trained with rendered 3D model views. In *Internaitonal Conference on Computer Vision (ICCV)*, 2015.

[39] $http://www.mikebarney.net/stego.html$

[40] $http://www.slideshare.net/kleinerperkins/2016-internet-trends-report/65-KPCB\_INTERNET\_TRENDS\_2016\_PAGE$

[41] J. C. Hernandez-Castroa, I. Blasco-Lopezb, J. M. Estevez-Tapiadora, and A. Ribagorda-Garnachoa. Steganography in games: a general methodology and its application to the game of Go. *Computers & Security*, 25(1): 64-71, 2006.

[42] J. F. Couchot, R. Couturier, C. Guyeux, and M. Salomon. Steganalysis via a convolutional neural network using large convolution filters for embedding process with same stego key. *arXiv:1605.07946v3*, 2016.

[43] J. Fridrich and M. Goljan. Digital image steganography using stochastic modulation. *Proceeding SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents V*, 2003.

[44] J. Fridrich and M. Goljan. On estimation of secret message length in lsb steganography in spatial domain. *Proc. of IST/SPIE Electronic Imaging: Security, Steganography, and Watermarking of Multimedia Contents VI*, vol. 5306, pp. 23-34, 2004.

[45] J. Fridrich and M. Goljan. Practical steganalysis of digital images: state of the art. *Proceeding SPIE on Security and Watermarking of Multimedia Contents IV*, vol. 4675, 2002.

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

[47] J. Fridrich, D. Soukal, and M. Goljan. Maximum likelihood estimation of length of secret message embedded using $\pm k$ steganography in spatial domain. *SPIE Pro-*

ceedings on Security, Steganography, and Watermarking of Multimedia Contents VII, vol.5681, pp. 595-606, 2005.

[48] J. Fridrich, M. Goljan, and D. Hogea. Attacking the OutGuess. In *Proceedings of the ACM Workshop on Multimedia and Security*, 2002.

[49] J. Fridrich, M. Goljan, P. Lisonek and D. Soukal. Writing on wet paper. *IEEE Transactions on Signal Processing*, 53(10): 3923-3935, 2005.

[50] J. Fridrich, T. Pevny, and J. Kodovsky. Statistically undetectable JPEG steganography: Dead ends, challenges, and opportunities. In *Proceedings of the 9th ACM Multimedia & Security Workshop*, pp. 3-14, 2007.

[51] J. Fridrich. Effect of cover quantization on steganographic fisher information. *IEEE Transactions on Information Forensics and Security*, 8(2): 361-373, 2013.

[52] J. Harmsen and W. Pearlman. Steganalysis of additive-noise modelable information hiding. In *Proceedings SPIE Security Watermarking Multimedia Contents*, vol. 5020, pp. 131-142, 2003.

[53] J. Huang. *Statistics of natural images and models*. PhD thesis of Brown University, 2000.

[54] J. Kodovsky and J. Fridrich. Effect of image downsampling on steganographic security. *IEEE Transactions on Information Forensics and Security*, 9(5): 752-762, 2014.

[55] J. Kodovsky and J.Fridrich. Influence of embedding strategies on security of steganographic methods in the JPEG domain. *SPIE Electronic Imaging, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, pp. 1-13, 2008.

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

[57] J. Mielikainen. LSB matching revisited. *IEEE Signal Processing Letters*, vol.13, pp. 285-287, 2006.

[58] J. Mielikainen. LSB matching revisited. *IEEE Signal Processing Letters*, vol.13, pp. 285-287, 2006.

[59] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, 12(11): 1338-1351, 2003.

[60] J. R. William. The Normal-Laplace distribution and its relatives. *Advances in Distribution Theory, Order Statistics, and Inference*, pp. 61-74, 2006.

[61] Jsteg, $ftp://ftp.funet.fi/pub/crypt/steganography/$ [37]

[62] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[63] K. Simonyan and A. Zisserman. Very deep convolutional networks for largescale image recognition. *International Conference on Learning Representation (ICLR)*, 2015.

[64] L. A. Gatys, A. S. Ecker, and M. Bethge. Texture synthesis using convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.

[65] L. Guo, J. Ni, and Y. Q. Shi. Uniform embedding for efficient JPEG steganography. *IEEE Transactions on Information Forensics and Security*, vol.9, pp. 814-825, 2014.

[66] L. Guo, J. Ni, W. Su, C. Tang, and Y. Q. Shi. Using Statistical image model for JPEG steganography: uniform embedding revisited. *IEEE Transactions on Information Forensics and Security*, vol. 10, pp. 2669-2680, 2015.

[67] L. Pibre, D. Ienco, M. Chaumont. Deep Learning for steganalysis is better than a Rich Model with an Ensemble Classifier, and is natively robust to the cover source-mismatch. *Media Watermarking, Security, and Forensics, IS&T Int. Symp. on Electronic Imaging*, 2016.

[68] L. Zhang, H. Wang and R. Wu. High-capacity steganography scheme for JPEG2000 baseline system. *IEEE Transactions on Image Processing*, 18(8): 1797-1803, 2009.

[69] M. Bianchini and F. Scarselli. On the complexity of neural network classifiers: a comparison between shallow and deep architectures. *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 8, 2014.

[70] M. Goljan, J. Fridrich, and T. Holotyak. New blind steganalysis and its implications. In *Proceeding of the SPIE, Security, Steganography, and Watermarking of Multimedia Contents VI*, vol. 6072, pp. 1-13, 2006.

[71] M. J. Huiskes and M. S. Lew. The MIR Flickr Retrieval Evaluation. *ACM International Conference on Multimedia Information Retrieval*, Vancouver, Canada, 2008.

[72] M. J. Z. Kermani. A robust steganography algorithm based on texture similarity using gabor filter. In *IEEE Symposium on Signal processing and Information Technology*, pp. 578-582, 2005.

[73] M. Kharrazi, H. T. Sencar and N. Memon. Cover selection for steganographic embedding. in *ICIP*, pp. 117-120, 2006.

[74] M. Kharrazi, H. T. Sencar, and N. Memon. Benchmarking steganographic and steganalysis techniques. *Proceedings of the SPIE*, vol. 5681, pp. 252-263, 2005.

[75] N. Provos. Defending Against Statistical Steganalysis. In *Proceeding 10th USENIX Security Symposium*, Washington, DC, 2001.

[76] P. Bas, T. Filler and T. Pevny. BOSS (break our steganography system), *http* : $//boss.gipsa-lab.grenoble-inp.fr$, 2009.

[77] P. Sallee. Model based steganography. *International Workshop on Digital Watermarking*, pp. 174-188, 2003.

[78] P. Su and C. J. Kuo. Steganography in JPEG2000 compressed images. *IEEE Transactions on Consumer Electronics*, 49(4): 824-832, 2003.

[79] Q. Liu. Steganalysis of DCTVembedding based adaptive steganography and YASS. In *Proceedings of the 13th ACM Multimedia & Security Workshop*, pp. 77-86, 2011.

[80] R. Chandramouli, M. Kharrazi, and N. Memon. Image steganography and steganalysis: concepts and practice.*Lecture Note Series, Institute for Mathematical Sciences, National University of Singapore*, pp.35-49, 2004.

[81] R. Cogranne and F. Retraint. An asymptotically uniformly most powerful test for LSB matching detection. *IEEE Transactions on Information Forensics and Security*, 8(3): 464-476, 2013.

[82] R. Cogranne, C. Zitzmann, L. Fillatre, F. Retraint, I. Nikiforov, P. Cornu. A cover image model for reliable steganalysis. *Information Hiding*, pp. 178-192, 2011.

[83] R. Cogranne. A cover image model for reliable steganalysis. *13th International Conference on Information Hiding*, pp. 178-192, 2011.

[84] R. Crandall, Some notes on steganography. Posted on Steganography Mailing List, $http://os.inf.tu-dresden.de/\ westfeld/crandall.pdf$, 1998.

[85] R. M. Neal and G. E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. *Learning in Graphical Models*, pp. 355-368, 1998.

[86] R. Reininger and J. D. Gibson. Distributions of the two-dimensional DCT coefficients for images. *IEEE Transactions on Communications*, 31(6): 835-839, 1983.

[87] R. Zhang. An effcient embedder for BCH coding for steganography. *IEEE Transactions on Information Theory*, vol.58, pp. 7272-7279, 2009.

[88] Ross J. Anderson, Proceedings of the First International Workshop on Information Hiding, $http://dl.acm.org/citation.cfm?id = 647594\&picked = prox$, 1996.

[89] S. Dumitrescu, X. L. Wu, and Z. Wang. Detection of lsb steganography via sample pair analysis. *IEEE Transactions Signal Processing*, 51(7): 1995-2007, 2003.

[90] S. Geman and C. Graffigne. Markov random field image models and their applications to computer vision. In *Proceedings of International Congress on Mathematicians*, 1986.

[91] S. Ioffe and C. Szegedy. Batch normalization: accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, 2015.

[92] S. Lyu, and H. Farid. Steganalysis using higher-order image statistics. *IEEE Transactions on Information Forensics and Security*, 1(1):111-119, 2006.

[93] S. Roth and M. J. Black. Fields of Experts: a framework for learning image priors. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2(2): 860-867, 2005.

[94] S. Tan and B. Li. Stacked convolutional auto-encoders for steganalysis of digital images, in *Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2014.

[95] T. Denemark, V. Sedighi, V. Holub, R. Cogranne, and J. Fridrich. Selection-channel-aware rich model for steganalysis of digital images. *IEEE Workshop on Information Forensic and Security (WIFS)*, 2014.

[96] T. Filler, J. Judas, and J. Fridrich. Minimizing additive distortion in steganography using Syndrome-Trellis Codes. *IEEE Transactions on Information Forensics and Security*, vol.6, pp. 920-935, 2011.

[97] T. Holotyak, J. Fridrich, and D. Soukal. Stochastic approach to secret message length estimation in ± embedding steganography. In *Proceedings of SPIE Electronic Imaging*, pp. 673-684, 2005.

[98] T. Holotyak, J. Fridrich, and S. Voloshynovskiy. Blind statistical steganalysis of additive steganography using wavelet higher order statistics. In *Proceedings of the 9th IFIP Conference on Communications and Multimedia Security*, 2005.

[99] T. Pevny and J. Fridrich. Benchmarking for steganography. *Information Hiding Conference*, vol. 5284, pp. 251-267, 2008.

[100] T. Pevny and J. Fridrich. Merging Markov and DCT features for multiclass JPEG steganalysis, In *Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, v. 6505, pp. 1-14, 2007.

[101] T. Pevny, P. Bas, and J. Fridrich. Steganalysis by subtractive pixel adjacency matrix. *IEEE Transactions on Information Forensics and Security*, 5(2): 215-224, 2010 [48]

[102] T. Pevny, T. Filler and P. Bas. Using high-dimensional image models to perform highly undetectable steganography. In *Information Hiding Conference*, pp. 161-177, 2010.

[103] T. Sharp. An implementation of key-based digital signal steganography. *In Proceeding of Information Hiding Workshop*, vol.2137, pp. 13-26, 2001.

[104] V. Holub and J. Fridrich. Designing steganographic distortion using directional filters. *IEEE Workshop on Information Forensic and Security (WIFS)*, 2012.

[105] V. Holub and J. Fridrich. Low complexity features for JPEG steganalysis using undecimated DCT. *IEEE Transactions on Information Forensics and Security*, vol. 10, pp. 219-228, 2014.

[106] V. Holub, and J.Fridrich. Projections of residuals for digital image steganalysis. *IEEE Transactions on Information Forensics and Security*, 8(12): 1996-2006, 2013.

[107] V. Holub, J. Fridrich, and T. Denemark. Universal distortion function for steganography in an arbitrary domain. *EURASIP Journal on Information Security*, 2014(1): 1-13, 2014.

[108] V. Jain and S. Seung. Natural image denoising with convolutional networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2008.

[109] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, 2010.

[110] V. Sedighi, R. Cogranne, and J. Fridrich. Content-Adaptive steganography by minimizing statistical detectability. *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 221-234, 2016.

[111] W. Bender, D. Gruhl, N. Morimoto, and A. Lu. Techniques for data hiding. *IBM System Journal*, vol. 35, pp. 313-336, 1996.

[112] W. Luo, F. Huang, and J. Huang. Edge adaptive image steganography based on LSB matching revisited. *IEEE Transactions on Information Forensics and Security*, 5(2): 201-214, 2010.

[113] X. Zhang and S. Wang. Steganography using multiple-base notational system and human vision sensitivity. *IEEE Signal Processing Letters*, vol.12, pp. 67-70, 2005.

[114] Y. Bengio, P. Simard, P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157-166, 1994.

[115] Y. Kim, Z. Duric, and D. Richards. Modified matrix encoding technique for minimal distortion steganography. *Proceedings of the 8th International Conference on Information Hiding*, pp. 314-327, 2006.

[116] Y. Petrov and L. Zhaoping. Local correlations, information redundancy, and sufficient pixel depth in natural images. *Journal of the Optical Society of America*, 20(1): 56-66, 2003.

[117] Y. Qian, J. Dong, W. Wang, and T. Tan. Deep learning for steganalysis via convolutional neural networks. *Media Watermarking, Security, and Forensics*, 2015.

[118] Y. Weiss and W. T. Freeman. What makes a good model of natural images? *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[119] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su. Reversible data hiding. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, pp. 354-362, 2006.