

Binghamton University

The Open Repository @ Binghamton (The ORB)

Graduate Dissertations and Theses

Dissertations, Theses and Capstones

2018

Side-information for steganography design and detection

Tomas Denmark

Binghamton University--SUNY, tdenema1@binghamton.edu

Follow this and additional works at: https://orb.binghamton.edu/dissertation_and_theses



Part of the [Electrical and Computer Engineering Commons](#)

Recommended Citation

Denemark, Tomas, "Side-information for steganography design and detection" (2018). *Graduate Dissertations and Theses*. 42.

https://orb.binghamton.edu/dissertation_and_theses/42

This Dissertation is brought to you for free and open access by the Dissertations, Theses and Capstones at The Open Repository @ Binghamton (The ORB). It has been accepted for inclusion in Graduate Dissertations and Theses by an authorized administrator of The Open Repository @ Binghamton (The ORB). For more information, please contact ORB@binghamton.edu.

SIDE-INFORMATION FOR STEGANOGRAPHY
DESIGN AND DETECTION

BY

TOMÁŠ DENEMARK

MS, Czech Technical University, Prague, 2012

DISSERTATION

Submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate School of Binghamton University
State University of New York
2018

Accepted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate School of Binghamton University
State University of New York
2018

May 1st, 2018

Jessica Fridrich, Chair and Faculty Advisor
Department of Electrical and Computer Engineering, Binghamton University

Matthias Kirchner, Member
Department of Electrical and Computer Engineering, Binghamton University

Scott Craver, Member
Department of Electrical and Computer Engineering, Binghamton University

Ronald Miles, Outside Examiner
Department of Mechanical Engineering, Binghamton University

Abstract

Today, the most secure steganographic schemes for digital images embed secret messages while minimizing a distortion function that describes the local complexity of the content. Distortion functions are heuristically designed to predict the modeling error, or in other words, how difficult it would be to detect a single change to the original image in any given area. This dissertation investigates how both the design and detection of such content-adaptive schemes can be improved with the use of side-information.

We distinguish two types of side-information, public and private:

Public side-information is available to the sender and at least in part also to anybody else who can observe the communication. Content complexity is a typical example of public side-information. While it is commonly used for steganography, it can also be used for detection. In this work, we propose a modification to the rich-model style feature sets in both spatial and JPEG domain to inform such feature sets of the content complexity.

Private side-information is available only to the sender. The previous use of private side-information in steganography was very successful but limited to steganography in JPEG images. Also, the constructions were based on heuristic with little theoretical foundations. This work tries to remedy this deficiency by introducing a scheme that generalizes the previous approach to an arbitrary domain. We also put forward a theoretical investigation of how to incorporate side-information based on a model of images. Third, we propose to use a novel type of side-information in the form of multiple exposures for JPEG steganography.

Acknowledgements

I would like to give my most profound gratitude to my Ph.D. adviser, Jessica Fridrich, for her support and guidance through my studies. She created a research environment that motivated me to push and improve myself professionally but also a friendly atmosphere which made me feel welcome and appreciated. For years, I will remember how she opened her home and invited all of us, students, as friends to join her and her family every Thanksgiving.

The current and past members of the Digital Data Embedding laboratory and the visiting scholars that joined us in our work, also deserve a tremendous amount of my thanks. They provided me with help, encouragement, and friendship.

Further, I wish to acknowledge my friends who made the lazy town of Binghamton so much more enjoyable and memorable. Mainly I would like to thank Tomáš Vávra, who was with me the whole time at least over the internet, and Kevin Hexemer, one of the kindest friends I ever had.

I owe an enormous debt to my family for incredible support and patience they offered to me before and after departing for my studies.

My most earnest acknowledgment goes to my wife, Toni. She provided me with love, trust, appreciation, emotional support, and gave me the strength to overcome all the trials. I thank her from the bottom of my hearth.

Lastly, I would like to acknowledge the financial support from NSF research grant No. 1561446 and FA9950-12-1-0124 by Air Force Office of Scientific Research that funded my work.

Tomáš Denemark
Binghamton, 2018

Contents

Preface	xxiii
1 Introduction and preliminaries	1
1.1 The steganographic channel	1
1.2 Steganographic security and steganalysis	2
1.2.1 Probabilistic model	2
1.2.2 Machine learning approach	3
1.2.2.1 Databases	3
1.2.2.2 Rich models	3
1.2.2.3 SRM	4
1.2.2.4 PSRM	5
1.2.2.5 Phase-aware JPEG feature sets	6
1.2.2.6 Classifiers	6
1.3 Content-adaptive steganography	7
1.4 Side-information	8
1.5 JPEG compression	9
2 Selection-channel aware attacks in spatial domain	11
2.1 Experimental setup	12
2.2 maxSRM	13
2.3 Experiments	14
2.4 Conclusion	16
3 Selection-channel aware attacks in DCT domain	19
3.1 Residual distortion measure	19
3.1.1 Final feature design	22
3.2 Experimental results	22
3.3 Conclusions	26

4	Selection-channel aware attacks in residual domain	29
4.1	Replacing change rates with L_1 distortion of residuals	29
4.1.1	Linear residuals	29
4.1.2	Non-linear residuals	31
4.2	Experiments	31
4.3	Conclusions	33
5	Steganography with precover	35
5.1	Discussion and relationship to prior art	37
5.2	Experiments	37
5.2.1	Spatial domain	37
5.2.2	JPEG domain	39
5.3	Conclusions	40
6	Model based side-informed steganography	43
6.1	Modeling acquisition	43
6.2	Side-informed steganography with multivariate Gaussian acquisition noise	44
6.2.1	Extension to JPEG domain	47
6.3	Connection to heuristic schemes	47
6.3.1	Model-based SI-MiPOD	48
6.3.2	Heuristic SI-MiPOD	49
6.3.3	Comparison of model-based and heuristic MiPOD	50
6.4	Experiments	51
6.4.1	Image sources	51
6.4.2	Spatial domain (BOSSColor)	51
6.4.3	JPEG domain	52
6.5	Public vs. private side-information and adaptivity	54
6.6	Conclusions	55
7	Multiple exposures	57
7.1	Preliminaries	57
7.2	Steganography with precover	58
7.3	Steganography with multiple JPEGs	61
7.3.1	Two exposures	61
7.3.2	Multiple exposures	62
7.4	Study with simulated acquisition noise	62
7.5	Datasets for experiments	64
7.5.1	BURSTbase	64

7.5.2	BURSTbaseH	65
7.6	Experiments	66
7.6.1	BURSTbase	66
7.6.2	BURSTbaseH	67
7.7	Conclusions	71
8	Natural steganography	73
8.1	Natural steganography in JPEG domain	73
8.2	Model in the spatial domain	73
8.2.1	Model in the DCT domain	74
8.2.2	Discussion	75
8.3	Overview of the algorithms	76
8.4	Database acquisition and shot noise distribution	78
8.4.1	Acquisition process	78
8.4.2	Shot noise distribution	79
8.5	Experiments	80
8.5.1	Results on MonoBase	80
8.5.2	Results on E1Base	81
8.5.3	Discussion	82
8.6	Conclusions and perspectives	84
9	Conclusion	87
A	SI-UNIWARD as published	89
B	Cost modulation as a function of quality factor	91
	Bibliography	93

List of tables

2.1	Effect of the co-occurrence scans on the detection error \bar{P}_E . WOW at 0.4 bpp, the 338-dimensional SQUARE submodel of SRM.	14
2.2	Average detection error \bar{P}_E for three embedding algorithms and four steganalysis feature sets.	15
3.1	Average P_E for three steganographic schemes for DCTR, GFR, and PHARM features and their selection-aware $\delta_{uSA}^{1/2}$ version for selected payloads, JPEG quality factors 75 and 95.	25
4.1	Detection of three steganographic algorithms for two payloads on BOSSbase 1.01 using the original maxSRM features and their proposed σ maxSRM form.	32
5.1	Mean E_{OOB} for HILL, S-UNIWARD and their side-informed variants when utilizing the quantization error after resizing with different kernels at 0.4 bpp.	40
6.1	Detection error when steganalyzing heuristic SI-MiPOD and model-based (MB) SI-MiPOD with SRM and maxSRMd2 and their JPEG counterparts with SRM/GFR (ignorant Warden), selection-channel-aware maxSRMd2/GFR (Warden aware of content,), and omniscient Warden aware of both of content and rounding error	55
7.1	Maximum and average MSE between two closest exposures from each burst in BURSTbaseH when constraining it to a fraction γ of best bursts.	66
7.2	Empirical security \bar{P}_E of embedding schemes, \mathcal{M} , J-UNIWARD, J2-UNIWARD, J2-UNIWARD implemented using STCs, and SI-UNIWARD on BURSTbase for a range of payloads, R , and quality factors.	68
7.3	Empirical security \bar{P}_E of embedding schemes, \mathcal{M} , J-UNIWARD, J2-UNIWARD and SI-UNIWARD on BURSTbaseH for a range of payloads, R , and quality factors for $\gamma = 0.1$	68
7.4	Empirical security \bar{P}_E of embedding schemes, \mathcal{M} , UED-JC, UED2-JC, and SI-UED-JC on BURSTbaseH for two payloads and two JPEG quality factors for $\gamma = 0.1$	69
8.1	Average payload in bits per pixel per MonoBase image embedded by Approach 3.	81
8.2	Minimum detection error when steganalyzing the NS in MonoBase with SRM, GFR, DCTR, and cc-JRM feature sets for five different approaches and a range of JPEG quality factors for a switch from ISO 320 to ISO 1000.	82

8.3	Detection error P_E when steganalyzing the NS in E1Base with SRM, GFR, DCTR, and cc-JRM feature sets for different approaches and JPEG quality factors for ISO switch 100 to 200. Note that the embedding capacity of SI-UNIWARD is limited to 1 bpnzac.	82
8.4	Average embedding rate for Approach 4 (MC pmf), E1Base.	83
8.5	Comparison between Approach 2 (simulated noise), which preserves intra-block dependencies, Approach 4 (independent embedding at each DCT coefficient), and simulated noise sampled independently for each DCT block.	84

List of figures

1.1.1	Components of the steganographic channel.	2
2.1.1	A set of 32 Gabor filters used in S-UNIGARD.	13
2.2.1	Four types of co-occurrence scan direction.	14
2.3.1	Embedding probability for payload 0.4 bpp using WOW (top right), S-UNIWARD (bottom left), and S-UNIGARD (bottom right) for a 128×128 grayscale cover image shown in top left (a 128×128 crop of '1013.pgm' from BOSSbase).	16
2.3.2	Detection error for all three algorithms when steganalyzing with maxSRMd2 with a fixed test payload ($\hat{\alpha} = 0.1$ for WOW and $\hat{\alpha} = 0.2$ for S-UNIWARD), versus the test payload set to the real payload, $\hat{\alpha} = \alpha$	17
2.3.3	Detection error increase when steganalyzing with the test payload $\hat{\alpha}$ chosen as in the text and the true payload α . The payload (x coordinates) were shifted by a small amount to prevent the markers and error bars from overlapping.	17
2.3.4	Improvement in detection error when steganalyzing with SRM versus maxSRMd2 or tSRM. The threshold in tSRM was optimized for each payload. From top down and from left right: WOW, S-UNIWARD, and S-UNIGARD.	18
3.1.1	Plot of $\delta_{uSA}^{1/2}$ versus δ_{EA} for one BOSSbase image for the DCTR filter bank. The first number pair above each scatter plot indicates the DCTR kernel (the spatial frequency of the DCT mode) while the second pair is the JPEG phase. Note that the square root forces an approximate linear relationship between both quantities.	23
3.2.1	Average P_E for three steganographic algorithms for DCTR, GFR, and PHARM features (patterns) and their selection-aware $\delta_{uSA}^{1/2}$ version (solid fill) versus payload, JPEG quality factors 75 and 95.	27
4.2.1	Detection error for HILL with spamPSRM, spamPSRM, and maxSRMd2.	32
4.2.2	Detection error for MVG with spamPSRM, spamPSRM, and maxSRMd2.	33
4.2.3	Detection error for S-UNIWARD with spamPSRM, σ spamPSRM, and maxSRMd2.	34
4.2.4	Detection error for WOW with spamPSRM, spamPSRM, and maxSRMd2.	34
5.2.1	Mean E_{OOB} for HILL (top) and S-UNIWARD (bottom) and their SI versions with the quantization error after resizing with Lanczos 3 kernel as the side-information when computing the costs from the unquantized and quantized cover.	38
5.2.2	By rows: cover image, its detail, embedding changes for HILL and SI3-HILL-U at 0.4 bpp for resizing with Lanczos 3 kernel.	39

5.2.3	Change rate for HILL and S-UNIWARD and their SI versions when resizing with Lanczos 3 kernel. The values are averages over all 10,000 images in our source. . . .	39
5.2.4	Mean E_{OOB} for HILL, S-UNIWARD and their SI versions with the quantization error after RGB to grayscale conversion as the side-information.	40
5.2.5	Mean E_{OOB} for SRM for HILL, S-UNIWARD and their SI versions with the quantization error after color depth reduction as the side-information.	41
5.2.6	Mean E_{OOB} for J-UNIWARD and its side-informed variants when utilizing the quantization error after JPEG compression when computing the costs from the unquantized cover and the cover. The used quality factor is 75%.	42
6.3.1	Embedding change probability $\beta^{(\text{SI})}$ as a function of variance σ^2 on a synthetic cover for $\alpha = 0.3$ nats using heuristic side-informed binary MiPOD (left) and model-based binary MiPOD (right).	51
6.4.1	Security of MiPOD and its heuristic and model-based SI versions with SI in the form of precover obtained by converting BOSSColor images to grayscale.	52
6.4.2	Security of two non side-informed and three side-informed embedding schemes as a function of payload in bpnzac on BOSSbase for JPEG quality factors 65, 75, 85, and 95.	53
6.4.3	Security of two non side-informed and three side-informed embedding schemes on BOSSbase as a function of JPEG quality factor for relative payload 0.2 bpnzac (left) and 0.4 (right).	53
7.2.1	Relative number of correctly and incorrectly determined embedding directions for steganography informed by the values of non-rounded DCT coefficients (precover) and by two JPEG images. See Section 7.4 for details.	58
7.2.2	Optimal modulation factor $m(Q)$ as a function of the JPEG quality factor Q . Left: BOSSbase 1.01 images with simulated acquisition noise. Right: BURSTbase. . . .	59
7.2.3	Empirical security, \bar{P}_E , as a function of the JPEG quality factor for relative payload $R = 0.4$ bpnzac for J2-UNIWARD, J-UNIWARD, and SI-UNIWARD. BOSSbase with simulated acquisition noise, low-complexity linear classifier trained with GFR.	60
7.2.4	MSE between $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(k)}$, $k = 2, \dots, 7$ from each burst averaged over all 9,310 bursts from BURSTbase. See Section 7.5 for notation and further details.	60
7.5.1	Gray dots: $\text{MSE}(\mathbf{X}^{(1)}, \mathbf{X}^{(2)})$ vs. average grayscale of $\mathbf{X}^{(1)}$ across images from BURSTbase. Circles: acquisition noise variance estimated from images of gray wall. Both at ISO 200.	65
7.6.1	Empirical security \bar{P}_E of J2-UNIWARD as a function of the JPEG quality factor Q on BURSTbase. Comparison with previous art for $R = 0.2$ bpnzac.	69
7.6.2	Empirical security \bar{P}_E of J2-UNIWARD as a function of the JPEG quality factor Q on BURSTbase. J2-UNIWARD \bar{P}_E for $R \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$ bpnzac, embedding simulated at rate-distortion bound.	69
7.6.3	Empirical security \bar{P}_E of J2-UNIWARD when the k th closest image from each burst from BURSTbase was used as side-information. Payload $R = 0.4$ bpnzac.	70
7.6.4	Empirical security \bar{P}_E of J-UNIWARD, J2-UNIWARD, and SI-UNIWARD as a function of γ best bursts from BURSTbaseH. JPEG quality factor 75, left column 0.2 bpnzac, right column 0.4 bpnzac.	70

7.6.5	Empirical security \overline{P}_E of J-UNIWARD, J2-UNIWARD, and SI-UNIWARD as a function of JPEG quality factor Q for $\gamma = 0.1$ best bursts from BURSTbaseH. Left column 0.2 bpnzac, right column 0.4 bpnzac.	70
7.6.6	Empirical security \overline{P}_E of UED-JC, UED2-JC, and SI-UED-JC as a function of γ best bursts from BURSTbaseH for two JPEG quality factors and two payloads. . .	71
7.6.7	Modulation factor versus average quantization step \overline{q} (real acquisitions).	72
7.7.1	Optimal modulation factor $m_{ij}(q, R)$ as a function of the quantization step q for relative payload $R = 0.4$ determined by minimizing the Bhattacharyya distance between cover and stego distributions on generalized Gaussian models of DCT coefficients. Left: low frequency DCT modes (i, j) , $3 \leq i + j \leq 4$ (second and third minor diagonal), Middle: medium frequency DCT modes (i, j) , $5 \leq i + j \leq 10$, Right: high frequency DCT modes (i, j) , $11 \leq i + j \leq 16$	72
8.2.1	First row: spatial 8×8 blocks. The second and third rows are samples where, for the purpose of comparison, the signal S is sampled directly in the DCT domain by sampling a 64-dimensional multivariate Gaussian distribution while S^s is sampled in the spatial domain and then converted to DCT coefficients.	76
8.4.1	Comparison between distributions of shot noise coming from different sensors. Histograms are computed from the photo-site values of one given channel (for color sensors) on a uniform patch. Dash lines represent Gaussian distributions with the same mean and variance as the histogram.	78
8.5.1	Detection error P_E for the pure, simulated noise, and SI-UNIWARD (Approaches 1, 2, and 6) for a switch from ISO 320 to ISO 1000 on MonoBase as a function of the JPEG quality factor. Approaches 3–5 exhibit security that is approximately equal to that of Approach 2 (simulated noise), see Table 8.2.	81
8.5.2	Experiment with a synthetic RAW image: co-occurrences of pixel pairs of adjacent pixels belonging either to adjacent blocks (a), to the same block (b), to adjacent blocks (c) or same block (d) after simulating noise that preserves dependencies only at the block-wise level.	83

List of algorithms

3.1	Pseudo-code for $\delta_{uSA}^{1/2}$ selection-channel aware JPEG features.	24
7.1	Pseudo-code for side-informed embedding with two JPEGs.	63

Preface

In Chapter 1, we introduce the notation and explain the basic concepts, tools, and algorithms this dissertation builds on. Most importantly, we define the side-information and propose to distinguish between private and public types of side-information.

Chapters 2 – 4 focus on the use of public side-information (available to anyone) for detection of content-adaptive steganography. First, a heuristic modification of the SRM feature set [39] is proposed in Chapter 2. In Chapter 3, the approach is generalized for use in JPEG domain, and finally, in Chapter 4 the insight from Chapter 3 is used to improve the feature set proposed in Chapter 2.

Chapters 5 – 6 propose ways to improve the design of steganography with private side-information (available only to the sender). Chapter 5 improves and generalizes a popular heuristic approach for JPEG compressed images when the uncompressed original is available. In Chapter 6, we propose a model-based solution for the same scenario and study the differences against the heuristic approach. Next, Chapter 7 introduces an algorithm for a different situation, when multiple JPEG compressed exposures of the same scene are available instead. Finally, Chapter 6 proposes a practical embedding scheme that embeds the secret information in a way to pass the image as if taken with higher ISO sensitivity.

The dissertation is concluded in Chapter 9.

All algorithms proposed and studied in Chapters 2 – 8 are contributions of this dissertation and have been published and peer-reviewed. Only Chapter 1 describes algorithms published by other authors.

Chapter 1

Introduction and preliminaries

This chapter introduces the notation used through the whole dissertation and explains the mathematical principles of steganography and steganalysis. Steganography is the art of sending secret messages not by obfuscation but by hiding the very existence of the communication. On the other side of the same coin, steganalysis scrutinizes communication channels and tries to prove or disprove the presence of steganography. Both arts have a rich history reaching into the ancient past. This dissertation does not cover the history or discuss the ethics of practicing steganography or steganalysis; an interested reader can find out more in many of the previously published works [34].

1.1 The steganographic channel

Steganography, exploits an already established communication channel between Alice (the sender) and Bob (the recipient). The channel is observed by a third party Eve (also called the Warden). A passive Warden is assumed in this work, meaning the Eve is allowed to monitor the channel and analyze any communication sent through it, but can not introduce any modifications to the data transmitted.

When no steganography is used, the messages sent through the channel are called *cover objects* or *covers*. Depending on the steganographic algorithm and a privately exchanged *key*, Alice can then modify, synthesize or select covers to create *stego objects* that carry her desired secret message. In this dissertation, steganography by *cover modification* is assumed. Steganography can be secure only when the cover objects are non-deterministic. While most techniques discussed in this dissertation can work (either straight or with some adjustments) with any non-deterministic data, from now on, unless stated otherwise, cover objects are grayscale uncompressed or JPEG images.

A steganographic system is comprised of a *cover source*, a *message source* and *embedding* and *extracting* functions. The cover source $S_c = \{\mathcal{C}, P_c\}$ is defined by the set of all possible cover images $\mathbf{X} \in \mathcal{C}$ and their distribution P_c . The message source $S_m = \{\mathcal{M}, P_m(S_c)\}$ is similarly defined by the set of all possible messages \mathbf{m} and their distribution. Note that the distribution P_m is a function of the cover source. Given a cover source and sometimes even a specific cover image, only some messages can be communicated. The embedding function $\text{Emb} : \mathcal{C} \times \mathcal{M} \times \mathcal{K} \rightarrow \mathcal{C}$ allows for creating of stego images $\mathbf{Y} = \text{Emb}(\mathbf{X}, \mathbf{k}, \mathbf{m})$, and its inverse, the extraction function $\text{Ext} : \mathcal{C} \times \mathcal{K} \rightarrow \mathcal{M}$, can extract the secret message \mathbf{m} if provided with the correct stego key \mathbf{k} out of the set of all possible stego keys \mathcal{K} . The stego key has to be exchanged before any communication can be done. The overview of all of the elements of the steganographic channel with a passive Warden can be seen in Figure 1.1.1.

To allow for large scale experiments, randomly generated messages with independent and uncorrelated bits are assumed. This assumption is quite reasonable as the original message can be

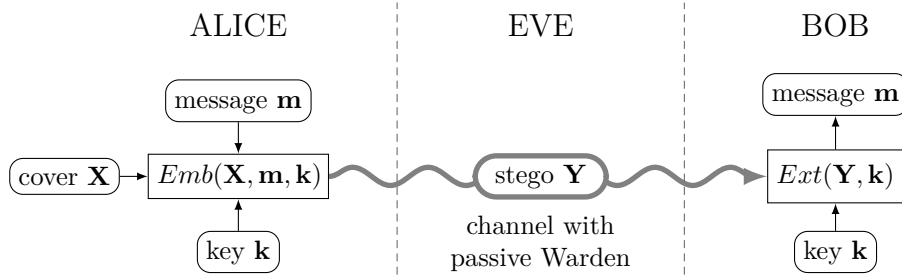


Figure 1.1.1: Components of the steganographic channel.

compressed or encrypted. Only the length of the message is important. In the spatial domain (e.g., bitmap images) we work with the *relative payload* α measured in bits per pixel (bpp). For historical reasons, in the DCT domain (JPEG images) the relative payload is measured in bits per non-zero DCT coefficient (bpnzac). This leads to an interesting quirk of the experiments with JPEG steganography in contrast to spatial domain, the absolute length of the payload is cover dependent.

1.2 Steganographic security and steganalysis

When developing new steganographic and steganalytic algorithms, it is essential to be able to measure their performance, so we can fairly compare them with each other and with the state-of-the-art. There are two main approaches to this problem; a theoretical one for the situation when the cover source and the embedding function can be described analytically, and a practical one for the cases when they can not.

1.2.1 Probabilistic model

The cover images are realizations of a random variable defined by the cover source, $\mathbf{X} \sim P_c$. Stego images, being modifications of cover images, follow a different distribution, $\mathbf{Y} \sim P_s$. While different, they should share the same support \mathcal{C} . The steganographic scheme is fully described by the embedding function and its steganographic security is defined as the statistical indistinguishability of the two distributions. The most popular measure was proposed by Cachin [11], the Kullback–Leibler divergence (KL divergence)

$$D_{\text{KL}}(P_c||P_s) = \sum_{\mathbf{X} \in \mathcal{C}} P_c(\mathbf{X}) \log \frac{P_c(\mathbf{X})}{P_s(\mathbf{X})}. \quad (1.2.1)$$

Using this metric, a steganographic scheme is called ϵ -secure if $D_{\text{KL}}(P_c||P_s) \leq \epsilon$ and perfectly secure if $D_{\text{KL}}(P_c||P_s) = 0$. Other statistical metrics and distances can be used, for example the Bhattacharyya distance.

If the exact cover object distribution or at least a good enough model is available to Alice, she can optimize her embedding function to minimize the KL divergence between the cover and stego object distributions.

Steganalysis can be formulated as a hypothesis testing problem with

$$H_0 : \mathbf{X} \sim P_c, \quad (1.2.2)$$

$$H_1 : \mathbf{X} \sim P_s. \quad (1.2.3)$$

The detector is then a binary mapping function fully defined by its critical region. The design of the best possible detector can be formulated as an optimization problem, Eve is trying to minimize the

detection error. It can be shown that under the Neyman–Pearson setting the optimal detector is the likelihood-ratio test. The problem with this approach is not only the need for accurate estimates of the cover and stego image distributions but also the computational infeasibility due to the large dimensionality of the cover space.

1.2.2 Machine learning approach

While heuristic, empirical assesmen of security using machine learning is much more common. Eve uses a trained classifier to distinguish between cover and stego images and measures its performance on a database that represents the cover source. This approach comes with its own set of problems. As the measure of success is a performance on a sizeable database, the whole process can still be computationally costly. Typically, the content in images is much stronger than the stego signal, so Eve has to use heuristically designed features, and the theoretical bounds on the performance will remain unknown. Also, the database has to accurately represent the cover source or an error called cover source mismatch will be introduced. This leads to cat and mouse games between Alice and Eve, overspecialization to a specific data sources. Few published papers tried to deepen the understanding of this issue.

1.2.2.1 Databases

To fairly compare the performance of different steganographic and steganalytic algorithms, a standardized database is neccessary. Since its introduction during the HUGO BOSS competition in 2010, the most prominent standardized database is BOSSBase, specifically the 1.01 version [7, 29]. It contains 10,000 color RAW images taken by seven different cameras that are developed in `dcraw`, resized and cropped into 512×512 grayscale 8-bit images. Unless stated otherwise, this database is used for all experiments presented in this dissertation.

Other databases used through out this work will be introduced when needed.

1.2.2.2 Rich models

The family of rich models are state-of-the-art features to describe the statistical properties of the noise present in cover and stego images. The most well-known member of this family is the SRM [39] feature set. The main idea behind of rich feature sets is to concatenate a large number of weak features from many submodels and let the classifier find relevant dependencies between them.

Each submodel fixes a denoising filter that tries to predict a pixel value from its neighborhood without using the central pixel's value. This way the potential stego signal in the pixel will not affect the prediction. The prediction is then subtracted from the pixel to get a so-called noise residual. High-quality denoising is not a priority, but the diversity of different denoising kernels is. Therefore, some of the SRM's denoising filters are simple edge detectors and some are more sophisticated. To later save on dimensionality and improve the stability of expected values, the noise residual values are quantized and truncated. Then, a four-dimensional co-occurrence matrix is constructed. Each bin records the frequency of specific four values occurring in sequence through the quantized noise residual. Finally, symmetries of natural images are exploited to again save on dimensionality and to improve robustness.

Many symmetrized co-occurrences obtained from different linear and non-linear submodels are concatenated to a final vector of features. Non-linear filters in the form of minimums and maximums of residuals of rotated linear filters are also employed..

This approach was pioneered by the SPAM [74] feature set for steganalysis in spatial (bitmap) domain. The SPAM set contains 686 features. It was later expanded into the SRM set by adding more linear submodels and the non-linear min-max submodels to a total of 34,671 features. The

PSRM [46, 47] improves on the performance of SRM by representing the noise residuals by collecting histograms of the noise residuals projected on random vectors instead of co-occurrences. The gain in security is however counterweighted by a significant increase in computational complexity.

Today, there exist numerous steganalysis features that are suitable for detection of JPEG steganography. Old embedding schemes, such as F5 [92], model-based steganography [78], Jsteg [90], Out-Guess [76], and Steghide [44], are best detected using statistics formed from quantized DCT coefficients, such as the JPEG Rich Model (JRM) [63]. Unfortunately, JRM is far less effective for detecting modern JPEG steganography, examples of which are UED [42, 43] and J-UNIWARD [51], which adapt their embedding changes to cover content. Such schemes are best detected with features assembled as histograms of noise residuals split by their JPEG phase (location w.r.t. the 8×8 pixel grid): DCT Residuals (DCTR) [49], PHase Aware Rich Model (PHARM) [50], and Gabor Filter Residuals (GFR) [85]. The splitting by phase is effective because the impact of the stego signal on pixels in a decompressed JPEG image depends on the JPEG phase.

1.2.2.3 SRM

Both the SRM and the PSRM extract the same set of noise residuals from the image under investigation. They differ in how they represent their statistical properties. The SRM uses four dimensional co-occurrences while the PSRM uses histograms of residual projections.

A noise residual is an estimate of the image noise component obtained by subtracting from each pixel its estimate (expectation) obtained using a pixel predictor from the pixel's immediate neighborhood. Both rich models use 45 different pixel predictors of two different types – linear and non-linear. Each linear predictor is a shift-invariant finite-impulse response filter described by a kernel matrix $\mathbf{K}^{(\text{pred})}$. The noise residual $\mathbf{Z} = (z_{ij})$ is a matrix of the same dimension as \mathbf{X} :

$$\mathbf{Z} = \mathbf{K}^{(\text{pred})} \star \mathbf{X} - \mathbf{X} \triangleq \mathbf{K} \star \mathbf{X}. \quad (1.2.4)$$

In (1.2.4), the symbol ' \star ' denotes the convolution with \mathbf{X} mirror-padded so that $\mathbf{K} \star \mathbf{X}$ has the same dimension as \mathbf{X} . This corresponds to the 'conv2' Matlab command with the parameter 'same'.

An example of a simple linear residual is $z_{ij} = x_{i,j+1} - x_{ij}$, which is the difference between a pair of horizontally neighboring pixels. In this case, the residual kernel is $\mathbf{K} = \begin{pmatrix} -1 & 1 \end{pmatrix}$, which means that the predictor estimates the pixel value as its horizontally adjacent pixel. This predictor is used in submodel 'spam14h' in the SRM.

All non-linear predictors in the SRM are obtained by taking the minimum or maximum of up to five residuals obtained using linear predictors. For example, one can predict pixel x_{ij} from its horizontal or vertical neighbors, obtaining thus one horizontal and one vertical residual $\mathbf{Z}^{(\text{h})} = (z_{ij}^{(\text{h})})$, $\mathbf{Z}^{(\text{v})} = (z_{ij}^{(\text{v})})$:

$$z_{ij}^{(\text{h})} = x_{i,j+1} - x_{ij}, \quad (1.2.5)$$

$$z_{ij}^{(\text{v})} = x_{i+1,j} - x_{ij}. \quad (1.2.6)$$

Using these two residuals, one can compute two non-linear 'minmax' residuals as:

$$z_{ij}^{(\text{min})} = \min\{z_{ij}^{(\text{h})}, z_{ij}^{(\text{v})}\}, \quad (1.2.7)$$

$$z_{ij}^{(\text{max})} = \max\{z_{ij}^{(\text{h})}, z_{ij}^{(\text{v})}\}. \quad (1.2.8)$$

The next step in forming the SRM involves quantizing \mathbf{Z} with a quantizer $Q_{-T,T}$ with centroids $Q_{-T,T} = \{-Tq, (-T+1)q, \dots, Tq\}$, where $T > 0$ is an integer threshold and $q > 0$ is a quantization step:

$$r_{ij} \triangleq Q_{-T,T}(z_{ij}), \forall i, j. \quad (1.2.9)$$

The next step in forming the SRM feature vector involves computing a co-occurrence matrix of fourth order, $\mathbf{C}^{(\text{SRM})} \in \mathcal{Q}_{-T,T}^4$, from four (horizontally and vertically) neighboring values of the quantized residual r_{ij} (1.2.9) from the entire image:¹

$$c_{d_0 d_1 d_2 d_3}^{(\text{SRM})} = \sum_{i,j=1}^{n_1, n_2-3} [r_{i,j+k} = d_k, \forall k = 0, \dots, 3], \quad (1.2.10)$$

$$d_k \in \mathcal{Q}_{-T,T}, \quad (1.2.11)$$

where $[P]$ is the Iverson bracket, which is equal to 1 when the statement P is true and to 0 when it is false. Note that the dimensionality of the co-occurrence is $|\mathcal{Q}_{-T,T}|^4 = 5^4 = 625$. To keep the co-occurrence bins well-populated and thus statistically significant, the authors of the SRM used $T = 2$ and $q \in \{1, 1.5, 2\}$. Finally, symmetries of natural images are leveraged to further marginalize the co-occurrence matrix to decrease the feature dimension and better populate the SRM feature vector (see Section II.C of [39]). For example, the 625 bins get reduced to 169 bins after symmetrization, while two 625-dimensional co-occurrences of min and max residuals can be symmetrized to 330.

The total dimension of the SRM with three quantization steps is 34,671. A smaller version of the SRM with a single quantization step $q = x \in \{1, 1.5, 2\}$ will be denoted as SRM qx , and it consists of 12,753 features.

1.2.2.4 PSRM

The predictors and residuals used in the PSRM are the same as those used in the SRM. Unlike the SRM, which captures the statistical properties of residuals using four-dimensional co-occurrences, the PSRM uses the first-order statistics (histograms) of projections of residuals onto multiple random directions. Given a noise residual \mathbf{Z} , a slightly simplified algorithm for computing the PSRM is:

1. Generate ν random matrices $\mathbf{\Pi}^{(k)} \in \mathbb{R}^{r \times s}$, $k \in \{1, \dots, \nu\}$.
 - r, s are uniformly randomly selected from $\{1, \dots, s_{max}\}$, where $s_{max} > 0$ is an integer parameter,
 - the elements of $\mathbf{\Pi}^{(k)}$ are independent realizations of a standard normal random variable $\mathcal{N}(0, 1)$,
 - the elements are normalized so that the Frobenius norm² $\|\mathbf{\Pi}^{(k)}\|_F = 1$.
2. For each $k \in \{1, \dots, \nu\}$, compute the residual projections $\mathbf{R}^{(k)} \triangleq \mathbf{Z} * \mathbf{\Pi}^{(k)}$.
3. For linear residuals, quantize $|r_{ij}^{(k)}|/q$ with a quantizer Q_T with $T + 1$ centroids $\mathcal{Q}_T = \{1/2, 3/2, \dots, T + 1/2\}$:

$$\tilde{r}_{ij}^{(k)} = Q_T(|r_{ij}^{(k)}|/q). \quad (1.2.12)$$

For non-linear residuals, quantize $r_{ij}^{(k)}/q$ with a quantizer $Q'_{-T,T}$ with $2T+2$ centroids $\mathcal{Q}'_{-T,T} = \{-T - 1/2, -T + 1/2, \dots, T + 1/2\}$:

$$\tilde{r}_{ij}^{(k)} = Q'_{-T,T}(r_{ij}^{(k)}/q). \quad (1.2.13)$$

¹This is an example of a horizontal co-occurrence.

²The Frobenius norm of matrix \mathbf{A} is defined as $\|\mathbf{A}\|_F = \sqrt{\text{trace}(\mathbf{A}^T \mathbf{A})}$.

4. Compute ν separate histograms of the quantized values:

$$\begin{aligned} h_m^{(k)} &= \left| \{(i, j) \mid \tilde{r}_{ij}^{(k)} = m + 1/2\} \right|, \\ m &\in \{0, 1, \dots, T - 1\}, \\ k &\in \{1, \dots, \nu\} \text{ for linear residuals,} \end{aligned} \quad (1.2.14)$$

$$\begin{aligned} h_m^{(k)} &= \left| \{(i, j) \mid \tilde{r}_{ij}^{(k)} = m + 1/2\} \right|, \\ m &\in \{-T, \dots, T - 1\}, \\ k &\in \{1, \dots, \nu\} \text{ for non-linear residuals.} \end{aligned} \quad (1.2.15)$$

Symmetries of natural images are also used to make the histograms better populated. Depending on the residual and the projection matrix $\mathbf{\Pi}^{(k)}$, the PSRM utilizes up to eight symmetries (rotation by multiples of 90 degrees, mirroring, etc.) for each random matrix $\mathbf{\Pi}^{(k)}$.

The standard parameter setup for the PSRM is as follows. The number of projections per residual is $\nu = 55$, the maximum projection matrix size $s_{max} = 8$, the quantization step $q = 1$, and the histogram threshold $T = 3$. This setup gives the PSRM the dimensionality of 12,870, which is similar to that of SRMqx.

1.2.2.5 Phase-aware JPEG feature sets

The DCTR [49], PHARM [50], and GFR [85] features are formed from noise residuals computed by convolving the decompressed (non-rounded) JPEG image \mathbf{X} (1.5.2) with kernel $\mathbf{G} \in \mathbb{R}^{k_1 \times k_2}$,

$$\mathbf{R}(\mathbf{X}, \mathbf{G}) = \mathbf{X} \star \mathbf{G}. \quad (1.2.16)$$

We note that because the convolution uses no padding (implemented with 'valid' in Matlab), $\mathbf{R} \in \mathbb{R}^{n'_1 \times n'_2}$ with $n'_1 = n_1 - k_1 + 1$ and $n'_2 = n_2 - k_2 + 1$. Next, the residual is quantized,

$$\mathbf{R}(\mathbf{X}, \mathbf{G}, Q) = Q_{\mathcal{Q}}(\mathbf{R}(\mathbf{X}, \mathbf{G})/q), \quad (1.2.17)$$

where $Q_{\mathcal{Q}}$ is a quantizer with centroids $\mathcal{Q} = \{0, 1, 2, \dots, T\}$, q is a fixed quantization step and T a truncation threshold. Each residual is used to compute the following 64 histograms, $0 \leq m \leq T$, $0 \leq i, j \leq 7$:

$$h_m^{(i,j)}(\mathbf{X}, \mathbf{G}, Q) = \sum_{a=1}^{\lfloor n'_1/8 \rfloor} \sum_{b=1}^{\lfloor n'_2/8 \rfloor} [|r_{ij}^{(a,b)}(\mathbf{X}, \mathbf{G}, Q)| = m]. \quad (1.2.18)$$

All $T + 1$ values, $h_0^{(i,j)}, \dots, h_T^{(i,j)}$, from each histogram are concatenated into a vector of $64 \times (T + 1)$ values and these vectors are then concatenated for kernels \mathbf{B} from some filter bank \mathcal{B} . To reduce the feature dimensionality, $64 \times (T + 1) \times |\mathcal{B}|$, and make the bins better populated, certain bins in the concatenated histograms are merged based on symmetries of \mathbf{G} and DCT bases. The DCTR feature set uses a filter bank with $|\mathcal{B}_{DCTR}| = 64$ kernels \mathbf{G} corresponding to 64 DCT bases $\mathbf{F}^{(k,l)}$, $0 \leq k, l \leq 7$. In PHARM, the kernels are obtained by convolving nine small-support pixel predictors with 100 random projection kernels (a total of $|\mathcal{B}_{PHARM}| = 900$ kernels), while in GFR $|\mathcal{B}_{GFR}| = 256$ Gabor filters (four support sizes, two Gabor phases, and 32 orientations) are employed. We note that the size, $k_1 \times k_2$, of the kernels in all three feature sets satisfies $1 \leq k_1, k_2 \leq 15$, which means that no kernel intersects more than four 8×8 pixel blocks.

1.2.2.6 Classifiers

The extracted features still need to be classified and the final performance measured for a fair comparison of different steganographic and steganalytical algorithms. While there are many choices,

the most popular is the total probability of error under equal priors

$$P_E = \min_{P_{FA}} \frac{1}{2} (P_{FA} + P_{MD}(P_{FA})), \quad (1.2.19)$$

where P_{FA} is the probability of false alarm and P_{MD} the probability of missed detection.

Progressively, the dimensionality of the feature sets grew to the point when state-of-the-art nonlinear classifiers like SVM would not train in reasonable time. The ensemble classifier [64], a random forest classifier optimized for high-dimensional input, became the classifier of choice as it is computationally cheap and in combination with the SRM features achieved the winning performance at the HUGO BOSS competition [7]. This classifier also offers an alternative detection performance measure, E_{OOB} , the out-of-bag estimate of the testing error. This value is used internally while training and it can be shown that this value is an unbiased estimate of P_E . To save on computational time, E_{OOB} was used in many published papers and also during earlier experiments in this dissertation. The use of this estimate was slowly abandoned in favor of the testing error averaged over 10 different random splits of the database into equally sized training and testing parts to be more inline with the practices in other machine learning fields.

Cogranne et al. [14] have shown that the ensemble classifier is close to a linear classifier and proposed a low-complexity linear classifier as an alternative. In some experiments in this dissertation, this linear classifier is used instead of the slower ensemble classifier.

1.3 Content-adaptive steganography

The most straightforward way to embed a secret message in a grayscale image while introducing only small changes is to replace the least significant bits of every pixel with bits of the secret message. This algorithm is called the least-significant bit replacement, or LSBR for short. It modifies each pixel by at most ± 1 , a change imperceptible to the human eye. However, given the naivete of the method, modifying even a small portion of pixels in this way introduces unnatural distortion to the histogram of the pixel values and ensures reliable detection [24, 25, 23, 54, 55, 58, 57, 68, 56, 59]. More advanced method, the least-significant bit matching, or LSBM, randomly adds or subtracts one to or from each pixel that needs modification. LSBR changes pixels' mean while LSBM preserves mean but increases variance. LSBM serves as a baseline, the simplest non-flawed image steganography.

Content-adaptive embedding schemes for digital images change individual pixels with probabilities determined from the local pixel neighborhood in order to execute the embedding changes primarily in regions where they are less detectable, such as textures and noisy areas where Eve experiences a large modeling error. The first adaptive methods described in the literature were designed for palette images [36]. One could also argue that schemes that hide message bits in non-zero DCT coefficients are naturally content adaptive since the changes strongly correlate with complex content. In 2010, the Edge Adaptive scheme was designed to hide data in pixel pairs with large differences [69]. The real boom of adaptive schemes started with the introduction of the concept of *embedding distortion* $D : \mathcal{C} \rightarrow \mathbb{R}$. While tools for working with arbitrary distortion function exist, we will limit ourselves to additive distortion functions

$$D(\mathbf{X}, \mathbf{Y}) = \sum_{i,j=1}^{n_1, n_2} \rho_{ij}(\mathbf{X}, y_{ij}), \quad (1.3.1)$$

where ρ_{ij} are non-negative and bounded functions determining the *costs* of replacing the cover pixel or DCT coefficient x_{ij} with y_{ij} . The process of embedding is then a optimization problem, Alice wants to minimize the total distortion under the constraint of communicating the desired payload. The optimal solution to this problem is known and prescribes the probabilities of change β_{ij} for each cover element (also called the *selection channel*) such that

$$\sum_{i,j=1}^{n_1, n_2} H(\beta_{ij}) = m, \quad (1.3.2)$$

where m is the prescribed message length (absolute payload), and $H = H(x)$ is the entropy function ($H(x) = -x \log_2(x) - (1-x) \log_2(1-x)$ for binary schemes and $H(x) = -2x \log_2(x) - (1-2x) \log_2(1-2x)$ for ternary):

$$\beta_{ij} = \frac{\exp(-\lambda \rho_{ij})}{1 + \exp(-\lambda \rho_{ij})} \quad (1.3.3)$$

if the embedding operation is constrained to be binary, and

$$\beta_{ij} = \frac{\exp(-\lambda \rho_{ij})}{1 + 2 \exp(-\lambda \rho_{ij})} \quad (1.3.4)$$

for a ternary scheme with equal costs of changing x_{ij} to $x_{ij} \pm 1$. The parameter λ is a Lagrange multiplier ensuring the constraint (1.3.2).

A practical embedding is then done using a coding scheme based on syndrome-trellis codes [28], that samples stego elements carrying a specific message while minimizing the total distortion near optimally. Examples of such content-adaptive steganographic schemes include HUGO [75], WOW [45], and the UNIWARD family [51].

Such schemes are currently best detected using machine learning, such as binary classifiers trained on examples of cover and stego images represented with higher-order statistics of noise residuals (the so-called rich media models). We wish to emphasize that the previous sentence also applies to modern steganographic schemes that hide messages in quantized DCT coefficients from a JPEG file, UED (Uniform Embedding Distortion) [42, 43] and J-UNIWARD [51]. The most accurate detection of such JPEG steganography is achieved with features that are computed in the spatial domain [49, 50, 85, 48] rather than from quantized DCT coefficients [63].

The cost functions of the above mentioned schemes are heuristic. A notable exception is scheme MiPOD [82], in which the embedding probabilities β_{ij} are derived from their impact on the cover multivariate Gaussian model by solving the following equation for each pixel ij :

$$\beta_{ij} I_{ij} = \lambda \ln \frac{1 - 2\beta_{ij}}{\beta_{ij}}, \quad (1.3.5)$$

where $I_{ij} = 2/\hat{\sigma}_{ij}^4$ is the Fisher information with $\hat{\sigma}_{ij}^2$ an estimated variance of the acquisition noise at pixel ij , and λ is a Lagrange multiplier determined by the payload size. To get the costs for practical embedding, the equation (1.3.3) or (1.3.4) can be reversed with arbitrary choice of λ . For the ternary embedding, the MiPOD costs are:

$$\rho_{ij} = \ln(1/\beta_{ij} - 2). \quad (1.3.6)$$

1.4 Side-information

In this work, we will show how steganography and steganalysis can be improved with the use of the so-called side-information. This information is not essential but can help compensate for the lack of a perfect model or exploit the imperfections of the process. We distinguish between two types of side-information:

- Private – information available only to Alice.
- Public – information entirely or at least partially available to both Alice and Eve.

Private side-information is helpful when designing steganography. For example, Alice may have a high-quality representation of the cover image called precover [57] and embed her secret while

processing the precover and/or converting it to a different format. Eve cannot fully recover the information lost during the processing and therefore is at a disadvantage. The first example of this technique is the embedding-while-dithering steganography [36], which embeds secrets when converting a true-color image to a palette format. By far the most common side-informed steganography today hides in JPEG images using non-rounded DCT coefficients [37, 61, 77, 52, 43, 18, 51]. In Chapter 5, we will introduce new steganographic schemes that exploit private side-information.

Public side-information at least in part survives the processing and the transfer through the embedding channel and may betray details about the approach and processes the other party is using. The first and possibly the most critical example of this is the assumption of the Kerckhoffs's principle. In the absolute majority of publications and through out this dissertation, the proposed steganographic algorithms are analyzed with the knowledge of the cover source and the specific algorithm used as the worst case scenario. The limitations of available steganalytic approaches are also a public side-information that led to the design of content-adaptive steganography. Alice exploits the fact that Eve does not have a perfect model of the cover source and can not distinguish between high frequency but deterministic content and random noise. Finally, since the embedding changes are small and the cost functions are generally insensitive towards them, the selection channel can be recovered by Eve and in turn used to improve the detection of content-adaptive schemes as studied in Chapters 2 to 4.

1.5 JPEG compression

JPEG compressed images and their DCT coefficients were mentioned several times in the previous sections. We now look at them in more detail.

For easier technical description, we only consider $n_1 \times n_2$ 8-bit grayscale images with n_1 and n_2 multiples of 8. A JPEG image will be represented with an array of quantized DCT (discrete cosine transform) coefficients of the same dimensions as the pixel representation of the image, $\mathbf{X} \in \{-1023, \dots, 1024\}^{n_1 \times n_2}$. Often, it will be useful to consider a block representation of \mathbf{X} . The (a, b) th 8×8 block of DCT coefficients, $1 \leq a \leq n_1/8$, $1 \leq b \leq n_2/8$, is formed by coefficients x_{kl} with k, l restricted to $1 + 8(a - 1) \leq k \leq 8a$, $1 + 8(b - 1) \leq l \leq 8b$, and will be denoted $\mathbf{X}^{(a,b)}$. The individual elements of $\mathbf{X}^{(a,b)}$ are $x_{kl}^{(a,b)}$, this time with $0 \leq k \leq 7$, $0 \leq l \leq 7$, hoping that no confusion will be created by using the indices k, l for two different purposes – when used in x_{kl} , their range is $1 \leq k \leq n_1, 1 \leq l \leq n_2$ while in a block, as in $x_{kl}^{(a,b)}$, their range is $0, \dots, 7$.

The (k, l) th DCT basis, $0 \leq k, l \leq 7$, is an 8×8 matrix $\mathbf{F}^{(k,l)} = (f_{ij}^{(k,l)})$, $0 \leq i, j \leq 7$:

$$f_{ij}^{(k,l)} = \frac{w_k w_l}{4} \cos \frac{\pi k(2i+1)}{16} \cos \frac{\pi l(2j+1)}{16}, \quad (1.5.1)$$

where $w_0 = 1/\sqrt{2}$ and $w_k = 1$ for $k > 0$. By decompressing the (a, b) th block of DCT coefficients, we obtain a corresponding block of 8×8 pixels $\tilde{x}_{ij}^{(a,b)}$, $0 \leq i, j \leq 7$:

$$\tilde{x}_{ij}^{(a,b)} = \sum_{k,l=0}^7 f_{kl}^{ij} q_{kl} x_{kl}^{(a,b)}, \quad (1.5.2)$$

where q_{kl} are the elements of the JPEG luminance quantization matrix. Note that in (1.5.2), the pixel values are *not rounded*. Putting all blocks into one $n_1 \times n_2$ matrix, the decompressed (non-rounded) image is represented with a matrix $\tilde{\mathbf{X}} \in \mathbb{R}^{n_1 \times n_2}$.

Finally, we note that a, b will be strictly used to index blocks, k, l for DCT coefficients, and i, j for pixels with the same range and conventions applied to both i, j and k, l .

Alternatively, we can reverse the following matrix form for DCT transformation of 8×8 spatial image \mathbf{P} :

$$\text{DCT}(\mathbf{P}) = \mathbf{A}(\mathbf{A}\mathbf{P}^t)^t = \mathbf{A}\mathbf{P}\mathbf{A}^t, \quad (1.5.3)$$

where

$$\mathbf{A} = \begin{bmatrix} \alpha_1 & \alpha_1 & \alpha_1 & \alpha_1 & \alpha_1 & \alpha_1 & \alpha_1 & \alpha_1 \\ \alpha_2 & \alpha_4 & \alpha_5 & \alpha_7 & -\alpha_7 & -\alpha_5 & -\alpha_4 & -\alpha_2 \\ \alpha_3 & \alpha_6 & -\alpha_6 & -\alpha_3 & -\alpha_3 & -\alpha_6 & \alpha_6 & \alpha_3 \\ \alpha_4 & -\alpha_7 & -\alpha_2 & -\alpha_5 & \alpha_5 & \alpha_2 & \alpha_7 & -\alpha_4 \\ \alpha_1 & -\alpha_1 & -\alpha_1 & \alpha_1 & \alpha_1 & -\alpha_1 & -\alpha_1 & \alpha_1 \\ \alpha_5 & -\alpha_2 & \alpha_7 & \alpha_4 & -\alpha_4 & -\alpha_7 & \alpha_2 & -\alpha_5 \\ \alpha_6 & -\alpha_3 & \alpha_3 & -\alpha_6 & -\alpha_6 & \alpha_3 & -\alpha_3 & \alpha_6 \\ \alpha_7 & -\alpha_5 & \alpha_4 & -\alpha_2 & \alpha_2 & -\alpha_4 & \alpha_5 & -\alpha_7 \end{bmatrix}, \quad (1.5.4)$$

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \\ \alpha_7 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \cos(\pi/4) \\ \cos(\pi/16) \\ \cos(\pi/8) \\ \cos(3\pi/16) \\ \cos(5\pi/16) \\ \cos(3\pi/8) \\ \cos(7\pi/16) \end{bmatrix}. \quad (1.5.5)$$

Note that the multiplication by \mathbf{A} and \mathbf{A}^t comes from first transforming the columns and then the rows of matrix \mathbf{A} .

The JPEG compression and its properties are studied in detail in [34].

Chapter 2

Selection-channel aware attacks in spatial domain

A potential weakness of content-adaptive schemes is that the rule that drives the distribution of the embedding change probabilities among individual elements of the cover is, by the Kerckhoffs' principle, also available to the steganalyst, who can use it to improve the detection. The very first attack of this type was described by Böhme at the rump session at the Information Hiding Workshop in 2005 (and officially published in 2014 [12]). This pertains, however, to a rather special case of public-key steganography implemented using LSB replacement and a specific version of wet paper codes. While attacks derived using the theory of statistical hypothesis testing (e.g., [94, 26, 13]) can incorporate the knowledge of embedding probabilities they are generally not as effective against modern adaptive embedding schemes as machine-learning based methods combined with rich statistical descriptors. Modifying the latter approach to consider the knowledge of the selection channel is, however, not easy as witnessed by the futile effort of the BOSS competition participants [7] attacking HUGO. In 2012, it was shown that an (approximate) knowledge of embedding change probabilities can be used to improve the accuracy of the weighted-stego attack on naive content-adaptive LSB replacement [81]. In [86], the authors managed to utilize rather strong artifacts in the selection channel to mount a very accurate attack on S-UNIWARD with an improperly chosen stabilizing constant (also see [51] for more details). Recently, Tang et al. [87] proposed the first general purpose feature set that utilizes the selection channel and is effective against modern content-adaptive steganography methods. Their attack, which we call tSRM (thresholded SRM), computes the residual co-occurrences from only t percent of pixels with the highest embedding change probabilities (lowest pixel costs). The value of t that leads to the best detection depends on the embedded payload size and the steganographic scheme. The authors reported the detection only for the WOW algorithm.

Modern content-adaptive embedding schemes mentioned above are all based on the same principle – the sender specifies the costs of changing individual pixels and then embeds the payload with the minimal total expected cost. The costs are determined by the local content, which means the Warden can estimate them from the stego image. If the Warden knows the payload size or if she can estimate it, she can also estimate the actual embedding change probabilities used by the sender (the selection channel) and hopefully mount an even more powerful informed attack. For an ignorant Warden, who does not know the sender's embedding strategy, the interaction between the Warden and the sender can be formulated as a non-cooperative strategic game with optimal strategy at the Nash equilibrium, which is generally a different strategy than the one that minimizes the KL divergence between cover and stego objects [80, 16]. Other formulations are certainly possible depending on the information available to the Warden. Ultimately, the problem of content-adaptive steganography and selection-channel-aware steganalysis should be resolved within such game-theoretic framework with an accurate statistical model for images and optimal Warden's detector. Due to the high

complexity of empirical objects [8], such as digital images, and the high complexity of solving the ensuing game, it is however unlikely that optimal practical strategies will ever be identified.

In this chapter, we follow the established paradigm of forming joint higher-order statistics of neighboring noise residuals as statistical descriptors. Our approach is reminiscent of the tSRM [87] but incorporates the selection channel in a different way. The four-dimensional co-occurrences are formed from *all* residuals rather than its proper subset, and, instead of populations, each bin holds the sum of maximum values of the four embedding change probabilities at the corresponding residuals. Since this model, which we call maxSRM, uses the statistic from all pixels, we obtain a more accurate detection. Additionally, and in contrast to the tSRM, if the payload size is known or can be estimated no other parameters need to be determined to steganalyze with maxSRM. Furthermore, the detection with maxSRM appears to suffer less when the embedded payload size is unknown.

2.1 Experimental setup

The detectors were trained as binary classifiers implemented using the FLD ensemble [64] with default settings. We evaluate the security using the P_E 1.2.19 measured on the testing set averaged over ten 5000/5000 database splits denoted as \bar{P}_E . The statistical spread is the standard deviation.

We selected three adaptive steganographic techniques that appear to be the state of the art as of running the experiments (May 2014): the Wavelet Obtained Weights (WOW) [45], S-UNIWARD implemented with the stabilizing constant $\sigma = 1$ as described in [51], and its variant that we call S-UNIGARD (described below). All three algorithms follow the paradigm of steganography by minimizing an additive distortion function. Assuming an $n_1 \times n_2$ grayscale cover image $\mathbf{X} = (x_{ij})$, the embedding starts by computing the costs ρ_{ij} of modifying pixel x_{ij} by 1 or by -1 (the costs of both modifications are equal). An optimal embedding scheme hides the secret message while minimizing the total cost of embedding (distortion) $D(\mathbf{X}, \mathbf{Y}) = \sum_{i,j=1}^{n_1, n_2} \rho_{ij} [x_{ij} \neq y_{ij}]$, where $[P]$ is the Iverson bracket $[P] = 1$ when P is true and $[P] = 0$ when P is false, and \mathbf{Y} is the stego image. Such an optimal scheme would modify pixel x_{ij} to $x_{ij} + 1$ with probability β_{ij} (and to $x_{ij} - 1$ with the same probability), where $\beta_{ij} = (1 + e^{\lambda \rho_{ij}})^{-1}$ [28] with $\lambda > 0$ determined from the payload constraint, $\sum_{i,j=1}^{n_1, n_2} H(\beta_{ij}) = Rn$, where $H(x) = -2x \log_2 x - (1 - 2x) \log_2 (1 - 2x)$ is the ternary entropy function in bits. In our tests, we used simulators of the embedding that indeed executed the changes with the probabilities β_{ij} . Practical embedding schemes that embed messages with nearly minimal distortion can be built using syndrome-trellis codes [28].

S-UNIGARD is built in the same way as S-UNIWARD with the three Wavelet Daubechies kernels replaced with Gabor filters (hence the letter 'G' replacing 'W' in the embedding scheme name), which are basically a set of differently oriented sinusoidal patterns modulated by a Gaussian kernel. Each kernel is obtained by sampling the following continuous function in \mathbb{R}^2 parametrized by the wavelength λ , the orientation angle θ , the phase offset ϕ , and the standard deviation σ of the Gaussian modulation:

$$G_{\lambda, \theta, \phi, \sigma, \gamma}(x, y) = \exp\left(-\frac{u^2 + \gamma^2 v^2}{2\sigma^2}\right) \cos\left(2\pi \frac{u}{\lambda} + \phi\right), \quad (2.1.1)$$

$$u = x \cos \theta + y \sin \theta, \quad (2.1.2)$$

$$v = -x \sin \theta + y \cos \theta. \quad (2.1.3)$$

In S-UNIGARD, we use $\lambda = 2$, two offsets $\phi \in \{0, \pi/2\}$, 16 directions $\theta \in \{0, \pi/16, \dots, 15\pi/16\}$, $\gamma = 0.5$, and $\sigma = 1$. The kernels are obtained by sampling $G_{\lambda, \theta, \phi, \sigma, \gamma}(x, y)$ at $x, y \in \{-5, -4, \dots, 4, 5\}$ giving the filters a support of 11×11 pixels (see all 32 Gabor filters in Figure 2.1.1). All kernels are made zero mean (high-pass) by subtracting the kernel mean from all its elements. Assuming the cover is an $n_1 \times n_2$ grayscale image $\mathbf{X} = (x_{ij})$, $1 \leq i \leq n_1$, $1 \leq j \leq n_2$, the cost of changing pixel x_{ij} to $x_{ij} \pm 1$ (obtaining an image $\mathbf{X}_{[i,j]}$) and leaving all other pixels intact is computed in the exact same manner as in S-UNIWARD,



Figure 2.1.1: A set of 32 Gabor filters used in S-UNIGARD.

$$\rho_{ij} \triangleq \sum_{k=1}^{32} \sum_{u=1}^{n_1} \sum_{v=1}^{n_2} \frac{|F_{uv}^{(k)}(\mathbf{X}) - F_{uv}^{(k)}(\mathbf{X}_{[i,j]})|}{s + |F_{uv}^{(k)}(\mathbf{X})|}, \quad (2.1.4)$$

where $F_{uv}^{(k)}(\mathbf{X}) = (\mathbf{X} \star \mathbf{G}^{(k)})_{uv}$ is the uv th elements of the mirror-padded convolution between \mathbf{X} and the k th Gabor filter $\mathbf{G}^{(k)}$, and s is a positive stabilizing constant. This constant affects the selection channel and needs to be chosen carefully to avoid introducing artifacts into the selection channel [86]. We determined it by a grid search on the grid $\mathcal{G} = \{10^{-15}, 10^{-14}, \dots, 10^0\}$ as the value that minimizes the out-of-bag (OOB) detection error on BOSSbase 1.01 [29] when steganalyzing with the 12,753-dimensional SRMQ1 model [39] and the ensemble classifier. The optimum was rather flat around $s \approx 10^{-2}$.

2.2 maxSRM

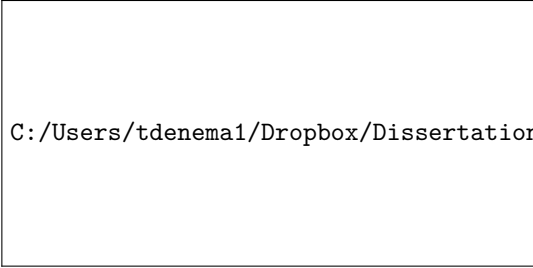
The proposed feature set is a variant of the so-called spatial rich model (SRM) described in [39]. The maxSRM is built in the same manner as the SRM but the process of forming the co-occurrence matrices is modified to consider the embedding change probabilities $\hat{\beta}_{ij}$ estimated from the analyzed image. The SRM consists of multiple co-occurrence matrices formed by four neighboring quantized noise residual samples. Let us assume that $\mathbf{R} = (r_{ij})$ is one such noise residual, for example, one that was obtained by predicting the pixel value x_{ij} as the average of its horizontal neighbors, $r_{ij} = x_{ij} - (x_{i,j-1} + x_{i,j+1})/2$, quantized to $\mathcal{Q} = \{-2, -1, 0, 1, 2\}$. The SRM uses 4D co-occurrences, which are 4D arrays defined as¹

$$c_{d_0 d_1 d_2 d_3} = \sum_{i,j=1}^{n_1, n_2-3} [r_{i,j} = d_k, \forall k = 0, \dots, 3]. \quad (2.2.1)$$

In maxSRM, we modify this definition to

$$\tilde{c}_{d_0 d_1 d_2 d_3} = \sum_{i,j=1}^{n_1, n_2-3} \max_{k=0, \dots, 3} \hat{\beta}_{i,j+k} [r_{i,j} = d_k, \forall k = 0, \dots, 3]. \quad (2.2.2)$$

¹This is an example of a horizontal co-occurrence.



C:/Users/tdenema1/Dropbox/Dissertation/data/coocetypes.png

Figure 2.2.1: Four types of co-occurrence scan direction.

Type	\bar{P}_E
hv	22.14±0.26
dm	22.22±0.33
d2	21.71±0.44
dd	21.66±0.29

Table 2.1: Effect of the co-occurrence scans on the detection error \bar{P}_E . WOW at 0.4 bpp, the 338-dimensional SQUARE submodel of SRM.

In other words, instead of adding a 1 to the corresponding co-occurrence bin, we add the maximum of the embedding change probabilities taken across the four residuals. This way, those groups of four of pixels with small probability of being changed will not affect the co-occurrence values much, while those where at least one pixel is likely to change will. We note that the rest of the process of forming the SRM stays exactly the same, including the symmetrization by sign and direction and merging into SRM submodels (see [39] for details). The proposed maxSRM has thus the same dimensionality as the SRM, which is 34,671.

To further boost the detection, we investigated another design component of the SRM, which is the co-occurrence scan direction. The original SRM uses horizontal and vertical scans (see the case 'hv' in Figure 2.2.1). In this section, we studied three other possibilities shown in the same figure – diagonal and minor-diagonal directions ('dm'), and two 'oblique' directions marked 'd2' and 'dd'. Because the oblique directions do not have a mirror symmetry, they allow collecting twice as much data for the co-occurrences, making them better populated. We observed in our experiments that the oblique directions do provide better detection across all tested algorithms. In Table 2.1, we give a small example of this positive effect with the SQUARE SRM submodel (dimension 338) for the WOW algorithm at 0.4 bpp. The diagonal directions are the worst while the oblique directions are very similar and give (in this case) an improvement of 0.4% in the detection error w.r.t. the 'hv' scan used in SRM. Thus, we decided to include in our tests the version of the maxSRM with all co-occurrence scan directions replaced with the oblique direction 'd2'. We will call this version of the rich model the maxSRMd2.

2.3 Experiments

As our first experiment, we provide the detection results for all three embedding algorithms (see Section 2.1) when steganalyzing with SRM, maxSRM, and maxSRMd2 under the ideal case when the steganalyst knows the embedded payload size. The results, shown in Table 2.2, point out several rather interesting facts. First, while the security of WOW and S-UNIWARD appears almost the same under the SRM, when the selection channel information is utilized, WOW becomes much more detectable (for small payloads by more than 10%). This is most likely because WOW's adaptivity is stronger in the sense that embedding probabilities of S-UNIWARD are more "spread out" (see

		0.05 bpp	0.1 bpp	0.2 bpp	0.3 bpp	0.4 bpp	0.5 bpp
WOW	SRM	.4572 ± .0026	.4026 ± .0028	.3210 ± .0038	.2553 ± .0028	.2060 ± .0022	.1683 ± .0023
	maxSRM	.3595 ± .0017	.3025 ± .0033	.2383 ± .0022	.1943 ± .0015	.1623 ± .0038	.1371 ± .0028
	maxSRMd2	.3539 ± .0024	.2997 ± .0023	.2339 ± .0041	.1886 ± .0036	.1543 ± .0036	.1306 ± .0021
	tSRM	.3765 ± .0035	.3160 ± .0032	.2574 ± .0035	.2143 ± .0027	.1815 ± .0026	.1517 ± .0027
S-UNIWARD	SRM	.4533 ± .0026	.4024 ± .0019	.3199 ± .0027	.2571 ± .0016	.2037 ± .0032	.1640 ± .0024
	maxSRM	.4209 ± .0032	.3684 ± .0033	.2981 ± .0032	.2431 ± .0016	.1992 ± .0022	.1633 ± .0028
	maxSRMd2	.4180 ± .0025	.3660 ± .0040	.2886 ± .0025	.2360 ± .0022	.1908 ± .0025	.1551 ± .0019
	tSRM	.4391 ± .0033	.3935 ± .0013	.3199 ± .0027	.2571 ± .0016	.2037 ± .0032	.1640 ± .0024
S-UNIGARD	SRM	.4667 ± .0020	.4214 ± .0035	.3384 ± .0015	.2774 ± .0024	.2278 ± .0033	.1811 ± .0027
	maxSRM	.4195 ± .0030	.3712 ± .0027	.3002 ± .0022	.2466 ± .0022	.2062 ± .0025	.1702 ± .0027
	maxSRMd2	.4170 ± .0024	.3673 ± .0018	.2957 ± .0024	.2409 ± .0035	.1985 ± .0027	.1647 ± .0028
	tSRM	.4335 ± .0022	.3867 ± .0041	.3205 ± .0045	.2660 ± .0029	.2183 ± .0033	.1782 ± .0025

Table 2.2: Average detection error \bar{P}_E for three embedding algorithms and four steganalysis feature sets.

Figure 2.3.1). Obviously, the difference between SRM and maxSRM will diminish with a decreasing degree of adaptivity of the embedding algorithm. Also notice that while S-UNIGARD appears more secure than S-UNIWARD under SRM, this difference ($\approx 2\%$) becomes negligible when the selection channel is utilized. Finally, the maxSRM is always better than SRM, pointing to the fact that utilizing the selection channel in the proposed manner indeed helps steganalysis. Moreover, the comparison between maxSRM and maxSRMd2 shows that the 'd2' co-occurrence scan is always better than the default 'hv' making the detection error smaller by as much as $\approx 1\%$. Since the dimensionality of both models is the same, there is no reason not to use maxSRMd2 over maxSRM.

Our next experiment was aimed at finding a fixed testing payload, $\hat{\alpha}$, used for computing the embedding probabilities that would provide an overall good performance when the real payload α is unknown. This will necessarily be a trade off between losing the detection for small versus large payloads. Based on tests with all three algorithms, it appears that a reasonable trade off is achieved when the test payload is fixed to a medium value of $\hat{\alpha} = 0.2$ bpp for S-UNIWARD and S-UNIGARD and to $\hat{\alpha} = 0.1$ bpp for WOW. Figures 2.3.2 and 2.3.3 show the detection error P_E and its increase when steganalyzing with a fixed test payload as opposed to the true payload. The performance drop averaged over payloads is below 1% and exhibits similar values and similar trends across the three tested algorithms.

In Figure 2.3.4, we compare the maxSRMd2 with the previously proposed tSRM by showing the improvement in detection error over the SRM under the assumption that the real payload is known ($\hat{\alpha} = \alpha$). The threshold t in tSRM was optimized for each tested payload and stego algorithm. While the maxSRMd2 feature set provides better detection for all three algorithms, the tSRM fails to improve detection of S-UNIWARD for all payloads larger than 0.1 bpp and is only marginally effective against S-UNIGARD for large payloads. The maxSRMd2 consistently outperforms tSRM, sometimes by more than 3%. Moreover, when the embedded payload size is known or can be estimated, the maxSRM can be readily used while applying the tSRM requires running potentially expensive experiments to determine the best threshold for each payload. Also, we found out that setting a fixed value of the threshold t in tSRM when the true payload is not known is much trickier. It appears that one needs to either settle for a smaller improvement over the entire range of payloads or sacrifice the improvement (or even take a penalty) for large payloads (see Figure 5 in [87]). Finally, we wish to point out that the maxSRM is a generalization of tSRM because the tSRM feature vector can be computed using maxSRM by preprocessing the embedding change probabilities, β_{ij} , and setting $\beta_{ij} = 1$ when the cost of pixel i, j is within the top $t\%$ of costs using and setting it to 0 otherwise.



Figure 2.3.1: Embedding probability for payload 0.4 bpp using WOW (top right), S-UNIWARD (bottom left), and S-UNIGARD (bottom right) for a 128×128 grayscale cover image shown in top left (a 128×128 crop of '1013.pgm' from BOSSbase).

2.4 Conclusion

While content-adaptive steganography is nowadays a mature subject, steganalysis that utilizes the probabilistic selection channel is much less developed. Even though detectors built from tractable cover models using the theory of statistical hypothesis testing can incorporate Bayesian priors in a relatively straightforward manner, it is unclear how to adapt the detectors built by training classifiers in heuristically assembled feature spaces. This topic is relevant as such detectors are indispensable for detecting modern content-adaptive steganographic schemes.

In this chapter, we propose a variant of the spatial rich model (the so-called maxSRM) modified to incorporate the knowledge of embedding change probabilities. Even though the proposed approach is heuristic, it does bring quite an improvement over features that do not consider the selection channel and it provides an interesting insight into the design of steganographic schemes. While the WOW and S-UNIWARD algorithms exhibit an essentially identical level of statistical detectability when tested with SRM, WOW is much more detectable with the selection-channel-aware maxSRM than S-UNIWARD. This is attributed to the varying degree of adaptivity of both algorithms. Apparently, WOW's selection channel is "overly adaptive," which makes this algorithm more vulnerable to maxSRM than the other algorithms. Moreover, while S-UNIGARD appears more secure than S-UNIWARD under SRM, this difference ($\approx 2\%$) becomes negligible when the selection channel is utilized. Steganography designers thus need to be aware of how the properties of the selection channel affect statistical detectability when designing future steganographic schemes.

The maxSRM also offers the following three important advantages over the previously proposed thresholded SRM (tSRM): 1) the detection error is always lower, 2) there is no need to determine any parameters when the embedded payload is known or can be estimated, 3) the loss of detection is less severe when the real payload is unknown.

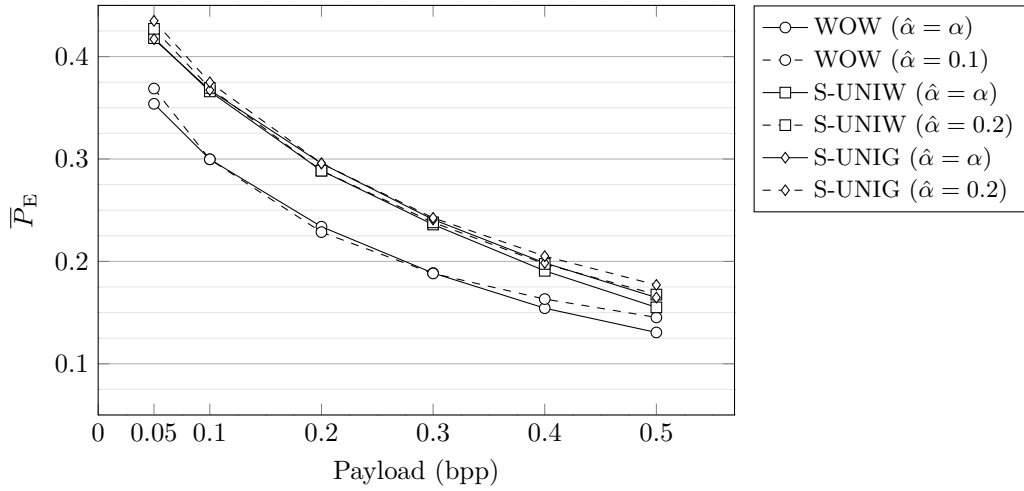


Figure 2.3.2: Detection error for all three algorithms when steganalyzing with maxSRMd2 with a fixed test payload ($\hat{\alpha} = 0.1$ for WOW and $\hat{\alpha} = 0.2$ for S-UNIWARD), versus the test payload set to the real payload, $\hat{\alpha} = \alpha$.

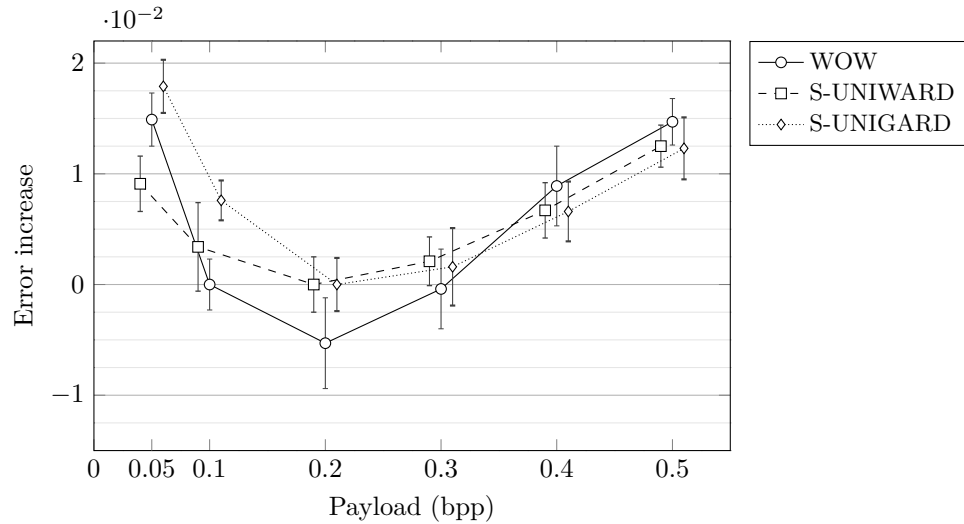


Figure 2.3.3: Detection error increase when steganalyzing with the test payload $\hat{\alpha}$ chosen as in the text and the true payload α . The payload (x coordinates) were shifted by a small amount to prevent the markers and error bars from overlapping.

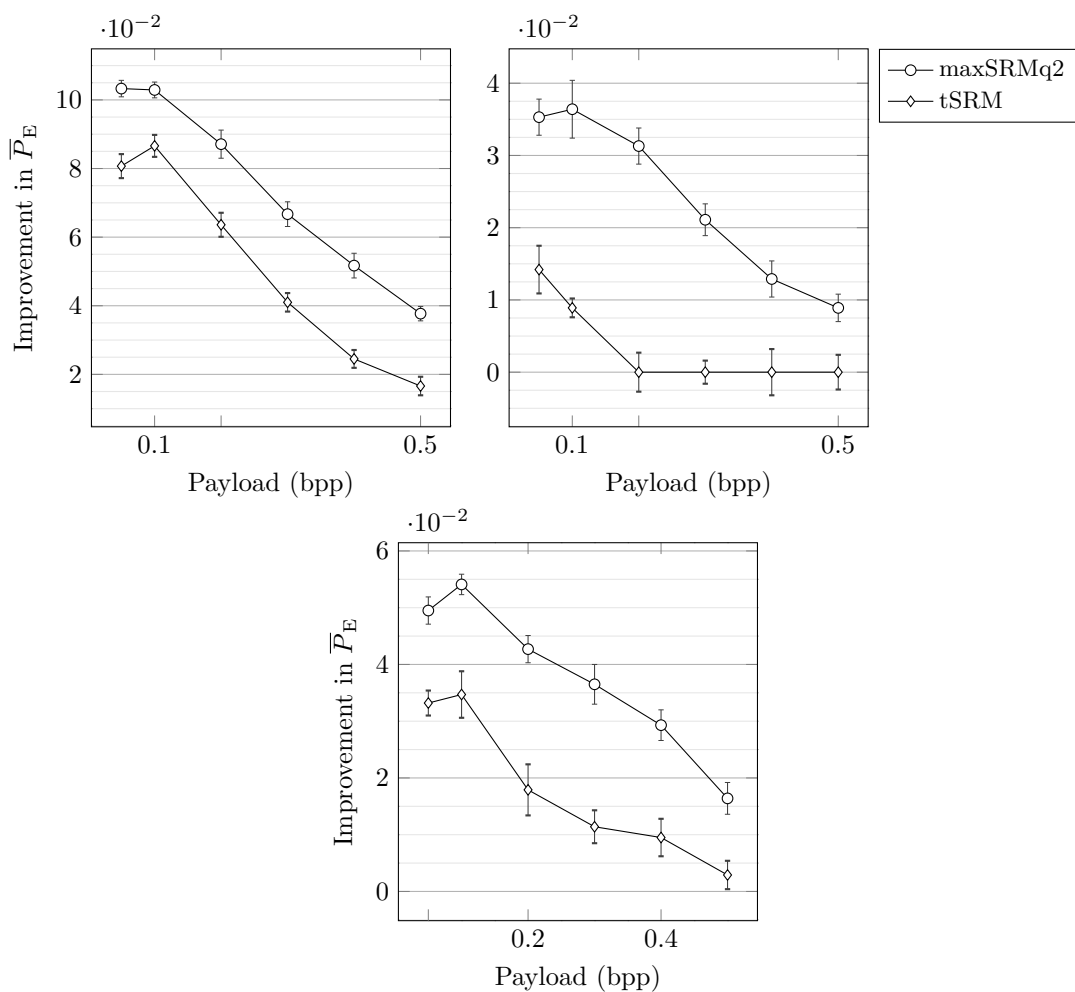


Figure 2.3.4: Improvement in detection error when steganalyzing with SRM versus maxSRMd2 or tSRM. The threshold in tSRM was optimized for each payload. From top down and from left right: WOW, S-UNIWARD, and S-UNIGARD.

Chapter 3

Selection-channel aware attacks in DCT domain

The way the selection channel is incorporated in steganalysis features in the previous chapter cannot be used for detection of JPEG steganography because the embedding and the steganalysis domains are different. In particular, the embedding changes applied to an 8×8 block of quantized DCT coefficients affect all 64 pixels and the modifications are no longer limited to ± 1 changes but can have a much larger amplitude depending also on the JPEG quality factor. Pixel change rate thus no longer properly characterizes the distortion at a pixel. On the other hand, knowing the embedding change probabilities of quantized DCT coefficients it is possible to compute the expected value of the distortion at each pixel. In this chapter, we show that by accumulating such a quantity in histograms of JPEG-phase-aware noise residuals [49, 50, 85], it is possible to construct spatial rich features that provide more accurate detection of current content-adaptive JPEG algorithms. The improvement appears to be the largest for small payloads and diminishes for large payloads when the embedding algorithm loses its content adaptivity.

3.1 Residual distortion measure

To incorporate the selection channel into the feature design, we inspired ourselves with the selection-channel-aware versions of the SRM [39] called maxSRM [21], where the co-occurrences of noise residuals accumulated the embedding change probabilities. Porting this concept directly to the features from the previous chapter for steganalysis of JPEG images is, however, not possible because the embedding changes are executed in the DCT domain. The embedding modifies the pixel values in the decompressed JPEG image by a wide range of values rather than by ± 1 . Instead of the embedding change probability, we propose to use a suitable measure of the residual distortion. To this end, we first derive the properties of the random variable representing the embedding distortion in the residual domain and then investigate several different measures of the distortion with the goal to obtain a quantity that can be evaluated efficiently.

We denote the quantized DCT coefficients in the (a, b) th block of the cover and stego image by $x_{kl}^{(a,b)}$ and $y_{kl}^{(a,b)} = x_{kl}^{(a,b)} + s_{kl}^{(a,b)}$, respectively, where $s_{kl}^{(a,b)}$ are the embedding changes, which are independent realizations of random variables $S_{kl}^{(a,b)}$ attaining the values in $\{-1, 0, 1\}$ with probabilities $\{\beta_{kl}^{(a,b)}, 1 - 2\beta_{kl}^{(a,b)}, \beta_{kl}^{(a,b)}\}$ determined by the steganographic scheme and the payload size. We stress that this model of embedding fits all modern JPEG steganographic algorithms, including both versions of UED and J-UNIWARD. Recalling (1.5.2), the difference between the non-rounded pixel values in the decompressed cover and stego images, $\tilde{x}_{ij}^{(a,b)} = \sum_{k,l=0}^7 f_{kl}^{ij} q_{kl} x_{kl}^{(a,b)}$ and

$\tilde{y}_{ij}^{(a,b)} = \sum_{k,l=0}^7 f_{kl}^{ij} q_{kl} y_{kl}^{(a,b)}$, respectively, is thus:

$$\tilde{s}_{ij}^{(a,b)} = \tilde{y}_{ij}^{(a,b)} - \tilde{x}_{ij}^{(a,b)} = \sum_{k,l=0}^7 f_{kl}^{ij} q_{kl} s_{kl}^{(a,b)}. \quad (3.1.1)$$

Because the embedding changes are mutually independent and because

$$E[S_{kl}^{(a,b)}] = 0, \quad (3.1.2)$$

$$\text{Var}[S_{kl}^{(a,b)}] = 2\beta_{kl}^{(a,b)} \quad (3.1.3)$$

we have

$$E[\tilde{S}_{ij}^{(a,b)}] = 0, \quad (3.1.4)$$

$$\text{Var}[\tilde{S}_{ij}^{(a,b)}] = 2 \sum_{k,l=0}^7 (f_{kl}^{ij})^2 q_{kl}^2 \beta_{kl}^{(a,b)}, \quad (3.1.5)$$

where we remind that, by our convention, $\tilde{S}_{ij}^{(a,b)} = \sum_{k,l=0}^7 f_{kl}^{ij} q_{kl} S_{kl}^{(a,b)}$ is the random variable whose realization is $\tilde{s}_{ij}^{(a,b)}$.

From (1.2.16) and the linearity of convolution, the difference, $\boldsymbol{\rho} \in \mathbb{R}^{n'_1 \times n'_2}$, between the residuals of stego and cover images can thus be expressed as

$$\boldsymbol{\rho}(\mathbf{S}) = \mathbf{R}(\tilde{\mathbf{Y}}, \mathbf{G}) - \mathbf{R}(\tilde{\mathbf{X}}, \mathbf{G}) = \tilde{\mathbf{S}} \star \mathbf{G}. \quad (3.1.6)$$

Technically, $\boldsymbol{\rho}$ also depends on the kernel \mathbf{G} but, in order to declutter the notation, we only explicitly write the dependence on the embedding changes \mathbf{S} as these are the most important. Also, to be specific, we will associate a given residual value with the position of the upper left corner of the kernel \mathbf{G} when performing the convolution.

Since the kernels \mathbf{G} for the features discussed in the previous chapter never intersect more than four different 8×8 pixel blocks, when computing a specific value of the residual using the convolution (1.2.16) the residual value will generally depend on either one 8×8 DCT block, when the kernel is positioned within one JPEG block, two blocks when the kernel straddles two adjacent blocks, or four blocks. Because the inverse DCT is linear and the residual also depends linearly on the non-rounded pixel values, the residual is thus a linear combination of 64, 128, or 256 DCT coefficients. In principle, it is thus possible to analytically compute the coefficients $\alpha_{kl}^{(u,v)}(i, j, \mathbf{G})$, $0 \leq u, v \leq 1$, $0 \leq i, j, k, l \leq 7$, in the linear combination of up to 256 DCT coefficients:

$$\rho_{ij}^{(a,b)}(\mathbf{S}) = \sum_{k,l=0}^7 \sum_{u,v=0}^1 \alpha_{kl}^{(u,v)}(i, j, \mathbf{G}) s_{kl}^{(a+u, b+v)}. \quad (3.1.7)$$

The coefficients $\alpha_{kl}^{(u,v)}(i, j, \mathbf{G})$ depend on the phase (i, j) , the kernel \mathbf{G} , as well as the quantization steps q_{kl} . For example, since the DCTR and GFR features use 8×8 kernels, for phase $(0,0)$ only 64 values of $\alpha_{kl}^{(u,v)}(i, j, \mathbf{G})$ will generally be non-zero. For phases $(0, k)$, $(k, 0)$, $k > 0$, there will be 128 non-zero values, and for the remaining 49 phases there will be 256 non-zero $\alpha_{kl}^{(u,v)}(i, j, \mathbf{G})$.

Because $E[S_{kl}^{(a,b)}] = 0$ for all (a, b) and k, l , we have $E[\rho_{ij}^{(a,b)}(\mathbf{S})] = 0$ as well. Thus, we will take some measure, δ , of the statistical spread of $\rho_{ij}^{(a,b)}(\mathbf{S})$ as a quantity that should be accumulated in the histograms of residuals (1.2.18) in a similar fashion as the embedding change probabilities are accumulated in the selection-channel-aware maxSRM [21], $0 \leq m \leq T$, $0 \leq i, j \leq 7$:

$$\bar{h}_m^{(i,j)}(\tilde{\mathbf{X}}, \mathbf{G}, Q, \boldsymbol{\beta}) = \sum_{a=1}^{\lfloor n'_1/8 \rfloor} \sum_{b=1}^{\lfloor n'_2/8 \rfloor} [|r_{ij}^{(a,b)}(\tilde{\mathbf{X}}, \mathbf{G}, Q)| = m] \cdot \delta(\rho_{ij}^{(a,b)}(\mathbf{S})). \quad (3.1.8)$$

In (3.1.8), $\bar{h}_m^{(i,j)}$ stands for the selection-channel-aware version of the histograms (1.2.18) and $(i, j) \in \{0, \dots, 7\}^2$ is the JPEG phase. Note that since the variance of $\rho_{ij}^{(a,b)}(\mathbf{S})$ depends on $\boldsymbol{\beta}$ (3.1.3), so does $\bar{h}_m^{(i,j)}$.

Two natural choices for δ are the standard deviation and the expectation of the absolute value of $\rho_{ij}^{(a,b)}(\mathbf{S})$ (c.f., (3.1.7)) :

$$\begin{aligned} \delta_{std}(\boldsymbol{\beta})_{ij}^{(a,b)} &= \sqrt{\text{Var}[\rho_{ij}^{(a,b)}(\mathbf{S})]}, \\ &= \sqrt{2 \sum_{k,l=0}^7 \sum_{u,v=0}^1 (\alpha_{kl}^{(u,v)}(i, j, \mathbf{G}))^2 \beta_{kl}^{(a+u, b+v)}}, \end{aligned} \quad (3.1.9)$$

$$\delta_{EA}(\boldsymbol{\beta})_{ij}^{(a,b)} = E[|\rho_{ij}^{(a,b)}(\mathbf{S})|]. \quad (3.1.10)$$

To clarify the above expressions, $\delta_{std}(\boldsymbol{\beta})_{ij}^{(a,b)}$ stands for the ij th element in the (a, b) th block in matrix $\delta_{std}(\boldsymbol{\beta}) \in \mathbb{R}^{n'_1 \times n'_2}$ and the same applies to $\delta_{EA}(\boldsymbol{\beta})$. Note that both $\delta_{std}(\boldsymbol{\beta})$ and $\delta_{EA}(\boldsymbol{\beta})$ depend on the change rates $\boldsymbol{\beta}$ (the selection channel), which is an $n_1 \times n_2$ array of embedding change probabilities arranged in the same fashion as the DCT coefficients, and on the kernel \mathbf{G} .

Neither (3.1.9) or (3.1.10) are, unfortunately, suitable for practical usage. The standard deviation $\delta_{std}(\boldsymbol{\beta})$ can be computed for all (a, b) and a given kernel G using one convolution $\mathbf{A} \star \boldsymbol{\beta}$, where \mathbf{A} is a 16×16 matrix with four 8×8 blocks $\mathbf{A}^{(u,v)} = ((\alpha_{kl}^{(u,v)}(i, j, \mathbf{G}))^2)_{k,l=0}^7$

$$\mathbf{A}(i, j, \mathbf{G}) = \begin{pmatrix} \mathbf{A}^{(0,0)} & \mathbf{A}^{(0,1)} \\ \mathbf{A}^{(1,0)} & \mathbf{A}^{(1,1)} \end{pmatrix}. \quad (3.1.11)$$

However, because there are 64 phases and $|\mathcal{B}|$ kernels, one thus needs to compute $64 \times |\mathcal{B}|$ convolutions, which is rather expensive even for the smallest filter bank of DCTR and completely prohibitive for PHARM with 900 filters.

The problem with $\delta_{EA}(\boldsymbol{\beta})$ is that it cannot be computed analytically and Monte Carlo estimation requires at least 200 simulated embeddings to obtain a value accurate within 10% (determined experimentally for DCTR and J-UNIWARD at 0.4 bpnzac, bits per non-zero AC DCT coefficient). This increases the number of required convolutions by a factor of 200. One possibility is to approximate the sum in $\rho_{ij}^{(a,b)}(\mathbf{S})$ (3.1.7) with a Gaussian random variable $\mathcal{N}(0, \sigma^2)$ for which one can easily verify that $E[|\mathcal{N}(0, \sigma^2)|] = 2\sigma/\sqrt{2\pi}$. Unfortunately, this brings us back to the prohibitive complexity of evaluating the variance. Also, note that with this approximation $\delta_{EA}(\boldsymbol{\beta})$ and $\delta_{std}(\boldsymbol{\beta})$ coincide.

To resolve the complexity issues, we turned our attention to how the JPEG-phase-aware features are formed [49, 50, 85]. They are computed in two steps by first decompressing the JPEG image to the spatial domain and then evaluating merely $|\mathcal{B}|$ convolutions. To substantially decrease the complexity, we will strive to keep a similar two-stage process. The variance, however, cannot be computed this way because after inverse DCT (1.5.2) the pixels' modifications are no longer independent, which would require computing the covariance for each four-block of adjacent 8×8 blocks. This problem is removed when we switch to an upper bound of $|\rho_{ij}^{(a,b)}(\mathbf{S})|$. An upper bound in the form of

$$|\rho_{ij}^{(a,b)}(\mathbf{S})| \leq \sum_{k,l=0}^7 \sum_{u,v=0}^1 |\alpha_{kl}^{(u,v)}(i, j, \mathbf{G})| \cdot |w_{kl}^{(a+u, b+v)}| \quad (3.1.12)$$

is, however, not useful as we still face the same prohibitive computational complexity when evaluating the expectation of the bound. Instead, we will first bound the absolute value of the residual (3.1.6):

$$|\boldsymbol{\rho}(\mathbf{S})| \leq |\tilde{\mathbf{S}}| \star |\mathbf{G}|, \quad (3.1.13)$$

and then further bound

$$|\tilde{s}_{ij}^{(a,b)}| \leq \sum_{k,l=0}^7 |f_{kl}^{ij}| q_{kl} |s_{kl}^{(a,b)}|. \quad (3.1.14)$$

Because $E[|S_{kl}^{(a,b)}|] = 2\beta_{kl}^{(a,b)}$, we have for the expectation

$$E[|\tilde{S}_{ij}^{(a,b)}|] \leq 2 \sum_{k,l=0}^7 |f_{kl}^{ij}| q_{kl} \beta_{kl}^{(a,b)} \triangleq w_{ij}^{(a,b)}(\boldsymbol{\beta}). \quad (3.1.15)$$

Finally, using (3.1.13)–(3.1.15) $\delta_{EA}(\boldsymbol{\beta})$ can be bounded by

$$\delta_{EA}(\boldsymbol{\beta}) = E[|\boldsymbol{\rho}(\mathbf{S})|] \leq \mathbf{W}(\boldsymbol{\beta}) \star |\mathbf{G}| \triangleq \delta_{uSA}(\boldsymbol{\beta}), \quad (3.1.16)$$

which can be efficiently evaluated by first computing $t_{ij}^{(a,b)} = 2 \sum_{k,l=0}^7 |f_{kl}^{ij}| q_{kl} \beta_{kl}^{(a,b)}$ by blocks (this is as computationally demanding as decompressing a JPEG image) and then convolving $\mathbf{W}(\boldsymbol{\beta})$ with the absolute value of the kernel. We used the subscript ‘uSA’ (Upper bounded Sum of Absolute values) for the bounding quantity.

We observed an approximately quadratic dependence between δ_{uSA} and δ_{EA} , $\delta_{uSA} \propto \delta_{EA}^2$, when used within the DCTR, PHARM, and GFR features. Thus, to obtain a quantity that is more closely related to the expectation of the residual distortion, we use the square root $\delta_{uSA}^{1/2}(\boldsymbol{\beta})$ meaning that it is applied to the $n'_1 \times n'_2$ matrix $\delta_{uSA}(\boldsymbol{\beta})$ elementwise. The above claims are supported by Figure 3.1.1, which shows $\delta_{uSA}^{1/2}(\boldsymbol{\beta})_{ij}^{(a,b)}$ versus $\delta_{EA}(\boldsymbol{\beta})_{ij}^{(a,b)}$ across all blocks (a, b) for sixteen different combinations of DCTR kernels \mathbf{G} and JPEG phases. The values of δ_{EA} were obtained using Monte Carlo simulations by embedding the image ‘1013.pgm’ from BOSSBase 1000-times. The first ordered pair above each plot shows the spatial frequency of the DCT kernel while the second ordered pair is the JPEG phase i, j . Note that with the exception of kernel-phase combinations (4, 3), (4, 1) and (4, 3), (6, 7), there appears to be an approximate linear relationship between $\delta_{uSA}^{1/2}$ and δ_{EA} . Qualitatively similar results were observed for the PHARM and GFR filter banks. The square root thus indeed makes $\delta_{uSA}^{1/2}(\boldsymbol{\beta})$ a rather good (and much more computationally efficient!) approximation of $\delta_{EA}(\boldsymbol{\beta})$.

3.1.1 Final feature design

We now summarize the final design of the features that will be subjected to experimental tests in the next section. In the pseudo-code 3.1, $\boldsymbol{\beta} \in \mathbb{R}^{n_1 \times n_2}$ is the selection channel in the form of a matrix of embedding change probabilities of DCT coefficients arranged in the same fashion as unquantized DCT coefficients, f_{kl}^{ij} are the DCT bases (1.5.1), and q_{kl} is the 8×8 JPEG luminance quantization matrix of the investigated image.

3.2 Experimental results

In this section, we subject the selection-channel-aware features described in Section 3.1 to a test on real imagery. The experiments are conducted on the standard database BOSSbase 1.01 [7] containing 10,000 grayscale images with 512×512 pixels. We ran the experiments on JPEG images with quality factors 75 and 95.

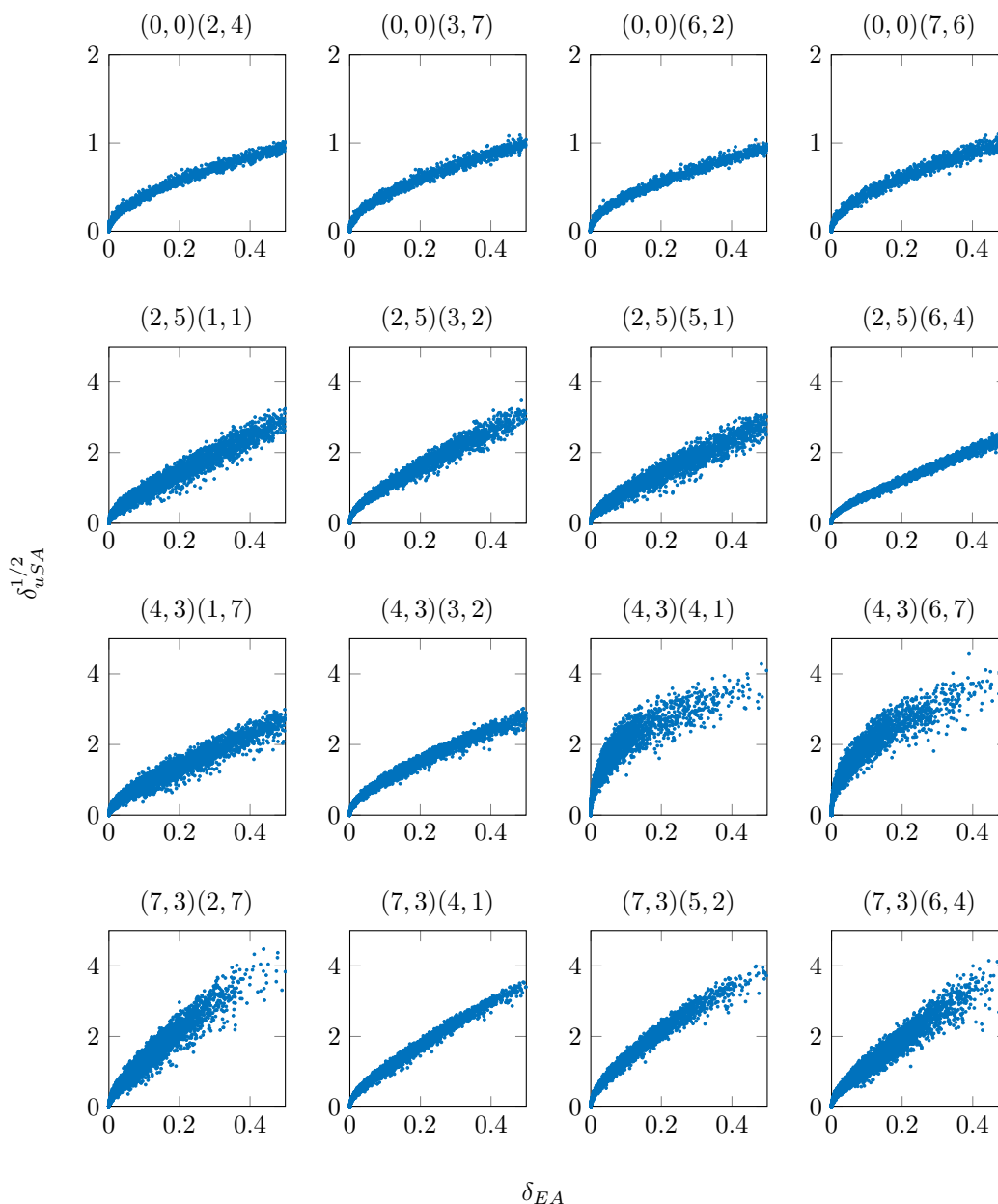


Figure 3.1.1: Plot of $\delta_{uSA}^{1/2}$ versus δ_{EA} for one BOSSbase image for the DCTR filter bank. The first number pair above each scatter plot indicates the DCTR kernel (the spatial frequency of the DCT mode) while the second pair is the JPEG phase. Note that the square root forces an approximate linear relationship between both quantities.

- 1: Select a JPEG-phase-aware feature set, which is equivalent to selecting the filter bank $\mathcal{B} \in \{\mathcal{B}_{DCTR}, \mathcal{B}_{PHARM}, \mathcal{B}_{GFR}\}$.
- 2: Decompress the JPEG image under investigation to the spatial domain (apply by blocks), denote the non-rounded pixel values with $\tilde{\mathbf{X}}$.
- 3: **for all** $\mathbf{G} \in \mathcal{B}$ **do**
- 4: Compute the residual $\mathbf{R}(\tilde{\mathbf{X}}, \mathbf{G}) = \mathbf{G} \star \tilde{\mathbf{X}}$ and quantize it $\mathbf{R}(\tilde{\mathbf{X}}, \mathbf{G}, Q) = Q_{\mathcal{Q}}(\mathbf{R}(\tilde{\mathbf{X}}, \mathbf{G})/q)$.
- 5: Compute $\mathbf{t}(\boldsymbol{\beta}) \in \mathbb{R}^{n_1 \times n_2}$ by blocks, $t_{ij}^{(a,b)}(\boldsymbol{\beta}) = 2 \sum_{k,l=0}^7 |J_{kl}^{ij}| q_{kl} \beta_{kl}^{(a,b)}$ for all blocks (a, b) .
- 6: Evaluate $\delta_{uSA}^{1/2}(\boldsymbol{\beta}) = \sqrt{\mathbf{t}(\boldsymbol{\beta}) \star |\mathbf{G}|}$ (square root applied in an elementwise fashion to all elements of the $n'_1 \times n'_2$ matrix $\mathbf{t}(\boldsymbol{\beta}) \star |\mathbf{G}|$).
- 7: Compute the following $64 \times (T + 1)$ values $\bar{h}_m^{(i,j)}(\tilde{\mathbf{X}}, \mathbf{G}, Q, \boldsymbol{\beta})$, $0 \leq m \leq T$, $0 \leq i, j \leq 7$:

$$\bar{h}_m^{(i,j)}(\tilde{\mathbf{X}}, \mathbf{G}, Q, \boldsymbol{\beta}) = \sum_{a=1}^{\lfloor n'_1/8 \rfloor} \sum_{b=1}^{\lfloor n'_2/8 \rfloor} [|r_{ij}^{(a,b)}(\tilde{\mathbf{X}}, \mathbf{G}, Q)| = m] \cdot \delta_{uSA}^{1/2}(\boldsymbol{\beta})_{ij}^{(a,b)}.$$

Note that the histogram bins $\bar{h}_m^{(i,j)}$ now depend on the selection channel $\boldsymbol{\beta}$.

- 8: **end for**
- 9: Concatenate $\bar{h}_m^{(i,j)}(\tilde{\mathbf{X}}, \mathbf{G}, Q, \boldsymbol{\beta})$, $\mathbf{G} \in \mathcal{B}$, $0 \leq i, j \leq 7$, $0 \leq m \leq T$, and form the final feature vector using the same symmetrization rules as those used for forming the JPEG-phase-aware features from $h_m^{(i,j)}(\tilde{\mathbf{X}}, \mathbf{G}, Q)$.

Algorithm 3.1: Pseudo-code for $\delta_{uSA}^{1/2}$ selection-channel aware JPEG features.

The steganographic algorithms tested in this section are the original version of UED [42] (UED-SC), its improved version [43] (UED-JC), and J-UNIWARD as described in [51]. We use three JPEG-phase-aware steganalysis feature sets: DCTR [49], PHARM [50], and Gabor Filter Bank (GFR) [85].

The detection accuracy is evaluated using the minimal total error probability under equal priors, $P_E = \min_{P_{FA}} \frac{1}{2}(P_{FA} + P_{MD})$, achieved on the test set averaged over ten 50/50 splits of the database. The symbols P_{FA} and P_{MD} stand for the false-alarm and missed-detection rates. The classifier is the FLD ensemble [64]. To inform the reader about the statistical significance of the improvements, we state that the mean absolute deviation of P_E over the ten ensemble runs ranges between 0.0005 and 0.0046, depending on the feature set, embedding algorithm, payload, and JPEG quality factor (also see Table 3.1).

Figure 3.2.1 shows the average detection error \bar{P}_E as a function of payload in bits per non-zero AC DCT coefficient (bpnzac) for three steganographic algorithms and two JPEG quality factors and payloads ranging from 0.05–0.5 bpnzac. The exact numerical values are in Table 3.1. For easy comprehension, color is used to highlight the embedding algorithm. Each combination of the embedding algorithm, payload, and JPEG quality factor has two partially overlapping bars with the solid color fill showing the performance of the selection-channel-aware features computed with $\delta_{uSA}^{1/2}$ while the original features correspond to the patterned column.

The GFR feature set always offers the most accurate detection irrespectively of the feature type, the embedding algorithm, payload, and quality factor. Making the features aware of the selection channel generally improves the detection for payload smaller than 0.3 bpnzac. The gain is larger for quality factor 75 than for 95. In some cases, the detection error drops by as much as 8% (UED-JC for 75 quality factor). With increasing payload, the gain decreases, which is natural because the embedding algorithms become less adaptive. For large payloads embedded with UED, the gain may even become negative (the detection slightly worsens). We verified that this loss is not due to the fact that the embedding change probabilities extracted from the stego and cover images differ as the loss remains unchanged when computing the features with embedding probabilities of the cover. This thus points to a small suboptimality within the quantity $\delta_{uSA}^{1/2}$ accumulated in the histograms.

J-UNI, QF 75%	0.05 bpp	0.1 bpp	0.2 bpp	0.3 bpp	0.4 bpp	0.5 bpp
DCTR	0.4769±0.0013	0.4400±0.0017	0.3412±0.0017	0.2410±0.0030	0.1553±0.0023	0.0887±0.0025
$\delta_{uSA}^{1/2}$ DCTR	0.4635±0.0028	0.4192±0.0015	0.3081±0.0021	0.2148±0.0026	0.1380±0.0019	0.0818±0.0015
GFR	0.4638±0.0019	0.4095±0.0013	0.2861±0.0037	0.1804±0.0029	0.1005±0.0021	0.0546±0.0018
$\delta_{uSA}^{1/2}$ GFR	0.4325±0.0016	0.3589±0.0025	0.2272±0.0026	0.1389±0.0019	0.0792±0.0016	0.0437±0.0008
PHARM	0.4746±0.0024	0.4284±0.0024	0.3131±0.0039	0.2096±0.0029	0.1259±0.0027	0.0720±0.0017
$\delta_{uSA}^{1/2}$ PHARM	0.4490±0.0017	0.3929±0.0023	0.2685±0.0027	0.1678±0.0034	0.0974±0.0021	0.0555±0.0016

UED-SC, QF 75%	0.05 bpp	0.1 bpp	0.2 bpp	0.3 bpp	0.4 bpp	0.5 bpp
DCTR	0.4301±0.0020	0.3497±0.0026	0.2013±0.0011	0.1034±0.0019	0.0427±0.0011	0.0120±0.0005
$\delta_{uSA}^{1/2}$ DCTR	0.3786±0.0018	0.3030±0.0024	0.1871±0.0028	0.1140±0.0027	0.0578±0.0020	0.0206±0.0008
GFR	0.4029±0.0028	0.3010±0.0036	0.1443±0.0030	0.0617±0.0011	0.0266±0.0009	0.0096±0.0006
$\delta_{uSA}^{1/2}$ GFR	0.3159±0.0023	0.2204±0.0018	0.1103±0.0018	0.0567±0.0008	0.0290±0.0011	0.0131±0.0008
PHARM	0.4168±0.0027	0.3207±0.0041	0.1580±0.0016	0.0711±0.0022	0.0296±0.0009	0.0121±0.0006
$\delta_{uSA}^{1/2}$ PHARM	0.3529±0.0023	0.2519±0.0035	0.1223±0.0024	0.0559±0.0019	0.0252±0.0008	0.0111±0.0006

UED-JC, QF 75%	0.05 bpp	0.1 bpp	0.2 bpp	0.3 bpp	0.4 bpp	0.5 bpp
DCTR	0.4358±0.0024	0.3593±0.0020	0.2192±0.0030	0.1204±0.0020	0.0593±0.0014	0.0203±0.0009
$\delta_{uSA}^{1/2}$ DCTR	0.3865±0.0020	0.3146±0.0025	0.2075±0.0032	0.1303±0.0023	0.0747±0.0026	0.0328±0.0014
GFR	0.4100±0.0023	0.3153±0.0024	0.1627±0.0022	0.0779±0.0016	0.0346±0.0018	0.0153±0.0007
$\delta_{uSA}^{1/2}$ GFR	0.3301±0.0040	0.2352±0.0028	0.1252±0.0008	0.0685±0.0017	0.0377±0.0008	0.0193±0.0010
PHARM	0.4213±0.0029	0.3376±0.0043	0.1837±0.0023	0.0895±0.0025	0.0418±0.0005	0.0187±0.0011
$\delta_{uSA}^{1/2}$ PHARM	0.3649±0.0035	0.2694±0.0036	0.1399±0.0028	0.0719±0.0015	0.0348±0.0007	0.0181±0.0008

J-UNI, QF 95%	0.05 bpp	0.1 bpp	0.2 bpp	0.3 bpp	0.4 bpp	0.5 bpp
DCTR	0.4965±0.0013	0.4826±0.0019	0.4424±0.0032	0.3820±0.0029	0.3081±0.0025	0.2310±0.0020
$\delta_{uSA}^{1/2}$ DCTR	0.4924±0.0042	0.4705±0.0026	0.4163±0.0034	0.3524±0.0036	0.2851±0.0030	0.2217±0.0027
GFR	0.4915±0.0019	0.4756±0.0013	0.4215±0.0016	0.3496±0.0018	0.2721±0.0028	0.1936±0.0023
$\delta_{uSA}^{1/2}$ GFR	0.4843±0.0015	0.4634±0.0027	0.4046±0.0024	0.3338±0.0029	0.2617±0.0038	0.1998±0.0026
PHARM	0.4953±0.0017	0.4835±0.0018	0.4416±0.0032	0.3787±0.0023	0.3079±0.0026	0.2311±0.0038
$\delta_{uSA}^{1/2}$ PHARM	0.4896±0.0024	0.4762±0.0022	0.4304±0.0020	0.3690±0.0022	0.2981±0.0014	0.2294±0.0019

UED-SC, QF 95%	0.05 bpp	0.1 bpp	0.2 bpp	0.3 bpp	0.4 bpp	0.5 bpp
DCTR	0.4826±0.0019	0.4555±0.0017	0.3829±0.0015	0.3004±0.0025	0.2095±0.0015	0.1201±0.0019
$\delta_{uSA}^{1/2}$ DCTR	0.4719±0.0026	0.4420±0.0025	0.3752±0.0026	0.3079±0.0020	0.2296±0.0017	0.1484±0.0022
GFR	0.4679±0.0018	0.4294±0.0016	0.3299±0.0022	0.2295±0.0035	0.1420±0.0027	0.0753±0.0012
$\delta_{uSA}^{1/2}$ GFR	0.4366±0.0031	0.3896±0.0019	0.2964±0.0028	0.2149±0.0031	0.1462±0.0029	0.0899±0.0031
PHARM	0.4779±0.0025	0.4455±0.0029	0.3577±0.0035	0.2605±0.0032	0.1674±0.0024	0.0982±0.0021
$\delta_{uSA}^{1/2}$ PHARM	0.4641±0.0029	0.4246±0.0035	0.3348±0.0018	0.2479±0.0021	0.1658±0.0019	0.1008±0.0022

UED-JC, QF 95%	0.05 bpp	0.1 bpp	0.2 bpp	0.3 bpp	0.4 bpp	0.5 bpp
DCTR	0.4835±0.0021	0.4598±0.0025	0.3908±0.0025	0.3095±0.0029	0.2180±0.0026	0.1216±0.0017
$\delta_{uSA}^{1/2}$ DCTR	0.4753±0.0026	0.4426±0.0039	0.3830±0.0040	0.3069±0.0025	0.2128±0.0027	0.1146±0.0030
GFR	0.4709±0.0014	0.4323±0.0016	0.3420±0.0024	0.2492±0.0028	0.1663±0.0028	0.0990±0.0027
$\delta_{uSA}^{1/2}$ GFR	0.4411±0.0020	0.3931±0.0021	0.3077±0.0025	0.2316±0.0034	0.1662±0.0035	0.1107±0.0024
PHARM	0.4994±0.0016	0.4490±0.0019	0.3708±0.0039	0.2828±0.0018	0.1947±0.0022	0.1220±0.0022
$\delta_{uSA}^{1/2}$ PHARM	0.4622±0.0025	0.4288±0.0022	0.3473±0.0021	0.2665±0.0033	0.1901±0.0014	0.1242±0.0031

Table 3.1: Average P_E for three steganographic schemes for DCTR, GFR, and PHARM features and their selection-aware $\delta_{uSA}^{1/2}$ version for selected payloads, JPEG quality factors 75 and 95.

The significant detection gain for small payloads far outweighs this small loss as it is more difficult to detect smaller payloads.

Finally, even though the goal of this section is not to benchmark steganography, it is interesting that the order of the three tested steganographic schemes by their empirical security does not change when switching to selection-channel-aware features and does not depend on the feature type either.

3.3 Conclusions

Steganalysis of content-adaptive steganography needs to take into account the probabilities with which the embedding modifies individual cover elements. However, incorporating this prior probabilistic knowledge (the selection channel) within detectors built as classifiers trained on examples of cover and stego features is quite challenging. The main complication stems from the fact that the quantity from which steganalysis features are formed are quantized noise residuals extracted in the pixel domain. When the embedding modifies JPEG DCT coefficients, the impact of embedding on residuals becomes even more complicated. Fortunately, if the residuals are obtained in a linear fashion from pixels, e.g., by convolving the image with a kernel, because the embedding changes are independent, it is possible to derive the impact of embedding on residuals analytically.

In this chapter, we investigate several quantities that measure the expected embedding distortion in the residual domain when embedding in the JPEG domain. In order to obtain a distortion measure that can be evaluated with acceptable computational complexity, we consider an upper bound on the L_1 distortion and transform it in a non-linear manner to make it strongly correlate with the true expected value of L_1 distortion. The resulting quantity can be efficiently computed using convolutions and is accumulated in residual histograms of three feature sets that are aware of the JPEG phase: DCTR, PHARM, and Gabor Filter Residuals (GFR). These feature sets were selected because they currently provide the most accurate detection of modern steganography in JPEG domain. The selection-channel-aware versions of these features provide a significant detection gain of content-adaptive JPEG steganography, especially for small payloads.

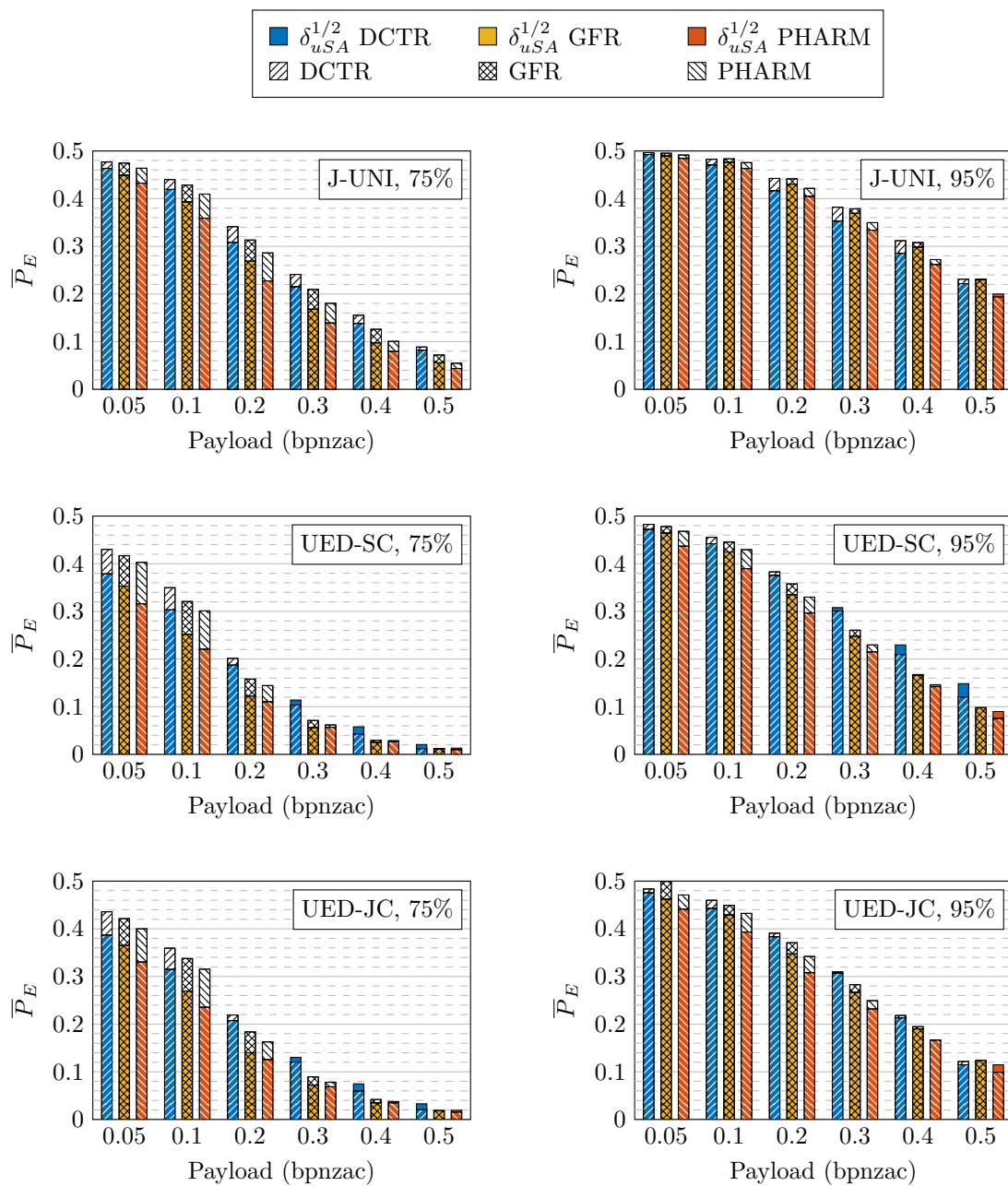


Figure 3.2.1: Average P_E for three steganographic algorithms for DCTR, GFR, and PHARM features (patterns) and their selection-aware $\delta_{u,SA}^{1/2}$ version (solid fill) versus payload, JPEG quality factors 75 and 95.

Chapter 4

Selection-channel aware attacks in residual domain

With the insight from the previous chapter, a careful reader will realize that the design of the maxSRMd2 feature set from the Chapter 2 is flawed. The residuals collected inside co-occurrences depend on numerous pixels and one can no longer associate a pixel change rate with a given residual sample. This chapter resolves this issue by replacing the change rate with the expected value of the residual distortion as the quantity that should be accumulated in the histograms (for JPEG phase-aware features and projection type features) and in co-occurrences (for SRM).

This extension is relatively straightforward for linear residuals since the relationship tying the embedding domain and the residual domain is linear. If the embedding changes are executed independently,¹ one can easily compute the expected value of the embedding distortion in the residual domain analytically. A major complication, however, occurs for non-linear residuals due to the necessity to compute marginals of high-dimensional probability mass functions. This is why the emphasis of this chapter is on rich representations formed from linear residuals.

4.1 Replacing change rates with L_1 distortion of residuals

As pointed out in the introduction, there is a discrepancy in maxSRM in the sense that we accumulate the embedding change probabilities of *pixels* in the co-occurrence bins of *residuals*. Thus, we need to move away from pixel change rates to some measure of the residual distortion. After all, if the features were formed from pixel values rather than residuals, the change rates are proportional to the expected value of the L_1 (and L_2) distortion. This is because in most modern steganographic schemes the cover pixel x_{ij} is modified to $y_{ij} = x_{ij} + 1$ and $y_{ij} = x_{ij} - 1$ with the same probability β_{ij} and thus $E[|x_{ij} - y_{ij}|] = E[|x_{ij} - y_{ij}|^2] = 2\beta_{ij}$.

We explain the approach only for linear residuals and then discuss the issues with non-linear residuals.

4.1.1 Linear residuals

We recall that a linear residual \mathbf{Z} in SRM is obtained by convolving the image with a kernel, this time we make the dependence of \mathbf{Z} on the image explicit:

$$\mathbf{Z}^{(\text{SRM})}(\mathbf{X}) = \mathbf{K} \star \mathbf{X}, \quad (4.1.1)$$

¹This is true for all current steganographic schemes with the notable exception of steganography that synchronizes the selection channel [17, 67].

and in coordinates:

$$z_{ij}^{(\text{SRM})}(\mathbf{X}) = \sum_{k,l} K_{kl} x_{i-k,j-l}. \quad (4.1.2)$$

The specific range for the indices k and l depends on the kernel support. Note that in PSRM the residual is additionally convolved with a projection matrix $\mathbf{\Pi}$:

$$\mathbf{Z}^{(\text{PSRM})}(\mathbf{X}) = \mathbf{\Pi} \star (\mathbf{K} \star \mathbf{X}) = (\mathbf{\Pi} \star \mathbf{K}) \star \mathbf{X} \quad (4.1.3)$$

due to the associativity of convolution. Thus, irrespectively of whether we deal with a linear residual from SRM or PSRM, the quantity whose sample statistic is collected (either a fourth-order co-occurrence or a histogram) is obtained by convolving the image with a kernel.

For steganographic schemes minimizing an additive distortion, message embedding is equivalent to adding noise whose distribution depends on the pixel location:

$$y_{ij} = x_{ij} + \xi_{ij},$$

where ξ_{ij} are independent random variables attaining their values in $\{-1, 0, 1\}$ with probabilities $\beta_{ij}, 1 - \beta_{ij}, \beta_{ij}$. Thus, each element of the difference $\mathbf{Z}(\mathbf{Y}) - \mathbf{Z}(\mathbf{X})$ is a random variable with

$$E[z_{ij}(\mathbf{Y}) - z_{ij}(\mathbf{X})] = E\left[\sum_{k,l} K_{kl} \xi_{i-k,j-l}\right] = 0, \quad (4.1.4)$$

$$\text{Var}[z_{ij}(\mathbf{Y}) - z_{ij}(\mathbf{X})] = 2 \sum_{k,l} K_{kl}^2 \beta_{i-k,j-l}. \quad (4.1.5)$$

While it is straightforward to evaluate the expectation of the L_2 norm

$$\begin{aligned} E\left[(z_{ij}(\mathbf{Y}) - z_{ij}(\mathbf{X}))^2\right] &= 2 \sum_{k,l} K_{kl}^2 \beta_{i-k,j-l} \\ &= \text{Var}[z_{ij}(\mathbf{Y}) - z_{ij}(\mathbf{X})] \\ &\triangleq \sigma_{ij}^2 \end{aligned} \quad (4.1.6)$$

due to the independence of embedding changes, it is much more difficult to compute the expectation of the absolute value, $E[|z_{ij}(\mathbf{Y}) - z_{ij}(\mathbf{X})|]$. We will thus consider a simplification and assume that $z_{ij}(\mathbf{Y}) - z_{ij}(\mathbf{X})$ is a zero-mean Gaussian random variable with variance (4.1.5). In this case, it is easy to evaluate

$$E[|z_{ij}(\mathbf{Y}) - z_{ij}(\mathbf{X})|] = \frac{2}{\sqrt{\pi}} \sqrt{\sum_{k,l} K_{kl}^2 \beta_{i-k,j-l}} \propto \sigma_{ij}. \quad (4.1.7)$$

In our experiments, the Gaussian approximation of the expectation of the L_1 distortion (4.1.7) worked much better than the L_2 distortion. This is why in the rest of this chapter, we only use σ_{ij} as the quantity that will be accumulated in co-occurrences in SRM and in histograms in PSRM of linear residuals.

For SRM, the selection-channel-aware features built from a linear residual will be formed by replacing the change rates $\hat{\beta}_{ij}$ in Eqs. (1.2.10) (for the horizontal and vertical scans) and (1.2.10) (for the 'd2' scan) with σ_{ij} :

$$c_{d_0 d_1 d_2 d_3}^{\sigma \text{SRM}} = \sum_{i,j=1}^{n_1, n_2-3} \max_{k=0, \dots, 3} \sigma_{i,j+k} [r_{i,j+k} = d_k, \forall k = 0, \dots, 3]. \quad (4.1.8)$$

For the PSRM, the histograms of linear residuals (1.2.14) are replaced with their σ version:

$$h_m^{(k)\sigma} = \sum_{i,j=1}^{n_1, n_2} \sigma_{ij} \times [\tilde{p}_{ij}^{(k)} = m + 1/2], \quad (4.1.9)$$

$$m \in \{0, 1, \dots, T - 1\}, k \in \{1, \dots, \nu\}. \quad (4.1.10)$$

At this point, we remark on the dimensionality of the σ -version of the PSRM. In the original PSRM, linear residuals are represented using only T bins because the last, $T + 1$ -st bin with centroid at $T + 1/2$ is uniquely determined by the other bins (the sum $\sum_{m \in \mathcal{Q}} h_m^{(k)} = n_1 n_2$). This is not true for $h_m^{(k)\sigma}$ as the sum of all $T + 1$ bins is no longer equal to the number of residual values $n_1 n_2$. In our experiments, we did not see any statistically significant benefit in using all $T + 1$ bins in $h_m^{(k)\sigma}$, which is why in the σ -version of the PSRM, we also skip the last $T + 1$ -st bin to keep the same feature dimensionality. Similarly, histograms of non-linear residuals in PSRM are represented using only $2T$ bins (both the first and the last values corresponding to centroids $-T - 1/2$ and $T + 1/2$ are skipped) and we keep the same arrangement in the σ -version.

4.1.2 Non-linear residuals

The situation for non-linear minmax residuals is significantly more complicated because the residuals whose minimum (maximum) is computed are generally dependent random variables. In the most extreme case, which corresponds to the 'minmax41' submodel in EDGE5x5 residual of SRM, the minimum (maximum) is taken over four values that each depend on 15 neighboring pixel values out of the local 5×5 neighborhood. Computing the expectation of the L_1 or L_2 norm thus requires marginalization of a 25-dimensional probability mass function with 3^{25} values. We were unable to find an algorithm whose computational complexity would be sufficiently low to make the feature extractor run in reasonable time. Out of the ideas that have been explored, we list the following.

One could work with simplifying assumptions, such as the Gaussianity of the underlying residuals and repeatedly leverage the analytic expression for the distribution of the minimum / maximum of two Gaussian variables [71]. The Gaussianity assumption will, however, be invalid for residuals with a small support, such as the 'minmax54' first-order residual, and this deviation will lead to suboptimality.

Another possibility is to estimate the expectation $E[|z_{ij}^{\min}(\mathbf{Y}) - z_{ij}^{\min}(\mathbf{X})|]$ using Monte-Carlo simulation, which is a rather expensive alternative. Nevertheless, this approach will tell us how much can be theoretically gained.

4.2 Experiments

This section contains the results of all experiments. They were conducted on the standard BOSSbase 1.01 [7] database containing 10,000 grayscale images with 512×512 pixels. The detection accuracy is evaluated using the minimal total error probability on the testing set under equal priors, $P_E = \min_{P_{FA}} \frac{1}{2}(P_{FA} + P_{MD})$, returned by the FLD ensemble [64] averaged over ten 50/50 splits of the database into a pair of training and testing sets.

Before moving to the actual experiments, we summarize the terminology. The version of maxSRM with the quantity $E[|z_{ij}(\mathbf{Y}) - z_{ij}(\mathbf{X})|]$ accumulated in co-occurrences will be denoted σ maxSRM. We note that for linear residuals σ_{ij} is computed using (4.1.7) while for min-max residuals, it is obtained using Monte Carlo simulations by embedding the image under investigation 500 times. For the PSRM, we only use the 'spam' type submodels corresponding to linear residuals. This gives our feature set dimensionality of 1,980. This feature set will be abbreviated σ spamPSRM.

		0.2 bpp			0.4 bpp		
		\bar{P}_E	min \bar{P}_E	max \bar{P}_E	\bar{P}_E	min \bar{P}_E	max \bar{P}_E
HILL	maxSRMq2d2	0.3181	0.3149	0.3228	0.2238	0.2174	0.2278
	σ maxSRMq2d2	0.3075	0.3015	0.3109	0.2132	0.2104	0.2146
WOW	maxSRMq2d2	0.2472	0.2400	0.2530	0.1658	0.1601	0.1732
	σ maxSRMq2d2	0.2449	0.2397	0.2509	0.1620	0.1569	0.1694
MVG	maxSRMq2d2	0.3291	0.3228	0.3336	0.2309	0.2287	0.2347
	σ maxSRMq2d2	0.3205	0.3160	0.3239	0.2202	0.2138	0.2303

Table 4.1: Detection of three steganographic algorithms for two payloads on BOSSbase 1.01 using the original maxSRM features and their proposed σ maxSRM form.

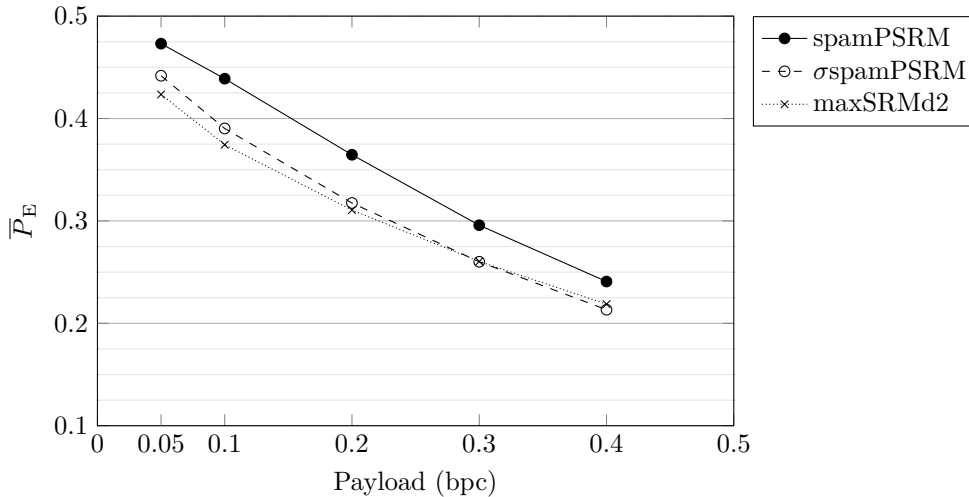


Figure 4.2.1: Detection error \bar{P}_E for HILL with spamPSRM, σ spamPSRM, and maxSRMd2.

Our first experiment demonstrates the potential of the proposed idea. We work with σ maxSRMq2d2 (d2 standing for the d2 scan of co-occurrences) with 12,753 features. Table 4.1 shows the results for three steganographic schemes, WOW [45], HILL [66], and MVG [84] with ternary embedding and Gaussian pixel residual model, and two payloads contrasting the detection error for the original maxSRMq2d2 and the proposed σ maxSRMq2d2. The improvement in the detection error \bar{P}_E ranges from 0.3% to almost 1.5%, depending on the embedding algorithm and payload.

The second experiment was executed with the spam part of the PSRM (with linear residuals only). We compare the spamPSRM subset of PSRM with σ spamPSRM (both dimensionality 1,980) because no other selection-channel-aware version of PSRM currently exists. The results appear in Figures 4.2.1–4.2.4. The improvement in the detection error is significant across all embedding algorithms and payloads, especially for HILL and WOW where the improvement in \bar{P}_E ranges between 2.5% and 6%. In fact, this relatively small σ spamPSRM with 1,980 features for HILL and MVG achieves comparable or better detection error than the computationally much more expensive maxSRMd2 (34,671 features). For HILL with payload 0.4 bpp the σ spamPSRM improves on maxSRMd2 by 0.5% and even on PSRM by 2.3%. Only in the case of S-UNIWARD the σ spamPSRM does not significantly improve on spamPSRM.

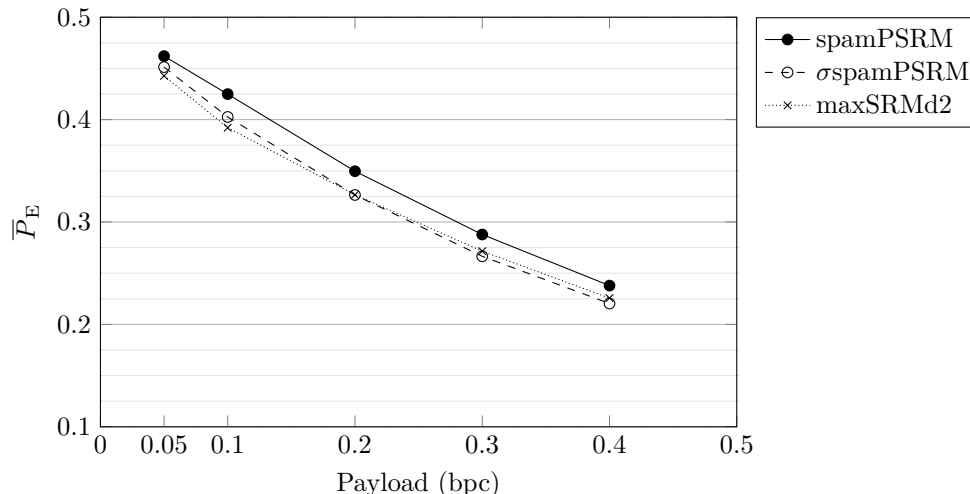


Figure 4.2.2: Detection error \bar{P}_E for MVG with spamPSRM, σ spamPSRM, and maxSRMd2.

4.3 Conclusions

Detection of modern content-adaptive steganography requires detectors built using machine learning fed with examples of cover and stego objects represented in a feature space. Currently, it is an open problem how to choose a suitable feature representation that would incorporate the knowledge of the embedding change probabilities of individual image elements, the selection channel. These probabilities are approximately available to the steganalyst because the pixel costs that were used for embedding can be relatively accurately estimated from the stego image and because the imprecise knowledge of the payload size does not affect steganalysis accuracy much.

The maxSRM feature set described in Chapter 2 calls for accumulating the pixel change probabilities in co-occurrences of noise residuals. However, because potentially many pixels contribute to one residual sample, one should compute the statistical impact of the embedding changes on the residual and accumulate this quantity instead. To this end, in this chapter we propose the expected value of the L_1 residual distortion due to embedding. For linear pixel predictors, the impact of embedding on the residual is easily obtained from the independence of embedding changes and the assumed Gaussianity of the distortion. For non-linear (min-max) predictors, however, the expectation of the L_1 distortion is difficult to obtain analytically due to the necessity to compute the expectation of a minimum (maximum) of up to five dependent random variables that themselves depend on up to 25 pixels. In this chapter, we compute such expectations using Monte Carlo simulations.

The proposed idea is applied to the SRM feature set and a subset of the PSRM that is built only from linear residuals (dimensionality 1,980). This reduction of the PSRM feature vector was needed to keep the computational complexity low. Experiments with three embedding schemes and the SRMq2d2 feature set showed that the proposed quantity indeed improves the detection by 0.5–1.5% depending on the embedding algorithm and payload. In the case of the PSRM, the improvement was quite substantial. Compared with the same subset of the original PSRM, the detection error dropped by up to 6% and was comparable and sometimes even slightly lower (for HILL and MVG) than using the entire (and much more computationally demanding) maxSRMd2 model.

We wish to stress that the proposed modification of the rich models does not increase their dimensionality. When the models are restricted only to the subset obtained from linear residuals, the increase in the computational complexity is negligible since the expectation of the L_1 distortion for one residual can be obtained using three convolutions.

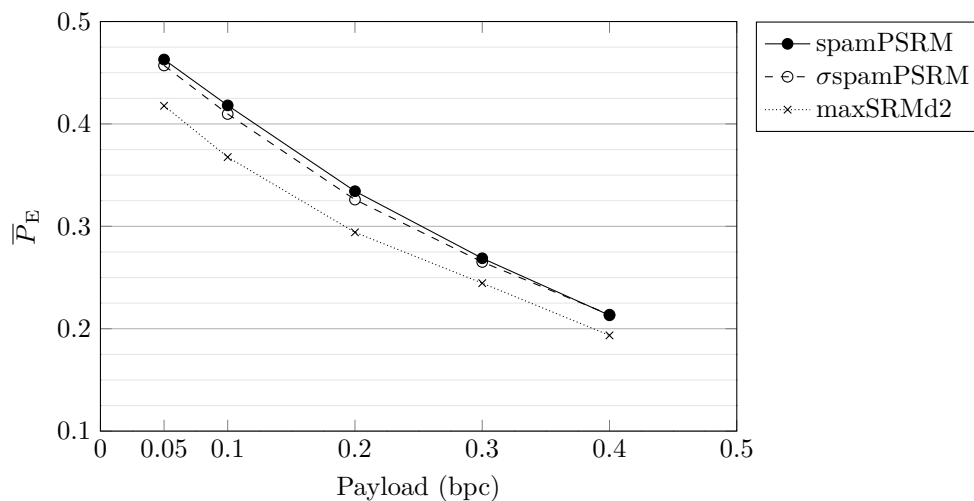


Figure 4.2.3: Detection error \bar{P}_E for S-UNIWARD with spamPSRM, σ spamPSRM, and maxSRMd2.

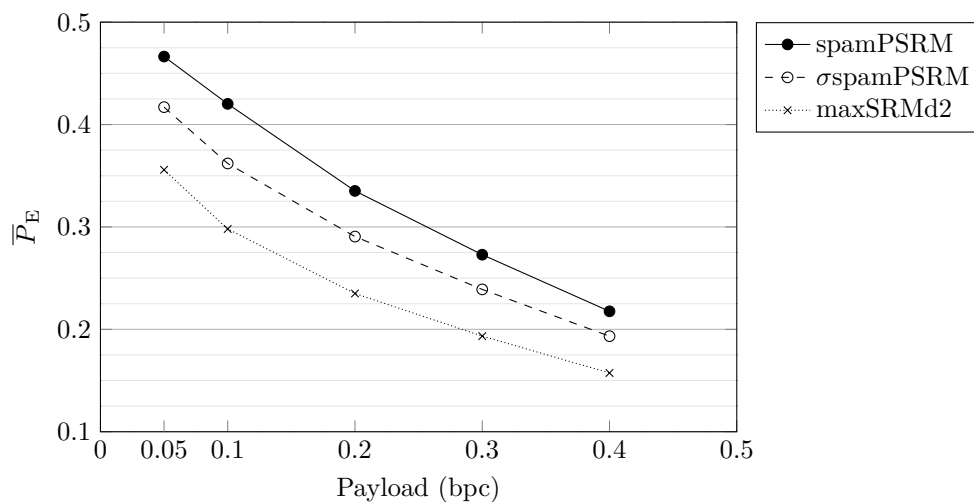


Figure 4.2.4: Detection error \bar{P}_E for WOW with spamPSRM, σ spamPSRM, and maxSRMd2.

Chapter 5

Steganography with precover

Often the cover images Alice sends to Bob undergo some processing, format conversion or even compression. If the party performing these generally lossy operations is Alice, she can utilize the lost information to improve the security of her embedding. This chapter talks about the history of side-informed steganography schemes and then introduces improvements and new approaches.

In the literature, the first published scheme that utilizes the private side-information is embedding-while-dithering [36]. This scheme improves on embedding in palette images (for example GIF) when Alice has access to the original, uncompressed bitmap image. There exist many naive steganographic schemes for palette image formats that for example embed the secret message by shuffling the order of the palette [65]. These methods are, however, vulnerable to targeted attacks, as the very presence of the shuffled palette is suspicious. A better method is to assign each color in the palette a parity and switch each pixel to the closest color with the correct parity. If Alice has access to the original uncompressed image, she can embed during compressing. In traditional GIF compression, to improve the visual quality, the quantization error after quantization of every pixel is tracked and diffused in a process called dithering. Embedding-while-dithering adds the error introduced by adding the stego signal to the quantization error and spreads it across the whole image achieving better visual quality and steganographic security.

An entirely different approach was introduced for the JPEG format in an algorithm called perturbed quantization [37]. The rounding errors of the quantization after DCT transform are used to modify the embedding algorithm. Side-informed techniques based on this idea gained on popularity and many more quickly appeared [61, 77, 91, 52, 43]. In combination with adaptive steganography, this approach culminated in the form of the algorithm SI-UNIWARD [51], which to this day stays as a benchmark for state-of-the-art algorithms for JPEG images. Incorporating side-information.

In this chapter, we introduce a simple idea how to incorporate side-information in any steganographic scheme that minimizes additive distortion. We specifically discuss two novel aspects, which include the departure from a binary embedding operation to ternary and the computation of costs from the unquantized cover.

We will recognize three types of images – a precover image \mathbf{P} , unquantized cover \mathbf{U} , and cover \mathbf{X} . The cover is obtained from the precover using some information-reducing operation that involves quantization as the last step. Even though the proposed approach can certainly be applied to color cover images, for simplicity of the exposition, we will assume that $\mathbf{X} = (x_{ij}) \in \mathcal{I}_L^{n_1 \times n_2}$, $\mathcal{I}_L = \{0, 1, \dots, 2^L - 1\}$, is an L -bit grayscale image with $n_1 \times n_2$ pixels. At this point, we refrain from formalizing the concept of the precover and merely state that it is some higher quality version of the cover. The precover may be in a different format than the cover, it may have a higher color depth, and may be larger than $n_1 \times n_2$ pixels. In this chapter, we also allow the precover to be color. We assume that there be a transformation T that maps the precover \mathbf{P} to $\mathbb{R}^{n_1 \times n_2}$ such that $\mathbf{X} = Q_L(T(\mathbf{P}))$ with $\mathbf{U} = T(\mathbf{P})$ the unquantized cover, where Q_L is a quantizer with 2^L centroids

\mathcal{I}_L . Symbolically,

$$\mathbf{P} \xrightarrow{T} \mathbf{U} \xrightarrow{Q_L} \mathbf{X}. \quad (5.0.1)$$

Furthermore, we will assume that we have a steganographic scheme \mathcal{A} designed to embed while minimizing an additive distortion function. This is currently the most successful paradigm for constructing steganographic schemes in any domain, including those based on non-additive distortion [27, 17]. Such steganography typically starts with computing the costs $(\rho_{ij}^{(\mathcal{A})})$ of changing each cover element x_{ij} . In this chapter, we will use only additive schemes \mathcal{A} where the cost of changing the X_{ij} by 1 and by -1 are the same. Again, most additive embedding schemes do possess this property, e.g., S-UNIWARD, J-UNIWARD [51], HILL [66], WOW [45], and UED [42]. The cost $\rho_{ij}^{(\mathcal{A})}$ typically depends on some local neighborhood of pixel x_{ij} . At this point, we note that in all schemes known to the authors it is possible to compute the costs from the unquantized cover \mathbf{U} instead of the cover \mathbf{X} . To distinguish such costs, we will explicitly mark this dependency: $\rho_{ij}^{(\mathcal{A})}(\mathbf{U})$ or $\rho_{ij}^{(\mathcal{A})}(\mathbf{X})$.

The side-information that our proposed general embedding scheme will utilize is the unquantized cover \mathbf{U} . The quantization error due to applying the quantizer Q_L will be denoted

$$e_{ij} = u_{ij} - x_{ij} = u_{ij} - Q_L(u_{ij}). \quad (5.0.2)$$

If the cover element is modified from x_{ij} to y_{ij} , the total distortion due to quantization and embedding with respect to the unquantized cover is thus

$$e'_{ij} = u_{ij} - y_{ij}. \quad (5.0.3)$$

The costs $\rho_{ij}^{(\text{SI})}$ of the side-informed version of the embedding algorithm \mathcal{A} are obtained by modulating the original costs by the difference $|e'_{ij}| - |e_{ij}|$:

$$\rho_{ij}^{(\text{SI})} = (|e'_{ij}| - |e_{ij}|)\rho_{ij}^{(\mathcal{A})}. \quad (5.0.4)$$

Note that $\rho_{ij}^{(\text{SI})} \geq 0$ because $|e_{ij}| \leq |e'_{ij}|$ for all ij as e_{ij} is the smallest amount u_{ij} can be modified to obtain a plausible cover value (a value from \mathcal{I}_L). The modulation in (5.0.4) makes intuitive sense because the new costs reflect not only the local image complexity but also take into account the distortion w.r.t. the unquantized cover. The hope is that by minimizing this distortion, the embedding will disturb the statistical properties of covers less.

At this point, we make a comparison to previous art. Virtually all previously proposed side-informed schemes were restricted to binary embedding operations. During embedding, the value u_{ij} was either quantized to $y_{ij} = x_{ij} = Q_L(u_{ij})$ or it was rounded “to the other side” $y_{ij} = x_{ij} + \text{sign}(u_{ij} - x_{ij}) = u_{ij} + \text{sign}(e_{ij}) - e_{ij}$, which means the embedding operation was inherently binary. In this case, $|e'_{ij}| - |e_{ij}| = 1 - 2|e_{ij}|$.

In this chapter, we allow ternary side-informed embedding. Indeed, when $e_{ij} \approx 0$, there is no reason to restrict the embedding to a binary operation as rounding to $y_{ij} = x_{ij} + \text{sign}(e_{ij})$ becomes almost as expensive ($|e'_{ij}| - |e_{ij}| = 1 - 2|e_{ij}|$) as rounding to $y_{ij} = x_{ij} - \text{sign}(e_{ij})$ ($|e'_{ij}| - |e_{ij}| = 1 + |e_{ij}| - |e_{ij}| = 1$). Thus, when using ternary embedding in our side-informed steganography, the costs of changing x_{ij} by ± 1 are not equal:

$$\rho_{ij}^{(\text{SI})+} = (1 - 2|e_{ij}|)\rho_{ij}^{(\mathcal{A})} \quad \text{if } y_{ij} = x_{ij} + \text{sign}(e_{ij}), \quad (5.0.5)$$

$$\rho_{ij}^{(\text{SI})-} = \rho_{ij}^{(\mathcal{A})} \quad \text{if } y_{ij} = x_{ij} - \text{sign}(e_{ij}). \quad (5.0.6)$$

The actual embedding needs to be executed with the multi-layered version of syndrome-trellis codes (STCs) [28]. An embedding simulator will change pixel x_{ij} by $\pm \text{sign}(e_{ij})$ with probabilities

$$\beta_{ij}^{(\pm)} = \frac{e^{-\lambda\rho_{ij}^{(\text{SI})\pm}}}{1 + e^{-\lambda\rho_{ij}^{(\text{SI})+}} + e^{-\lambda\rho_{ij}^{(\text{SI})-}}}, \quad (5.0.7)$$

with $\lambda > 0$ determined by the payload length M (in bits):

$$M = \sum_{i,j} -\beta_{ij}^+ \log_2 \beta_{ij}^+ - \beta_{ij}^- \log_2 \beta_{ij}^- - (1 - \beta_{ij}^+ - \beta_{ij}^-) \log_2 (1 - \beta_{ij}^+ - \beta_{ij}^-). \quad (5.0.8)$$

5.1 Discussion and relationship to prior art

The modulation of costs by the difference $|e'_{ij}| - |e_{ij}|$ has been proposed in the past. The BCHopt embedding scheme [77] for JPEG images modulates the costs in the form of the quantization steps with this factor. The same factor also appears in EBS [91], NPQ [52], UED [43], and SI-UNIWARD [51]. Here, we note that the description of SI-UNIWARD as appeared in [51] does not correspond to the actual implementation available on the authors' web site (http://dde.binghamton.edu/download/stego_algorithms/). We elaborate on this issue in the appendix. What is new in our proposal is using the factor to modulate the costs of any additive steganography, for example, in the spatial domain. Additionally, we propose two more innovations – the ternary embedding operation and we also compute the costs of \mathcal{A} from the unquantized cover \mathbf{U} rather than the cover \mathbf{X} . It is shown in the next section that both further improve the empirical security.

5.2 Experiments

The precover source for all our experiments was the BOSSbase 1.01 [7] database with images in their full resolution RAW format. We used a script that utilized 'ufraw' to convert them to the RGB TIFF format of the same resolution. This included gain adjustment, gamma correction, and color interpolation. All subsequent processing was done in Matlab rather than ImageMagick to obtain an easy access to the non-rounded values. The final quantizer Q_L used $L = 8$ for spatial domain and $L = 11$ for experiments in the JPEG domain. For brevity, the transformation T will be described symbolically by arrows between different image representations in the form $(\text{color})_{\text{type}}^{\text{size}}$, where $\text{color} \in \{\text{RAW}, \text{RGB}, \text{GRAY}\}$, $\text{size} \in \{\text{FULL}, 512^2\}$ for full-size and 512×512 images, and $\text{type} \in \{n\text{B}, \text{DBL}\}$ for n bit integers and doubles.

A side-informed scheme based on embedding algorithm \mathcal{A} that uses q -ary embedding operation and computes the costs from image $\mathbf{C} \in \{\mathbf{U}, \mathbf{X}\}$ will be denoted as $\text{SI}q\text{-}\mathcal{A}\text{-}\mathbf{C}$.

All detectors were trained as binary classifiers implemented using the FLD ensemble [64] with default settings. The security is evaluated using the ensemble's 'out-of-bag' (OOB) error E_{OOB} averaged over ten ensemble runs with different seeds. In the spatial domain, we steganalyze with the SRM features [39] while PHARM features [50] were used for the JPEG domain.

5.2.1 Spatial domain

In this subsection, we investigate the empirical security of side-informed S-UNIWARD [51] and HILL [66] when \mathbf{U} represents non-rounded pixel values. The goal of the three experiments below is to investigate the effect of the transformation T , the difference between binary and ternary embedding, and between computing the costs from \mathbf{U} and \mathbf{X} . Because our cover images were obtained using operations from Matlab instead of ImageMagick, our cover source differs from the original BOSSbase 1.01, and the detection errors will be different than the ones reported in [66] and [51]. This is not an issue as we are primarily interested in the relative improvement of the side-informed schemes over the original algorithms.

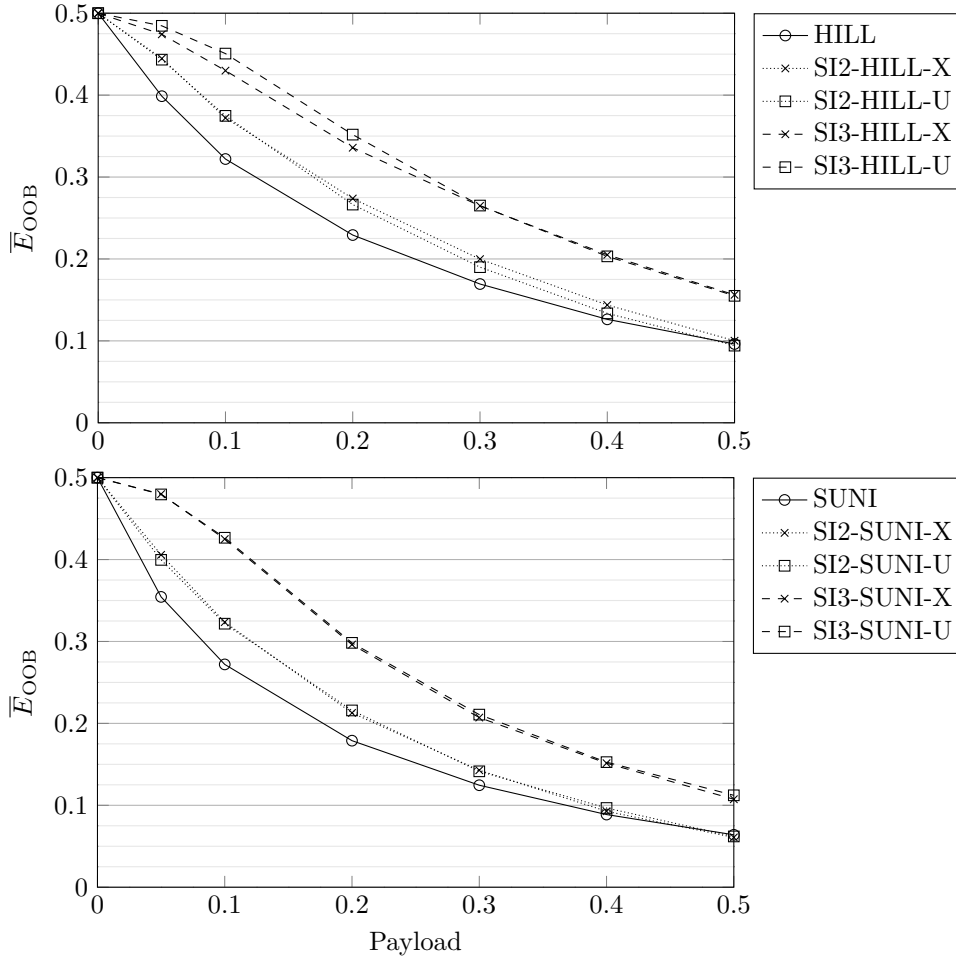


Figure 5.2.1: Mean E_{OOB} for HILL (top) and S-UNIWARD (bottom) and their SI versions with the quantization error after resizing with Lanczos 3 kernel as the side-information when computing the costs from the unquantized and quantized cover.

Resizing

$$T : (\text{RAW})^{\text{FULL}} \longrightarrow (\text{RGB})_{8\text{B}}^{\text{FULL}} \xrightarrow{\text{gray}} (\text{GRAY})_{8\text{B}}^{\text{FULL}} \xrightarrow{\text{crop, resize}} (\text{GRAY})_{\text{DBL}}^{512^2}$$

The results for HILL and S-UNIWARD (Fig. 5.2.1) point out two important facts. First, the choice between computing the costs from \mathbf{U} and \mathbf{X} has a small effect on the overall detection because the costs of both algorithms are insensitive to small perturbations of the cover. Thus, in all our subsequent experiments we use just the costs computed from the unquantized cover, $\rho_{ij}(\mathbf{U})$. Second, the gain in security when using the ternary embedding over the binary is significant. Third, the side-informed schemes achieve significantly higher security despite making more embedding changes (Fig. 5.2.2) and embedding into smooth areas (Fig. 5.2.3). This means that the security of SI schemes solely hinges upon the difficulty of estimating the rounding errors from the quantized (and embedded) image. Finally, Table 5.1 shows the effect of the resizing kernel in Matlab's 'imresize' for SI-HILL and SI-S-UNIWARD at 0.4 bpp. The filter 'nearest' is missing as it does not produce any rounding error.

Color conversion

$$T : (\text{RAW})^{\text{FULL}} \longrightarrow (\text{RGB})_{8\text{B}}^{\text{FULL}} \xrightarrow{\text{crop, resize}} (\text{RGB})_{8\text{B}}^{512^2} \xrightarrow{\text{gray}} (\text{GRAY})_{\text{DBL}}^{512^2}$$

In Fig. 5.2.4, we compare HILL and S-UNIWARD and their side-informed variants when utilizing

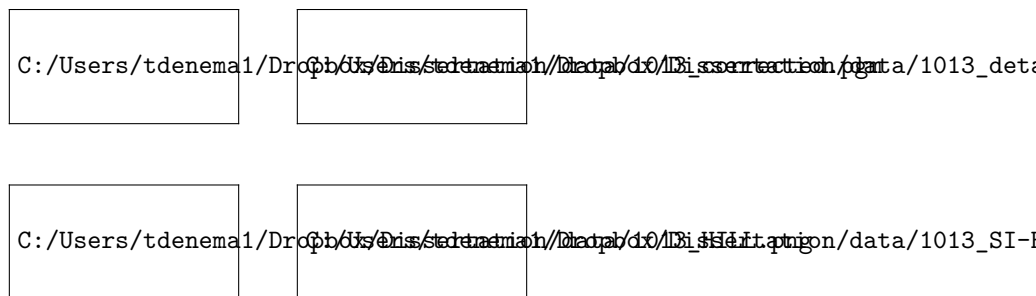


Figure 5.2.2: By rows: cover image, its detail, embedding changes for HILL and SI3-HILL-U at 0.4 bpp for resizing with Lanczos 3 kernel.

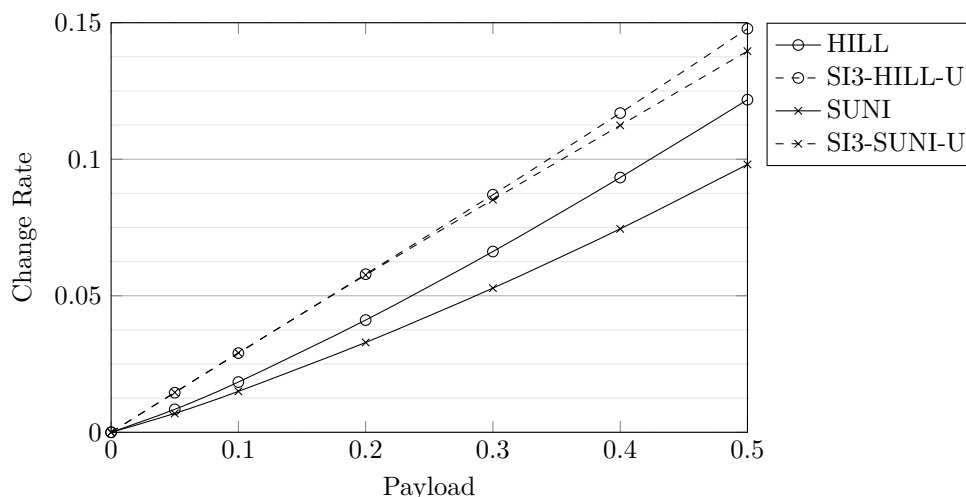


Figure 5.2.3: Change rate for HILL and S-UNIWARD and their SI versions when resizing with Lanczos 3 kernel. The values are averages over all 10,000 images in our source.

the color conversion rounding error. The images are first resized so that the smaller side is 512 pixels and then cropped to square. The conversion used a linear combination of individual RGB channels with coefficients [0.2989, 0.5870, 0.1140]. This experiment again shows a strong positive influence of the ternary embedding operation, increasing E_{OOB} by almost 10%.

Quantization

$$T : (\text{RAW})_{\text{FULL}}^{\text{FULL}} \rightarrow (\text{RGB})_{\text{16B}}^{\text{FULL}} \xrightarrow{\text{crop}} (\text{RGB})_{\text{16B}}^{512^2} \xrightarrow{\text{gray}} (\text{GRAY})_{\text{DBL}}^{512^2}$$

In this case, the transformation T does not include resizing, which makes the cover images smoother than in the previous experiments. Thus, the steganographic algorithms adapt to the acquisition noise naturally present in the image rather than the content. It is rather interesting that for this source HILL and S-UNIWARD have almost identical performance (Fig. 5.2.5). The gain of ternary embedding is mostly pronounced for medium payloads.

5.2.2 JPEG domain

The precover source for all experiments in this subsection was the BOSSbase 1.01 database with images compressed using Matlab’s ‘imwrite’ with 75% quality factor. Here, the unquantized cover \mathbf{U} are the non-rounded DCT coefficients divided by the corresponding quantization steps.

Fig. 5.2.6 shows the security of J-UNIWARD and its side-informed variants. Note that the SI embed-

	bilinear	bicubic	box	triangle	cubic	Lanczos2	Lanczos3
HILL	0.0978	0.1564	0.2038	0.0981	0.1543	0.1539	0.1684
SI3-HILL-U	0.1731	0.2411	0.2899	0.1738	0.2421	0.2462	0.2652
SUNI	0.0646	0.1092	0.1249	0.0651	0.1072	0.1081	0.1233
SI3-SUNI-U	0.1261	0.1941	0.2562	0.1262	0.1916	0.1935	0.2061

Table 5.1: Mean E_{OOB} for HILL, S-UNIWARD and their side-informed variants when utilizing the quantization error after resizing with different kernels at 0.4 bpp.

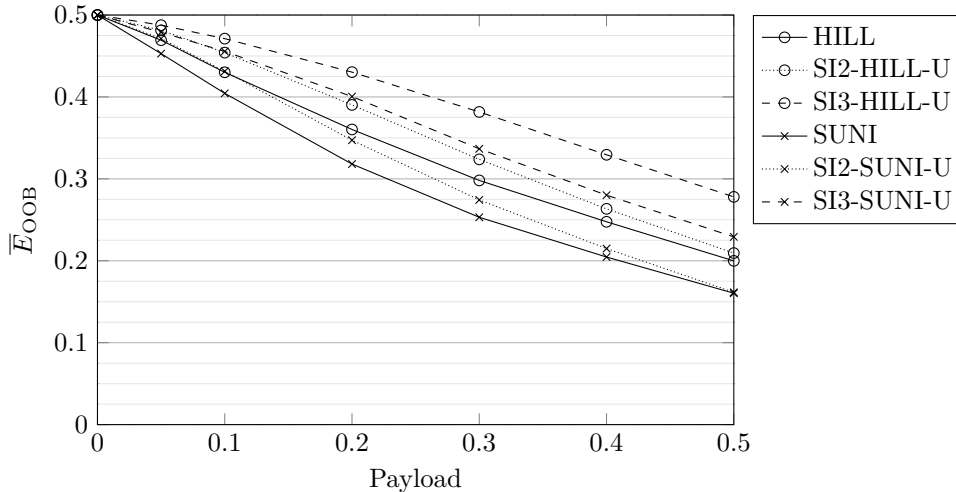


Figure 5.2.4: Mean E_{OOB} for HILL, S-UNIWARD and their SI versions with the quantization error after RGB to grayscale conversion as the side-information.

ding in JPEGs exhibits very different properties than in the spatial domain. The difference between the costs $\rho_{ij}(\mathbf{U})$ and $\rho_{ij}(\mathbf{X})$ is now more influential while the choice of the embedding operation (binary vs. ternary) is small with the ternary embedding giving a slightly worse performance. Both observations can be attributed to the much larger quantization step. Indeed, the harsher quantization removes more information about the cover source, which makes the costs computed from the unquantized cover more tightly related to detectability. On the other hand, the larger quantization step makes the cost of ternary embedding also higher than in the spatial domain. Finally, note that the actual implementation of SI-UNIWARD [51] corresponds to SI2-JUNI-U (Appendix A).

5.3 Conclusions

Side-informed steganography has been studied in the past but was limited only to JPEG and palette images. In this chapter, we formalize a general principle for utilizing side-information in any steganographic scheme that minimizes distortion, further enhance security by allowing ternary embedding, and investigate the impact of computing embedding costs from quantized and unquantized covers. The investigation is experimental and carried out for resizing, color depth reduction, and color to grayscale conversion in the spatial domain and for quantization during JPEG compression. The gain in empirical security and the effect of the proposed measures appears to depend mainly on the ratio of the quantization step used for the final quantization and the image dynamic range. In the spatial domain, this ratio is small, which makes the effect of computing the costs from the unquantized cover rather than the quantized cover negligible. This is also because the selection channel of modern spatial domain embedding schemes is insensitive to small perturbations. On the other hand, the

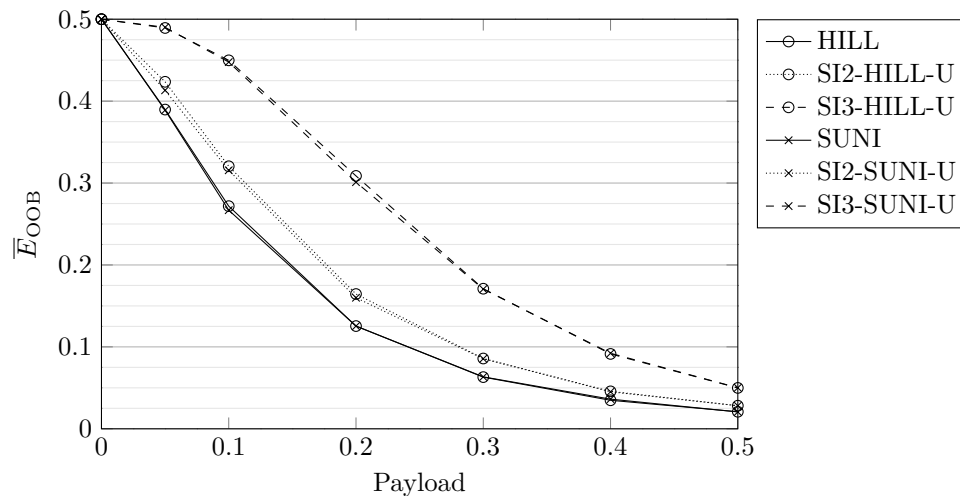


Figure 5.2.5: Mean E_{OOB} for SRM for HILL, S-UNIWARD and their SI versions with the quantization error after color depth reduction as the side-information.

effect of allowing a ternary embedding operation is quite significant because rounding “to both sides” is less expensive due to the fine quantization step. The situation is exactly the opposite for JPEG images. There, the ternary embedding does not bring any improvement due to the large amplitude of embedding changes while the costs computed from the unquantized precover give better security as more information about the precover source is lost due to the much harsher quantization.

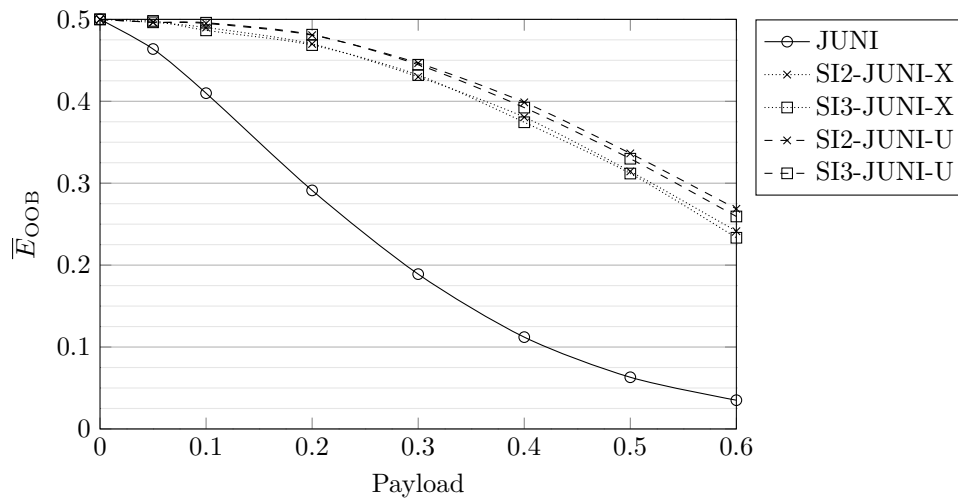


Figure 5.2.6: Mean E_{OOB} for J-UNIWARD and its side-informed variants when utilizing the quantization error after JPEG compression when computing the costs from the unquantized cover and the cover. The used quality factor is 75%.

Chapter 6

Model based side-informed steganography

Digital media files are extremely complex objects that are notoriously hard to describe with sufficiently accurate and estimable statistical models. This is the main reason for why current steganography in such empirical sources [9] lacks perfect security and heavily relies on heuristics, such as embedding “costs” and intuitive modulation factors. Similarly, practical steganalysis resorts to increasingly more complex high-dimensional descriptors (rich models) and advanced machine learning paradigms, including ensemble classifiers and deep learning.

Despite the success of side-informed schemes, there appears to be an alarming lack of theoretical analysis that would either justify the heuristics or suggest a well-founded (and hopefully more powerful) approach. In [35], the author has shown that the precover compensates for the lack of the cover model. In particular, for a Gaussian model of acquisition noise, precover-informed rounding is more secure than embedding designed to preserve the cover model estimated from the precover image assuming the cover is “sufficiently non-stationary.” The Natural Steganography approach discussed in Chapter 8 is also worth mentioning.

Inspired by the success of the multivariate Gaussian model in steganography for digital images [82, 40, 84], in this chapter we adopt the same model for the precover and then derive the embedding rule to minimize the KL divergence between cover and stego distributions. The side-information is used to estimate the parameters of the acquisition noise and the noise-free scene.

6.1 Modeling acquisition

An image \mathbf{P} acquired using an imaging sensor has two components – the true scene \mathbf{T} and acquisition imperfections (noise) \mathbf{N} :

$$\mathbf{P} = \mathbf{T} + \mathbf{N}(\mathbf{T}, \boldsymbol{\theta}). \quad (6.1.1)$$

While the scene \mathbf{T} is deterministic, the acquisition noise \mathbf{N} is stochastic in nature. It depends on \mathbf{T} because, for example, the variance of the photonic (shot) noise linearly depends on light intensity (the so-called heteroscedastic noise model [30]). Other random components of \mathbf{N} , including readout and electronic noise depend on the particular sensor. We exclude from \mathbf{N} imperfections that are consistent from picture to picture (of the same scene under the same conditions and camera settings), which include the photo-response non-uniformity and dark current. Demosaicking, color correction, and additional filtering applied to the acquired image either in the camera or during post-production introduce dependencies into spatially neighboring samples of \mathbf{N} , turning it into a random field parametrized by $\boldsymbol{\theta}$, a vector encompassing the properties of the imaging hardware, camera settings,

as well as the processing pipeline. Note that following the notation from (5.0.1), the uncompressed cover \mathbf{U} and precover \mathbf{P} are identical.

Fundamentally, the steganographic capacity of \mathbf{P} is the entropy $H(\mathbf{N})$. However, embedding a payload of this size undetectably is generally possible only in very special cases when the processing pipeline is drastically simplified. Recently, Bas [3] has showed that RAW images acquired using a monochrome sensor (a sensor not equipped with a color filter array) with no spatial filtering applied to them can carry a rather large payload with a very low level of empirical detectability. The method is called “steganography by cover source switching” because the stego image statistically resembles an image acquired with a higher sensor gain setting (ISO). In general, however, constructing a steganographic method capable of embedding $H(\mathbf{N})$ bits is likely infeasible in virtually all practical situations because of the daunting complexity and non-stationarity of the random field \mathbf{N} . More seriously, many elements of the processing pipeline are unknown to the sender as most digital camera manufacturers use proprietary demosaicking algorithms with local content-dependent rules for interpolating the missing colors as well as proprietary content-adaptive algorithms for denoising and sharpening.

6.2 Side-informed steganography with multivariate Gaussian acquisition noise

It is intuitively clear that incorporating even partial information about the noise-free scene \mathbf{T} and the statistical properties of \mathbf{N} at the sender should improve security. Indeed, the sender has a fundamental advantage because, in contrast to the Warden, she may have access to the acquisition oracle (the digital camera) as well as the scene being imaged. For example, she can utilize the RAW sensor capture or multiple acquisitions of the same scene or even “manufacture” the side-information prior to embedding by subjecting the precover image to an information-reducing operation, such as quantization, downsampling, format conversion, and/or reduction of the dynamic range.

In this chapter, we adopt a tractable model of the acquisition noise estimated from the available side-information and derive the embedding rule by minimizing the KL divergence between cover and stego images. The precover is modeled as an array of $n = n_1 \times n_2$ independent but not necessarily identically distributed random variables P , $i = 1, \dots, n_1$, $j = 1 \dots, n_2$. The error introduced by color interpolation is also included in the acquisition noise since we view demosaicking as part of acquisition. Therefore, p_{ij} will naturally have a larger variance in textured and noisy regions where color interpolation algorithms are less accurate. Since a commonly adopted model of acquisition noise inherent to the sensor is the Gaussian distribution (the heteroscedastic model [30, 89]), we adopt this model in this chapter as well:

$$p_{ij} \sim \mathcal{N}(t_{ij}, \sigma_{ij}^2), \quad (6.2.1)$$

where t_{ij} is the precover value that would be registered by the sensor in the absence of acquisition noise. Finally, we assume that the ij th cover element x_{ij} is obtained by rounding a specific realization of p_{ij} . This is a simplification because in practice p_{ij} will be constrained to a finite dynamic range. Note that the rounding encompasses more general uniform scalar quantizers.

We constrain ourselves to the simplest case when the sender has one precover p_{ij} , which will be used to obtain an estimate of the true scene, \hat{t}_{ij} , as well as the acquisition noise variance, $\hat{\sigma}_{ij}$. We note that the methodology proposed here can be extended to more general types of side-information, such as multiple precovers / covers.

The embedding operation considered is binary, meaning that each precover element p_{ij} will be either rounded to $x_{ij} = [p_{ij}]$ or to $\bar{x}_{ij} = [p_{ij}] + \text{sign}(p_{ij} - [p_{ij}])$ with the convention that when p_{ij} is an integer, $\bar{x}_{ij} \in \{x_{ij} - 1, x_{ij} + 1\}$ is chosen uniformly randomly.

Denoting the stego image elements with y_{ij} , the embedding will change x_{ij} to x_{ij} with probability β_{ij} :

$$\Pr\{y_{ij} = x_{ij}\} = 1 - \beta_{ij} \quad (6.2.2)$$

$$\Pr\{y_{ij} = \bar{x}_{ij}\} = \beta_{ij}, \quad (6.2.3)$$

effectively embedding $h_2(\beta_{ij})$ nats, where $h_2(x) = -x \ln x - (1-x) \ln(1-x)$ is the binary entropy function.

Denoting the closed interval $\mathcal{I}_{ij} = [x_{ij}, \bar{x}_{ij}]$ when $\bar{x}_{ij} > x_{ij}$ and $\mathcal{I}_{ij} = [\bar{x}_{ij}, x_{ij}]$ when $\bar{x}_{ij} \leq x_{ij}$, the embedding is designed to minimize the KL divergence between cover and stego distributions conditioned on precover values $p_{ij} \in \mathcal{I}_{ij}$. WLOG, in what follows we will assume that $\bar{x}_{ij} > x_{ij}$. This conditional probability for the cover $x_{ij} = [p_{ij}]$ is

$$\Pr\{[p_{ij}] = x_{ij} | p_{ij} \in \mathcal{I}_{ij}\} = \frac{f_{ij}(x_{ij})}{f_{ij}(x_{ij}) + f_{ij}(\bar{x}_{ij})} \quad (6.2.4)$$

$$\Pr\{[p_{ij}] = \bar{x}_{ij} | p_{ij} \in \mathcal{I}_{ij}\} = \frac{f_{ij}(\bar{x}_{ij})}{f_{ij}(x_{ij}) + f_{ij}(\bar{x}_{ij})} \quad (6.2.5)$$

where

$$f_{ij}(x_{ij}) = Q\left(\frac{x_{ij} - \hat{t}_{ij}}{\hat{\sigma}_{ij}}\right) - Q\left(\frac{x_{ij} + 1/2 - \hat{t}_{ij}}{\hat{\sigma}_{ij}}\right) \quad (6.2.6)$$

$$f_{ij}(\bar{x}_{ij}) = Q\left(\frac{x_{ij} + 1/2 - \hat{t}_{ij}}{\hat{\sigma}_{ij}}\right) - Q\left(\frac{x_{ij} + 1 - \hat{t}_{ij}}{\hat{\sigma}_{ij}}\right) \quad (6.2.7)$$

with $Q(x)$ the tail probability of a standard normal random variable $\mathcal{N}(0, 1)$.

The ij th pixel in the stego image is modeled as a discrete random variable y_{ij} with range $\{x_{ij}, \bar{x}_{ij}\}$ with pmf

$$\Pr\{y_{ij} = x_{ij} | p_{ij} \in \mathcal{I}_{ij}\} = \frac{g_{ij}(x_{ij})}{g_{ij}(x_{ij}) + g_{ij}(\bar{x}_{ij})}, \quad (6.2.8)$$

$$\Pr\{y_{ij} = \bar{x}_{ij} | p_{ij} \in \mathcal{I}_{ij}\} = \frac{g_{ij}(\bar{x}_{ij})}{g_{ij}(x_{ij}) + g_{ij}(\bar{x}_{ij})}, \quad (6.2.9)$$

where

$$g_{ij}(x_{ij}) = (1 - \beta_{ij}^{(SI)})f_{ij}(x_{ij}) + \beta_{ij}^{(SI)}f_{ij}(\bar{x}_{ij}) \quad (6.2.10)$$

$$g_{ij}(\bar{x}_{ij}) = \beta_{ij}^{(SI)}f_{ij}(x_{ij}) + (1 - \beta_{ij}^{(SI)})f_{ij}(\bar{x}_{ij}). \quad (6.2.11)$$

Denoting for compactness $h_{ij}^L = f_{ij}(x_{ij})$ and $h_{ij}^R = f_{ij}(\bar{x}_{ij})$, a straightforward derivation gives the

following KL divergence between ij th cover and stego elements conditioned on $p_{ij} \in \mathcal{I}_{ij}$:

$$d_{ij} = D_{\text{KL}} \left(x_{ij} \parallel y_{ij} \mid p_{ij} \in \mathcal{I}_{ij} \right) \quad (6.2.12)$$

$$= \frac{h_{ij}^{\text{L}}}{h_{ij}^{\text{L}} + h_{ij}^{\text{R}}} \log \frac{h_{ij}^{\text{L}}}{(1 - \beta_{ij}^{(\text{SI})})h_{ij}^{\text{L}} + \beta_{ij}^{(\text{SI})}h_{ij}^{\text{R}}} \quad (6.2.13)$$

$$+ \frac{h_{ij}^{\text{R}}}{h_{ij}^{\text{L}} + h_{ij}^{\text{R}}} \log \frac{h_{ij}^{\text{R}}}{\beta_{ij}^{(\text{SI})}h_{ij}^{\text{L}} + (1 - \beta_{ij}^{(\text{SI})})h_{ij}^{\text{R}}} \quad (6.2.14)$$

$$= -\frac{h_{ij}^{\text{L}}}{h_{ij}^{\text{L}} + h_{ij}^{\text{R}}} \log \left(1 - \beta_{ij}^{(\text{SI})} + \beta_{ij}^{(\text{SI})} \frac{h_{ij}^{\text{R}}}{h_{ij}^{\text{L}}} \right) \quad (6.2.15)$$

$$- \frac{h_{ij}^{\text{R}}}{h_{ij}^{\text{L}} + h_{ij}^{\text{R}}} \log \left(1 - \beta_{ij}^{(\text{SI})} + \beta_{ij}^{(\text{SI})} \frac{h_{ij}^{\text{L}}}{h_{ij}^{\text{R}}} \right) \quad (6.2.16)$$

$$\doteq \frac{\left(\beta_{ij}^{(\text{SI})} \right)^2}{2(h_{ij}^{\text{L}} + h_{ij}^{\text{R}})} \times \left(h_{ij}^{\text{L}} \left(1 - \frac{h_{ij}^{\text{R}}}{h_{ij}^{\text{L}}} \right)^2 + h_{ij}^{\text{R}} \left(1 - \frac{h_{ij}^{\text{L}}}{h_{ij}^{\text{R}}} \right)^2 \right) \quad (6.2.17)$$

$$= \frac{\left(\beta_{ij}^{(\text{SI})} \right)^2}{2} \frac{(h_{ij}^{\text{L}} - h_{ij}^{\text{R}})^2}{h_{ij}^{\text{L}} + h_{ij}^{\text{R}}} \left(\frac{1}{h_{ij}^{\text{L}}} + \frac{1}{h_{ij}^{\text{R}}} \right) \quad (6.2.18)$$

$$\doteq \frac{1}{2} \left(\beta_{ij}^{(\text{SI})} \right)^2 I_{ij}^{(\text{SI})}, \quad (6.2.19)$$

where

$$I_{ij}^{(\text{SI})} = \frac{(h_{ij}^{\text{L}} - h_{ij}^{\text{R}})^2}{h_{ij}^{\text{L}} h_{ij}^{\text{R}}} \quad (6.2.20)$$

is the Fisher information obtained by expanding the logarithms using Taylor series w.r.t. $\beta^{(\text{SI})}$ at $\beta^{(\text{SI})} = 0$ and keeping only the leading term.

The total KL divergence between the cover and stego objects $\mathbf{X} = (x_{ij})$, $\mathbf{Y} = (y_{ij})$, $i = 1, \dots, n_1$, $j = 1 \dots, n_2$, is the sum

$$D_{\text{KL}}(\mathbf{X} \parallel \mathbf{Y}) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} d_{ij} \doteq \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \frac{1}{2} \beta_{ij}^{(\text{SI})} I_{ij}^{(\text{SI})}. \quad (6.2.21)$$

The actual embedding change rates $\beta_{ij}^{(\text{SI})}$ are determined by minimizing $D_{\text{KL}}(\mathbf{X} \parallel \mathbf{Y})$ under the payload constraint

$$\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} h_2(\beta_{ij}^{(\text{SI})}) = \alpha n, \quad (6.2.22)$$

where α is expressed in nats per pixel. Similar to MiPOD [82], the proposed scheme minimizes the sum of pixels' Fisher information weighted by squared change rates as in this case the embedding "cost" relates to statistical detectability. Note that minimizing the KL divergence makes the design optimal only against an omniscient Warden who knows the exact actions of the embedder, including the rounding errors e_{ij} . When the problem of embedding and detection are formulated withing a game-theoretical framework when both the sender and the Warden randomize their strategies of how to distribute (detect) the payload, the sender should also minimize a weighted sum of squared change rates to operate at the Nash equilibrium of a zero-sum game when the payoff is defined as Warden's test power for a bounded false alarm [60].

The optimization problem can be approached in a standard manner using the method of Lagrange

multipliers by solving the equations $\partial L / \partial \beta_{ij}^{(\text{SI})} = 0$, where

$$L = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \frac{1}{2} \left(\beta_{ij}^{(\text{SI})} \right)^2 I_{ij}^{(\text{SI})} - \lambda \left(h_2(\beta_{ij}^{(\text{SI})}) - \alpha n \right). \quad (6.2.23)$$

This leads to n non-linear equations for $\beta_{ij}^{(\text{SI})}$

$$\beta_{ij}^{(\text{SI})} I_{ij}^{(\text{SI})} = \lambda^{(\text{SI})} \log \frac{1 - \beta_{ij}^{(\text{SI})}}{\beta_{ij}^{(\text{SI})}}, \quad (6.2.24)$$

for each ij , which can be solved numerically, for example, by a binary search over $\lambda^{(\text{SI})} \in [0, \infty]$ in a similar manner as described in [40]. This embedding algorithm will be called side-informed MiPOD (SI-MiPOD).

6.2.1 Extension to JPEG domain

The proposed scheme is extended to work with JPEG images in a straightforward manner. The independence imposed on the noise \mathbf{N} and linearity of the DCT allows us to easily compute the variance of each non-rounded DCT coefficient given the estimated variances of pixels $\hat{\sigma}_{ij}$. Thus, the same MVG model of acquisition noise can be applied to non-rounded DCT coefficients.

Given an 8×8 block of pixel values $p_{ij} \in \{0, \dots, 255\}$, $0 \leq i, j \leq 7$, in an uncompressed image \mathbf{P} , the kl th DCT coefficient D_{kl} , $0 \leq k, l \leq 7$, before rounding is a linear combination of pixel values

$$U_{kl} = 1/Q_{kl} \sum_{i,j=0}^7 f_{ij}^{(k,l)} p_{ij}, \quad (6.2.25)$$

or in matrix form symbolically $\mathbf{U} = \text{DCT}(\mathbf{P})$, where

$$f_{ij}^{(k,l)} = \frac{w_k w_l}{4} \cos \frac{\pi k(2i+1)}{16} \cos \frac{\pi l(2j+1)}{16}, \quad (6.2.26)$$

$w_0 = 1/\sqrt{2}$, $w_k = 1$ for $k > 0$, and Q_{kl} is the JPEG quantization matrix. With the adopted acquisition model (6.2.1), the unrounded DCT coefficients u_{kl} are independent samples from an array of Gaussian variables $\mathcal{N}(\mu_{kl}, \sigma_{kl}^2)$. The parameters of the Gaussian acquisition noise can be estimated as

$$\hat{\sigma}_{kl}^2 = 1/Q_{kl}^2 \sum_{i,j=0}^7 \left(f_{ij}^{(k,l)} \right)^2 \hat{\sigma}_{ij}^2, \quad (6.2.27)$$

where $\hat{\sigma}_{ij}^2$ are pixel variances estimated in the spatial domain using, e.g., the variance estimator used in MiPOD. With a single precover, the mean is estimated as

$$\hat{\mu}_{kl} = u_{kl}. \quad (6.2.28)$$

Having estimated the MVG parameters, the proposed side-informed embedding rounds u_{kl} to either $[u_{kl}]$ or $[u_{kl}] + \text{sign}(e_{kl})$, $e_{kl} = u_{kl} - [u_{kl}]$ as described above. This embedding algorithm will be called side-informed J-MiPOD (SI-J-MiPOD).

6.3 Connection to heuristic schemes

We now contrast the derived embedding rule with heuristic binary side-informed embedding with precover (see Eq. (5.0.4)). When only the precover value p_{ij} is available as side-information, $\hat{t}_{ij} = p_{ij}$ is a minimum variance unbiased estimate under our acquisition model. To obtain a closed-form expression, we work in the limit of small and large values of $\hat{\sigma}_{ij}$ for both schemes.

6.3.1 Model-based SI-MiPOD

Without loss on generality, we will assume that $e_{ij} \geq 0$. Recalling the formula for f_{ij} (6.2.6) and Taylor expansion of $Q(x) \approx \frac{1}{2} - \frac{1}{\sqrt{2\pi}}x + \frac{1}{6\sqrt{2\pi}}x^3 + \mathcal{O}(x^5)$ at $x = 0$, for large $\hat{\sigma}_{ij}$

$$\begin{aligned} h_{ij}^L &= Q\left(\frac{[p_{ij}] - p_{ij}}{\hat{\sigma}_{ij}}\right) - Q\left(\frac{[p_{ij}] + 1/2 - p_{ij}}{\hat{\sigma}_{ij}}\right), \\ &= Q\left(\frac{-e_{ij}}{\hat{\sigma}_{ij}}\right) - Q\left(\frac{-e_{ij} + 1/2}{\hat{\sigma}_{ij}}\right) \end{aligned} \quad (6.3.1)$$

$$\doteq \frac{1}{2\hat{\sigma}_{ij}\sqrt{2\pi}} - \frac{1}{6\hat{\sigma}_{ij}^3\sqrt{2\pi}} \left(\frac{3}{2}e_{ij}^2 - \frac{3}{4}e_{ij} + \frac{1}{8} \right) \quad (6.3.2)$$

and

$$\begin{aligned} h_{ij}^R &= Q\left(\frac{[p_{ij}] + 1/2 - P_{ij}}{\hat{\sigma}_{ij}}\right) - Q\left(\frac{[p_{ij}] + 1 - p_{ij}}{\hat{\sigma}_{ij}}\right), \\ &= Q\left(\frac{-e_{ij} + 1/2}{\hat{\sigma}_{ij}}\right) - Q\left(\frac{-e_{ij} + 1}{\hat{\sigma}_{ij}}\right) \end{aligned} \quad (6.3.3)$$

$$\doteq \frac{1}{2\hat{\sigma}_{ij}\sqrt{2\pi}} - \frac{1}{6\hat{\sigma}_{ij}^3\sqrt{2\pi}} \left(\frac{3}{2}e_{ij}^2 - \frac{9}{4}e_{ij} + \frac{7}{8} \right). \quad (6.3.4)$$

Thus,

$$(h_{ij}^L - h_{ij}^R)^2 \doteq \frac{(1/2 - e_{ij})^2}{16\hat{\sigma}_{ij}^6 2\pi} \quad (6.3.5)$$

and

$$1/h_{ij}^L \doteq 1/h_{ij}^R \doteq 2\hat{\sigma}_{ij}\sqrt{2\pi} \quad (6.3.6)$$

for large $\hat{\sigma}_{ij}$. Finally, the Fisher information (6.2.20) becomes

$$I_{ij}^{(SI)} \doteq \frac{1}{4\hat{\sigma}_{ij}^4} (1/2 - |e_{ij}|)^2. \quad (6.3.7)$$

This expression for the FI (6.3.7) has been written for the general case when e_{ij} can be both negative and positive. It informs us that, for large noise variance, the rounding error should modulate the Fisher information by $(1/2 - |e_{ij}|)^2$, which achieves a similar effect as multiplying the costs by $1 - 2|e_{ij}|$ in heuristic cost-based side-informed embedding schemes. We remark that the approximation (6.3.7) is fairly accurate as long as $\hat{\sigma}_{ij} \gtrsim 2$ for all values of e_{ij} .

For a small $\hat{\sigma}_{ij}$, the terms h_{ij}^L and h_{ij}^R in (6.2.20) can be simplified using the asymptotic expression for $Q(x) \approx \frac{1}{\sqrt{2\pi x}} e^{-x^2/2}$ for large x :

$$h_{ij}^L = Q\left(\frac{[p_{ij}] - p_{ij}}{\hat{\sigma}_{ij}}\right) - Q\left(\frac{[p_{ij}] + 1/2 - p_{ij}}{\hat{\sigma}_{ij}}\right), \quad (6.3.8)$$

$$\doteq 1 - \frac{\exp(-(-e_{ij})^2/(2\hat{\sigma}_{ij}^2))}{\sqrt{2\pi}(e_{ij}/\hat{\sigma}_{ij})} - \frac{\exp(-(1/2 - e_{ij})^2/(2\hat{\sigma}_{ij}^2))}{\sqrt{2\pi}((1/2 - e_{ij})/\hat{\sigma}_{ij})}, \quad (6.3.9)$$

$$\doteq 1. \quad (6.3.10)$$

Following the same steps,

$$h_{ij}^R \doteq \frac{\exp(-(1/2 - e_{ij})^2/(2\hat{\sigma}_{ij}^2))}{\sqrt{2\pi}(1/2 - e_{ij})/\hat{\sigma}_{ij}}, \quad (6.3.11)$$

and finally

$$I_{ij}^{(\text{SI})} = \frac{(h_{ij}^{\text{R}} - h_{ij}^{\text{L}})^2}{h_{ij}^{\text{L}} h_{ij}^{\text{R}}} \doteq \frac{1}{h_{ij}^{\text{R}}} \quad (6.3.12)$$

$$\doteq \frac{1}{\sqrt{2\pi}} \frac{1/2 - e_{ij}}{\hat{\sigma}_{ij}} \exp\left((1/2 - e_{ij})^2 / (2\hat{\sigma}_{ij}^2)\right). \quad (6.3.13)$$

For small $\hat{\sigma}_{ij}$, we will derive an approximate solution to (6.2.24) by taking its logarithm

$$\ln \beta_{ij}^{(\text{SI})} + \ln I_{ij}^{(\text{SI})} \doteq \ln \lambda^{(\text{SI})} + \ln\left(-\ln\left(\beta_{ij}^{(\text{SI})}\right)\right), \quad (6.3.14)$$

from which $\beta_{ij}^{(\text{SI})} \doteq \lambda^{(\text{SI})} / I_{ij}^{(\text{SI})}$ and thus

$$\beta_{ij}^{(\text{SI})} \doteq \frac{\lambda^{(\text{SI})}}{I_{ij}^{(\text{SI})}} = \frac{1}{\sqrt{2\pi}} \frac{\hat{\sigma}_{ij}}{1/2 - e_{ij}} \exp\left(-(1/2 - e_{ij})^2 / (2\hat{\sigma}_{ij}^2)\right). \quad (6.3.15)$$

When $\hat{\sigma}_{ij} \ll 1/2 - e_{ij}$, the value of $\beta_{ij}^{(\text{SI})}$ will rapidly approach zero.

6.3.2 Heuristic SI-MiPOD

To obtain closed-form expressions, we will again consider the case of a small $\hat{\sigma}_{ij}$ and large $\hat{\sigma}_{ij}$. The heuristic side-informed MiPOD starts with MiPOD's embedding costs ρ_{ij} derived in (1.3.6), which are then modulated by the rounding error $\rho_{ij}^{(\text{SI})} = \rho_{ij} (1 - 2|e_{ij}|)$. The side-informed embedding probabilities are

$$\beta_{ij}^{(\text{SI})} = \frac{e^{-\lambda^{(\text{SI})} \rho_{ij}^{(\text{SI})}}}{1 + 2e^{-\lambda^{(\text{SI})} \rho_{ij}^{(\text{SI})}}} = \frac{1}{2 + e^{\lambda^{(\text{SI})} \rho_{ij}^{(\text{SI})}}}, \quad (6.3.16)$$

where $\lambda^{(\text{SI})}$ is a Lagrange multiplier determined from the payload constraint $\sum_{ij} h_2(\beta_{ij}^{(\text{SI})}) = \alpha n$. The embedding can be reinterpreted as MiPOD with Fisher information

$$\begin{aligned} I_{ij}^{(\text{SI})} &= \frac{\lambda^{(\text{SI})}}{\beta_{ij}^{(\text{SI})}} \ln\left(1/\beta_{ij}^{(\text{SI})} - 2\right), \\ &= \lambda^{(\text{SI})} (2 + e^{\lambda^{(\text{SI})} \rho_{ij}^{(\text{SI})}}) \lambda^{(\text{SI})} \rho_{ij}^{(\text{SI})}. \end{aligned} \quad (6.3.17)$$

First, we carry out the analysis for large $\hat{\sigma}_{ij}$. To this end, we rewrite Eq. (1.3.5) determining MiPOD's embedding change rates as

$$\exp(\beta_{ij} I_{ij} / \lambda) = 1/\beta_{ij} - 2. \quad (6.3.18)$$

Recalling that $I_{ij} = 2/\hat{\sigma}_{ij}^4 \ll 1$ for large $\hat{\sigma}_{ij}$, the left side can be approximated using Taylor expansion as

$$1 + \frac{\beta_{ij} I_{ij}}{\lambda} + \frac{\beta_{ij}^2 I_{ij}^2}{2\lambda^2} = 1/\beta_{ij} - 2 \quad (6.3.19)$$

which is equivalent to

$$\frac{1}{2\lambda^2} y_{ij}^3 + \frac{1}{\lambda} y_{ij}^2 + 3y_{ij} = I_{ij} \quad (6.3.20)$$

where $y_{ij} = I_{ij} \beta_{ij}$. For small I_{ij} , the first-order solution to this cubic equation is $y_{ij}^{(1)} \doteq \frac{I_{ij}}{3}$ and the second-order solution

$$y_{ij}^{(2)} = \frac{I_{ij}}{3} - \frac{1}{3\lambda} \left(\frac{I_{ij}}{3}\right)^2, \quad (6.3.21)$$

which gives us $\beta_{ij} \doteq \frac{1}{3} - \frac{I_{ij}}{27\lambda}$. Substituting this approximation to Eq. (1.3.6) and keeping only the leading term gives $\rho_{ij} \doteq \frac{I_{ij}}{3\lambda}$ and for the Fisher information (6.3.17)

$$\begin{aligned} I_{ij}^{(\text{SI})} &\propto \rho_{ij}^{(\text{SI})} = \rho_{ij}(1 - 2|e_{ij}|) \\ &\propto I_{ij}(1 - 2|e_{ij}|). \end{aligned} \quad (6.3.22)$$

For small $\hat{\sigma}_{ij}$, I_{ij} will be large and $\beta_{ij} \doteq \lambda/I_{ij}$ small, as derived in the previous section, and therefore, $\rho_{ij} \approx \ln(I_{ij}/\lambda - 2)$. Substituting this result into (6.3.16) and recalling that $I_{ij} = 2\hat{\sigma}_{ij}^{-4}$:

$$\begin{aligned} \beta_{ij}^{(\text{SI})} &= \frac{1}{2 + \exp(\lambda^{(\text{SI})}\rho_{ij}^{(\text{SI})})} \\ &\doteq \exp(-\lambda^{(\text{SI})}\rho_{ij}^{(\text{SI})}) = \exp(-\lambda^{(\text{SI})}\rho_{ij}(1 - 2|e_{ij}|)) \\ &\doteq \exp\left(-\lambda^{(\text{SI})}\ln(I_{ij}/\lambda - 2)(1 - 2|e_{ij}|)\right) \\ &= C(\lambda, e_{ij})\hat{\sigma}_{ij}^{4\lambda^{(\text{SI})}(1-2|e_{ij}|)}, \end{aligned} \quad (6.3.23)$$

where $C(\lambda, e_{ij})$ does not depend on $\hat{\sigma}_{ij}$.

6.3.3 Comparison of model-based and heuristic MiPOD

Let's compare the properties of heuristic SI-MiPOD and model-based SI-MiPOD using the above approximations and experimentally.

For $\hat{\sigma}_{ij} \gtrsim 2$, the Fisher information in model-based SI-MiPOD is modulated by $(1 - 2|e_{ij}|)^2$ (6.3.7) while in the heuristic SI-MiPOD by $(1 - 2|e_{ij}|)$ (6.3.22). Therefore, model-based SI-MiPOD embeds a higher payload in pixels with large noise variance.

For small $\hat{\sigma}_{ij}$, the embedding probabilities $\beta_{ij}^{(\text{SI})}$ for model-based SI-MiPOD rapidly approach zero for all values of e_{ij} (6.3.15), while for the heuristic SI-MiPOD the behavior of the probabilities $\beta_{ij}^{(\text{SI})}(\hat{\sigma}_{ij})$ depends on the exponent $4\lambda^{(\text{SI})}(1 - 2|e_{ij}|)$ (6.3.23). As $e_{ij} \rightarrow 1/2$, the exponent approaches zero and the embedding probability as a function of $\hat{\sigma}_{ij}$ should become concave. Summarizing both observations, we conclude that the model-based SI-MiPOD is more adaptive to the acquisition noise variance $\hat{\sigma}_{ij}^2$ than heuristic SI-MiPOD.

All quantitative conclusions reached above are now confirmed on an artificial precover P_{ij} with 5×64 pixels with all 5×64 combinations of values of five rounding errors $e_i \in \{0, 0.125, 0.25, 0.375, 0.495\}$ and 64 variances linearly spaced between 0.01 and 64. WLOG, we set the precover values to $P_{ij} = \hat{T}_{ij} = e_i$.

First, the embedding change rates β_{ij} were computed for payload α nats [82, 40], converted to costs ρ_{ij} using Eq. (1.3.6), and modulated $\rho_{ij}^{(\text{SI})} = (1 - 2e_i)\rho_{ij}$ as in Eq. (5.0.4). The modulated costs were then used to obtain the change rates via $\beta_{ij}^{(\text{SI})} = \exp(-\lambda^{(\text{SI})}\rho_{ij}^{(\text{SI})})/(1 + \exp(-\lambda^{(\text{SI})}\rho_{ij}^{(\text{SI})}))$ with $\lambda^{(\text{SI})}$ determined by the same payload α .

Figure 6.3.1 shows the embedding change probability $\beta_{ij}^{(\text{SI})}$ for $\alpha = 0.3$ nats as a function of the variance σ_j^2 for each e_i with the five curves corresponding to five values of e_i , $i = 1, \dots, 5$. The image on the left is for the heuristic binary SI-MiPOD (see [18] and Eq. (5.0.4)) while the image on the right corresponds to the model-based SI-MiPOD. The lines positioned lower correspond to lower e_i . Note that both schemes embed maximum possible entropy when $e \rightarrow 1/2$. In agreement with our analysis, for small variance, model-based SI-MiPOD is more conservative (embeds with smaller change rates) than heuristic SI-MiPOD. On the other hand, for large variance, the model-based version embeds with larger change rates. In other words, model-based SI-MiPOD is more content adaptive than its heuristic counterpart. While this may make the model-based approach

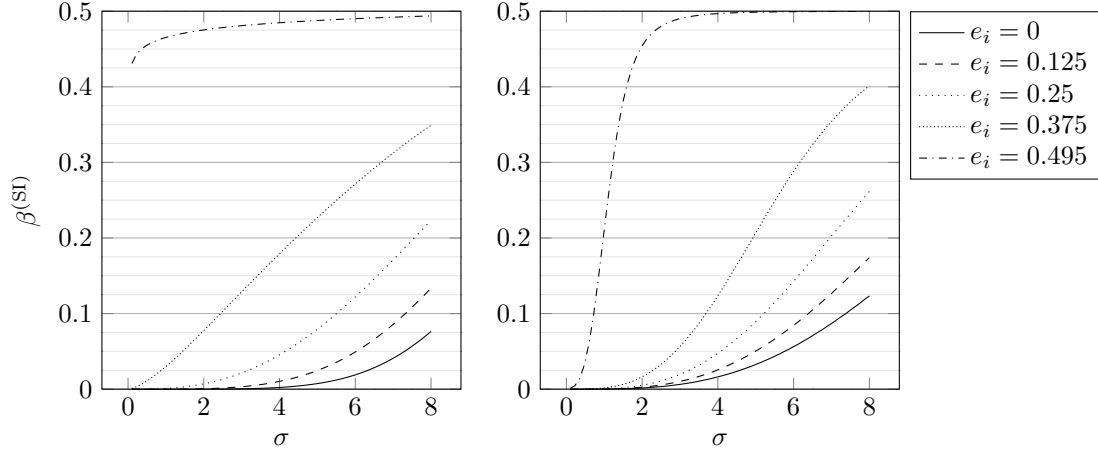


Figure 6.3.1: Embedding change probability $\beta^{(\text{SI})}$ as a function of variance σ^2 on a synthetic cover for $\alpha = 0.3$ nats using heuristic side-informed binary MiPOD (left) and model-based binary MiPOD (right).

more vulnerable to attacks utilizing the selection channel, such attacks are much more difficult to implement for the Warden because she does not have access to the rounding errors. More on this topic appears in Section 6.5.

6.4 Experiments

In this section, we provide the results and interpretation of all experiments in both spatial and JPEG domains. We start with the description of our image sources. To contrast the performance of heuristic side-informed schemes with the model-based versions, we include experiments on uncompressed as well as JPEG images.

6.4.1 Image sources

Two primary data sources will be used: BOSSbase 1.01 with 10,000 512×512 *color* images, called BOSSColor, and the same database of *grayscale* images, which will be addressed as BOSSbase. All images were all taken in the RAW format by seven different cameras, downsampled and cropped to the final size of 512×512 pixels. The script used for the conversion and processing is also available from the same web site as the database itself [1]. To create BOSSColor, we modified the script to skip the conversion from RGB to grayscale applied when creating BOSSbase.

6.4.2 Spatial domain (BOSSColor)

Figure 6.4.1 shows the results of our first experiment on real imagery. BOSSColor images were converted to grayscale using the formula $P = 0.2989R + 0.5870G + 0.1140B$, producing non-rounded precover values p_{ij} , which would then be rounded to 8bit integers $x_{ij} = \lfloor p_{ij} \rfloor$ to obtain the cover source of 8-bit grayscale 512×512 images. Estimates of pixels' noise variance $\hat{\sigma}_{ij}^2$ were obtained from precover grayscale values p_{ij} using the same variance estimator as in MiPOD (Section V in [82]). Because the selection channel depends on the rounding error e_{ij} , it is not clear how to utilize selection-channel-aware feature sets. Thus, we carry out steganalysis using the spatial rich model (SRM) [39] with the low-complexity linear classifier [14] as a classifier. The empirical detectability

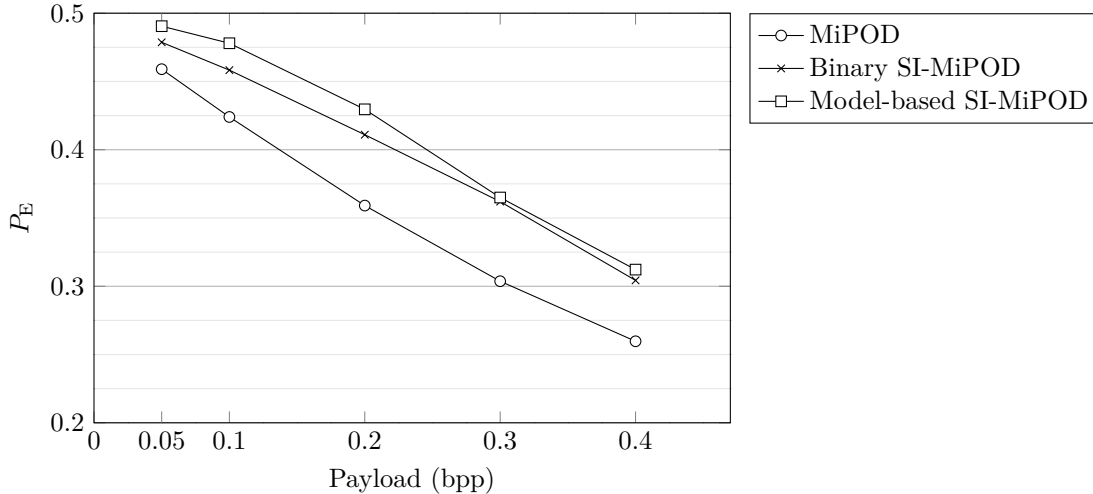


Figure 6.4.1: Security of MiPOD and its heuristic and model-based SI versions with side-information in the form of precover obtained by converting BOSSColor images to grayscale.

was measured using the minimal total probability of error, $P_E = \min_{P_{FA}}(P_{MD} + P_{FA})/2$, where P_{MD} and P_{FA} are missed-detection and false-alarm rates.

Alongside the above proposed side-informed technique with $\hat{t}_{ij} = p_{ij}$ and $\hat{\sigma}_{ij}^2$, we also tested the (non side-informed) MiPOD itself on grayscale cover images with pixel values $x_{ij} = [p_{ij}]$ and variances $\hat{\sigma}_{ij}^2$, and its binary side-informed version with costs modulated as described in (5.0.4). We note that all three embedding schemes were simulated using an embedding simulator.

The model-based binary SI-MiPOD improves the (ternary) MiPOD by 3–5% and the heuristic SI-MiPOD by up to 2%. No improvement over the heuristic method was observed for payloads larger than 0.3 bpp. We expect that further gain may be obtained by searching for the best parameters of the variance estimator in our model.

6.4.3 JPEG domain

We start with a note that with the introduction of model-based side-informed JPEG steganography (SI-J-MiPOD) as described at the end of the previous section, there now exists an equivalent embedding algorithm that does not employ side-information, which we call J-MiPOD. It starts from a JPEG cover image that is decompressed to the spatial domain (without rounding or clipping), pixel variances are estimated in the spatial domain using MiPOD’s variance estimator, and the corresponding variances of DCT coefficients are computed using (6.2.27). MiPOD is then directly applied as described in [82] to quantized DCT coefficients of the cover JPEG. That is, the coefficients are changed by ± 1 with equal probabilities that are computed to minimize the KL divergence between cover and stego DCT coefficients under a payload constraint.

In this section, we thus compare the empirical security of five embedding algorithms: two that do not use side-information and start with a JPEG cover (J-UNIWARD and J-MiPOD) and three side-informed schemes, SI-UNIWARD, heuristic SI-J-MiPOD and model-based SI-J-MiPOD as described at the end of the previous section. The detector was built with the Gabor Filter Residuals (GFR) [85] features and the low-complexity linear classifier.

Figure 6.4.2 shows the detection error P_E as a function of payload in bits per non-zero AC DCT coefficient (bpnzac) for four JPEG quality factors while Figure 6.4.3 shows the same detection error but as a function of JPEG quality factor for two payloads. Neither figure contains error bars for

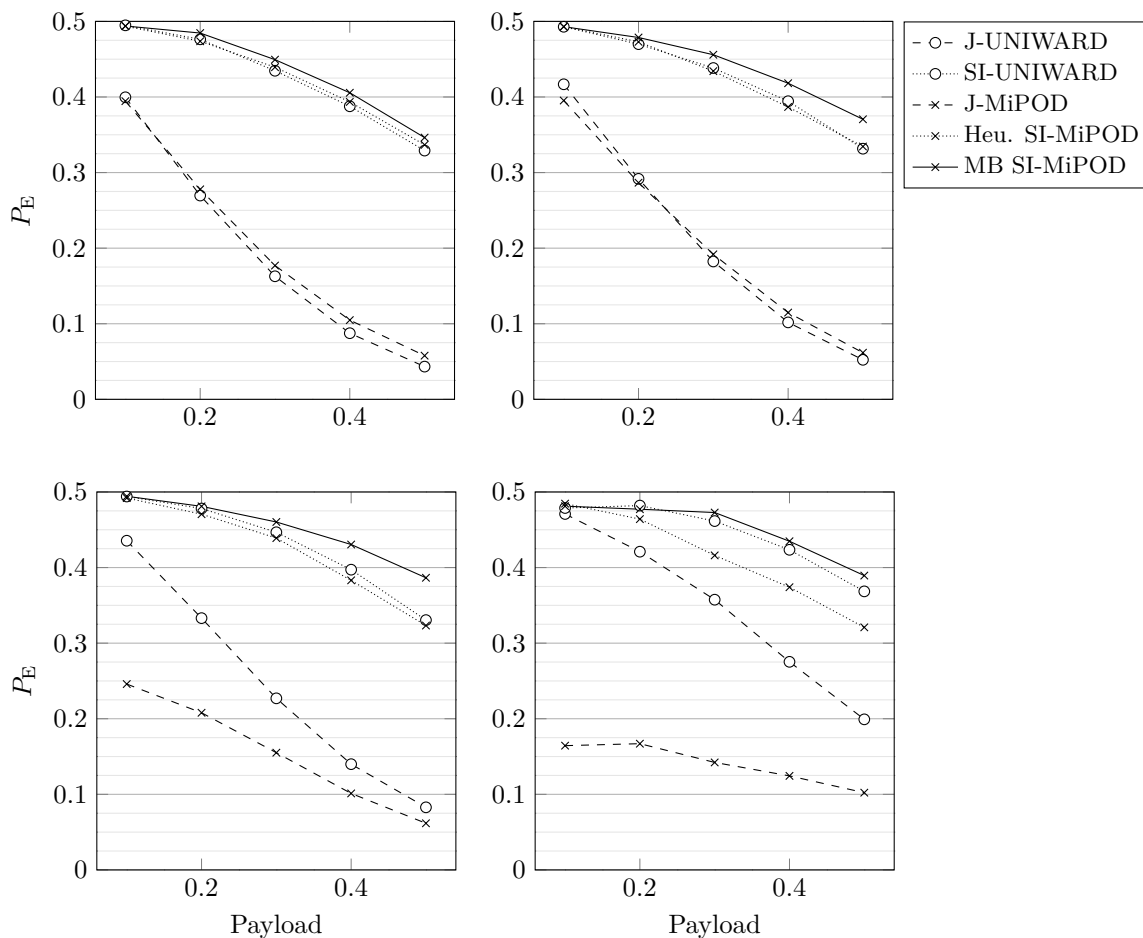


Figure 6.4.2: Security of two non side-informed and three side-informed embedding schemes as a function of payload in bpnzac on BOSSbase for JPEG quality factors 65, 75, 85, and 95.

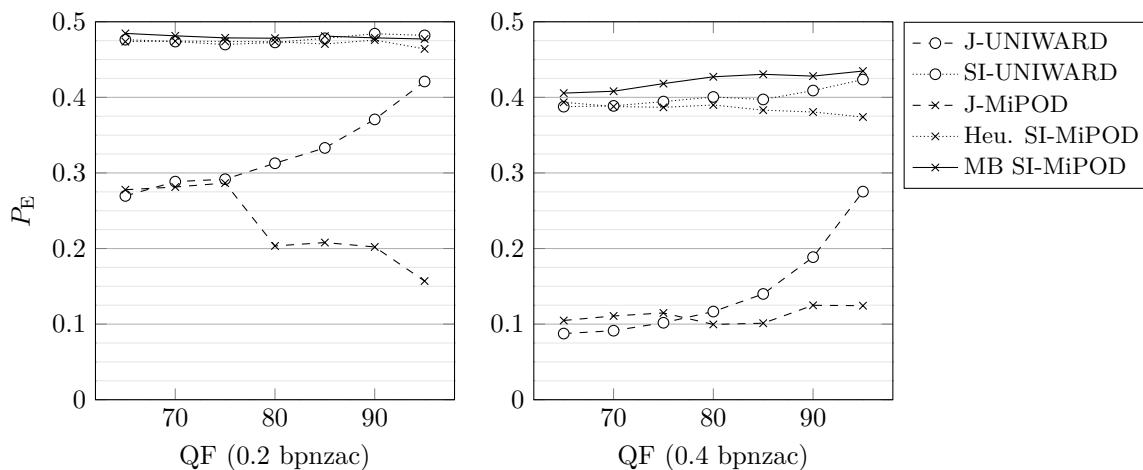


Figure 6.4.3: Security of two non side-informed and three side-informed embedding schemes on BOSSbase as a function of JPEG quality factor for relative payload 0.2 bpnzac (left) and 0.4 (right).

better readability. The average statistical spread of the results in terms of the standard deviation across ten database splits was 0.0029 with 91% of all spreads falling into the range 0.0010–0.0040.

It is comforting to confirm that the model-based SI-J-MiPOD is more secure than the heuristic SI-J-MiPOD, which supports the proposed theoretical approach to side-informed embedding. The gain increases with increased JPEG quality factor and becomes up to almost 7% for QF 95 and payload 0.5 bpnzac. For small payloads, all three side-informed schemes are nearly undetectable across all quality factors as evidenced in the left graph of Figure 6.4.3. For larger payloads, the Model-based SI-J-MiPOD clearly becomes the most secure tested scheme, outperforming SI-UNIWARD as well. It is also interesting to point out that the non side-informed J-MiPOD is on par with J-UNIWARD for lower quality factors (up to 75) but then J-MiPOD starts losing w.r.t. J-UNIWARD. We attribute this to the effect of compression on estimation of pixel variance because when both algorithms are fed costs and variances estimated from the precover, J-MiPOD is more secure than J-UNIWARD across all tested payloads and quality factors. We thus hypothesize that J-MiPOD might benefit from fine-tuning the variance estimator to decompressed JPEG images. Since this topic does not concern side-informed steganography, it is left to our future effort.

6.5 Public vs. private side-information and adaptivity

The selection channel in side-informed steganography is determined by both content complexity via the pixel variance $\hat{\sigma}_{ij}^2$ and by the side-information, the rounding error e_{ij} . The pixel variance is only slightly changed by the embedding itself and thus constitutes a *public* side-information available to the Warden. On the other hand, the rounding error e_{ij} is extremely difficult to estimate from the cover/stego image even for images with simple content, such as blue sky images. Although we cannot cite a source for this claim, this finding is based on our previous unpublished effort and should be entirely plausible considering the fact that, for example, it is generally impossible to obtain an accurate estimate of unquantized values of DCT coefficients in a JPEG file. The only publications related to improving the quality of JPEG decompressed images relate to visually suppressing blockiness artifacts, which is a task that is already difficult and much less demanding than estimating the unquantized DCT coefficients. In short, it is a reasonable assumption that the rounding error is a *private* side-information unavailable to the Warden.

Thus, ideally an embedding scheme should be less adaptive to content ($\hat{\sigma}_{ij}^2$) and more strongly adaptive to the rounding error e_{ij} . Since we designed the model-based SI-MiPOD to minimize the KL divergence between cover and stego distributions, we are essentially assuming an omniscient Warden who knows both $\hat{\sigma}_{ij}^2$ and e_{ij} . Based on the analysis in the above section on comparison between heuristic and model-based SI-MiPOD, the latter is more strongly adaptive to content (the acquisition noise variance) than the former. While the experiments in the previous section show that the model-based approach is less detectable when steganalyzed with the selection-channel-*unaware* SRM (GFR) features (an ignorant Warden), the situation reverses when the steganalyst utilizes the public selection channel (content). However, the model-based approach is indeed by design much less detectable w.r.t. the hypothetical omniscient Warden as long as the adopted model is good enough. Table 6.1 summarizes the results for side-informed embedding in the spatial domain (when converting an RGB image to grayscale) and in JPEG domain when steganalyzing with SRM (GFR) features, which corresponds to an ignorant Warden, maxSRMd2 and selection-channel-aware GFR features (SCA-GFR) [15], which simulates Warden aware of the public side-information, the content, and the omniscient Warden fully informed by both content ($\hat{\sigma}_{ij}^2$) and the private side-information, the rounding error e_{ij} .

In the spatial domain, strangely enough maxSRMd2 detects worse than SRM. On the other hand, in the JPEG domain the GFR seems to detect the embedding as reliably as the SCA variant of the features.

We note that there are really no other options for any realistic Warden who does not know the rounding errors e_{ij} . Fixing e_{ij} at some medium value is virtually the same as fixing it at any other

Experiment	Payload	Scheme	Ignorant	Aware of σ	Aware of σ, e
RGB	0.2 bpp	Heu SI-MiPOD	0.4104±0.0031	0.4402±0.0033	0.2306±0.0020
		MB SI-MiPOD	0.4203±0.0024	0.3951±0.0072	0.3466±0.0029
	0.4 bpp	Heu SI-MiPOD	0.3002±0.0022	0.3317±0.0025	0.2065±0.0026
		MB SI-MiPOD	0.3012±0.0029	0.2696±0.0031	0.2572±0.0031
JPEG	0.2 bpnzac	Heu SI-J-MiPOD	0.4740±0.0034	0.4761±0.0029	0.4529±0.0031
		MB SI-J-MiPOD	0.4787±0.0022	0.4751±0.0023	0.4630±0.0020
	0.4 bpnzac	Heu SI-J-MiPOD	0.3868±0.0040	0.3953±0.0031	0.3618±0.0015
		MB SI-J-MiPOD	0.4182±0.0022	0.4210±0.0018	0.4038±0.0026

Table 6.1: Detection error when steganalyzing heuristic SI-MiPOD and model-based (MB) SI-MiPOD with SRM and maxSRMd2 and their JPEG counterparts with SRM/GFR (ignorant Warden), selection-channel-aware maxSRMd2/GFR (Warden aware of content, $\hat{\sigma}_{ij}^2$), and omniscient Warden aware of both of content $\hat{\sigma}_{ij}^2$ and rounding error e_{ij} .

value. This is because the rounding error is only in a multiplicative factor that modulates the costs (Eq. (5.0.4) or (5.0.5) and (5.0.6)) in the heuristic schemes as well as the Fisher information (6.3.7) in the model-based scheme.

6.6 Conclusions

Steganography with precover at the sender has come a long way. The main progress has been due to advanced coding techniques coupled with a heuristic incorporation of the precover in the embedding algorithm. The typical heuristic calls for modulating costs by $1 - 2|e|$, where $-1/2 < e \leq 1/2$ is the rounding error. Despite the success of side-informed schemes, such as SI-UNIWARD or UED, little has been done to design the embedding algorithm from general principles. This chapter attempts to rectify this situation.

We start by adopting a multivariate Gaussian model for the precover, modeling thus the process of acquiring a digital image using an imaging sensor. By constraining the embedding rule to be binary, the embedding change rates are derived to minimize the total KL divergence between cover and stego models estimated from the available precover while enforcing the payload constraint. In contrast to heuristic schemes, in our model-based approach the rounding error e modulates the Fisher information by multiplying it by $(1 - 2|e|)^2$. The resulting embedding is shown to be more adaptive to content than heuristic side-informed embedding schemes. On experiments with images represented in the spatial and JPEG domain, we demonstrate that the newly derived model-based side-informed steganography enjoys a higher level of empirical security than heuristic embedding schemes when detecting with selection-channel-*unaware* features (ignorant Warden). The same holds when steganalyzing with selection-channel-*aware* features fully informed by the rounding error (omniscient Warden). This is because the model-based schemes were designed to minimize the KL divergence, which implicitly assumes an omniscient Warden. With features that incorporate the knowledge of the content adaptivity, however, the model-based approach is more detectable than heuristic schemes at least in the spatial domain. Optimal embedding should thus be designed within a game-theoretic framework in which both the sender and the Warden randomize their strategies.

The framework introduced in this chapter lends itself to more general forms of side-information, such as multiple acquisitions of the same cover.

Chapter 7

Multiple exposures

Most consumer electronic devices, such as cell phones, tablets, and low-end digital cameras save their images only in the JPEG format and thus do not give the user access to non-rounded DCT coefficients. In this case, Alice can utilize a different type of side-information – she can take multiple JPEG images of the same scene. This research direction has not been developed as much mostly due to the difficulty of acquiring the required imagery and modeling the differences between acquisitions. Prior work on this topic includes [31, 33, 32] where the authors made multiple scans of the same printed image on a flat-bed scanner and then attempted to model the acquisition noise. Unfortunately, this requires acquiring a potentially large number of scans, which makes this approach rather labor intensive. Moreover, differences in the movement of the scanner head between individual scans lead to slight spatial misalignment that complicates using this type of side-information properly. Because this problem is especially pronounced when embedding in the pixel domain, in this chapter we work with multiple images acquired in the JPEG format as we expect quantized DCT coefficients to be naturally more robust to small differences between acquisitions. Since our intention is to design a practical method, we avoid the difficult and potentially extremely time consuming task of modeling the differences between acquisitions [31, 33, 32] and make the approach work well even when mere *two* images are available to Alice. In another relevant prior art [73], the authors proposed embedding by stitching patches from multiple acquisitions in a predefined pattern. The individual patches are not modified and are therefore statistically indistinguishable from the original images. However, as the authors discussed in their paper there are likely going to be detectable differences between individual patches and inconsistencies at their boundaries. Furthermore, the required number of acquisitions quickly grows with the length of the secret message. By using 150 acquisitions of the same scene (scans), the authors were able to embed only 0.157 bits per non-zero AC coefficient on average.

7.1 Preliminaries

For simplicity and WLOG, we will work with 8-bit $n_1 \times n_2$ grayscale images with pixels $\mathbf{P} = (p_{ij}) \in \mathcal{R}^{n_1 \times n_2}$, $\mathcal{R} = \{0, \dots, 255\}$, with both n_1 and n_2 multiples of 8. During JPEG compression, \mathbf{P} is divided into disjoint blocks of 8×8 pixels, $p_{ij}^{(a,b)}$, $1 \leq i, j \leq 8$, $1 \leq a \leq n_1/8$, $1 \leq b \leq n_2/8$, where (a, b) is the block index. Discrete cosine transform (DCT) is then applied to each block, resulting in 8×8 blocks of DCT coefficients $u_{ij}^{(a,b)}$, $\mathbf{U}^{(a,b)} = \text{DCT}(\mathbf{P}^{(a,b)})$, where $\mathbf{U}^{(a,b)}$ and $\mathbf{P}^{(a,b)}$ are 8×8 matrices of DCT coefficients and pixels in the (a, b) th block, respectively. The next step in JPEG compression involves dividing $u_{ij}^{(a,b)}$ by quantization steps q_{ij} , and rounding to integers $x_{ij}^{(a,b)} = Q_1(u_{ij}^{(a,b)}/q_{ij})$, where $Q_1(\cdot)$ quantizes to $\{-1023, \dots, 1024\}$ and $\mathbf{Q} = (q_{ij})$ is the luminance quantization matrix. The quantized DCT coefficients $x_{ij}^{(a,b)}$ are then losslessly encoded, appended with a header, and saved as a JPEG file.

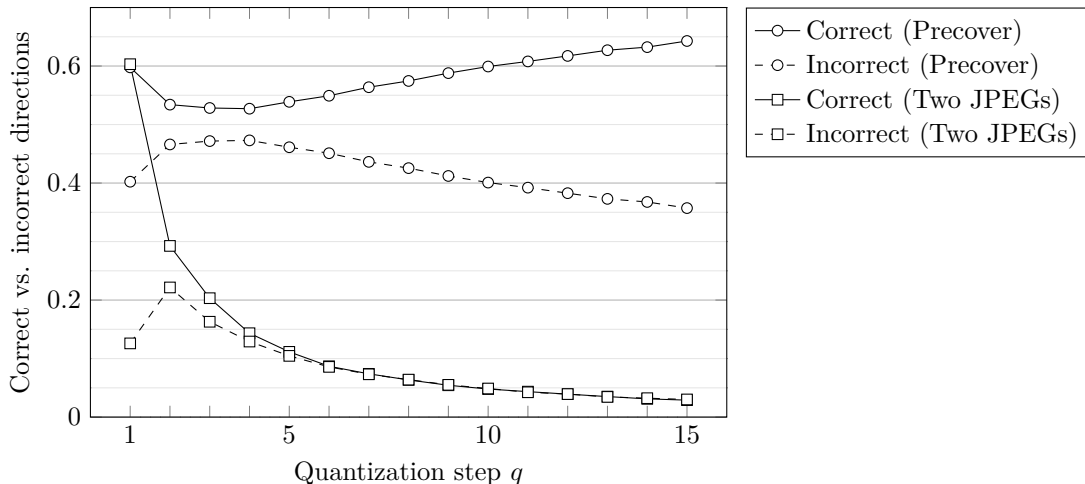


Figure 7.2.1: Relative number of correctly and incorrectly determined embedding directions for steganography informed by the values of non-rounded DCT coefficients (precover) and by two JPEG images. See Section 7.4 for details.

Throughout this chapter, we will use indices i, j to index DCT coefficients in an image as well as in a specific (a, b) th block. Thus, in x_{ij} , the range of indices i, j is over the entire $n_1 \times n_2$ image while in $x_{ij}^{(a,b)}$ it is restricted to $1 \leq i, j \leq 8$. We believe that this switching from global to block-based indexing is natural, it simplifies the language, and should not become a source of confusion.

A generalized Gaussian distribution with density

$$f_{GG}(x; \mu, \alpha, b) = \frac{\alpha}{2b\Gamma(1/\alpha)} \exp\left(-\left|\frac{x - \mu}{b}\right|^\alpha\right), \quad (7.1.1)$$

where μ, α, b are the mean, shape, and width parameters, will be denoted $\mathcal{G}(\mu, \alpha, b)$.

Images acquired using an imaging sensor are noisy measurements of the true scene \mathbf{T} by which we understand the image rendered by the camera lens. The randomness in the form of noise or imperfections is introduced by several separate mechanisms [53], which include the shot noise (photon noise), dark current, and electronic and readout noise. Note that defective pixels and the photo-response non-uniformity are deterministic imperfections that are fixed for a given camera. Formally, $\mathbf{P} = \mathbf{T} + \mathbf{N} \in \mathcal{R}^{n_1 \times n_2}$, where \mathbf{N} is the acquisition noise and \mathbf{T} is a parameter that is unknown to both Alice and the Warden but technically not random. An additive white Gaussian (AWG) model $n_{ij} \sim \mathcal{N}(0, \sigma_a^2)$ is rather accurate for RAW sensor capture of a uniformly lit scene but only an approximation for images with natural content where the variance is a linear function of pixel intensity (the heteroscedastic noise [30, 89]). For a sensor capable of registering color, color interpolation and correction introduce dependencies among neighboring values of n_{ij} and across color channels. Additional local dependencies are introduced by filtering that may be applied inside the camera, such as denoising and sharpening, and by lens distortion correction, making the statistical properties of the random field n_{ij} extremely complicated.

7.2 Steganography with precover

With the exception of YASS [79], all modern embedding schemes for JPEG images, whether or not they use precover, are implemented within the paradigm of distortion minimization. The steganographer first specifies the cost of modifying each cover element (DCT coefficient) and then embeds the payload so that the expected value of the total induced distortion (the sum of costs of all changed

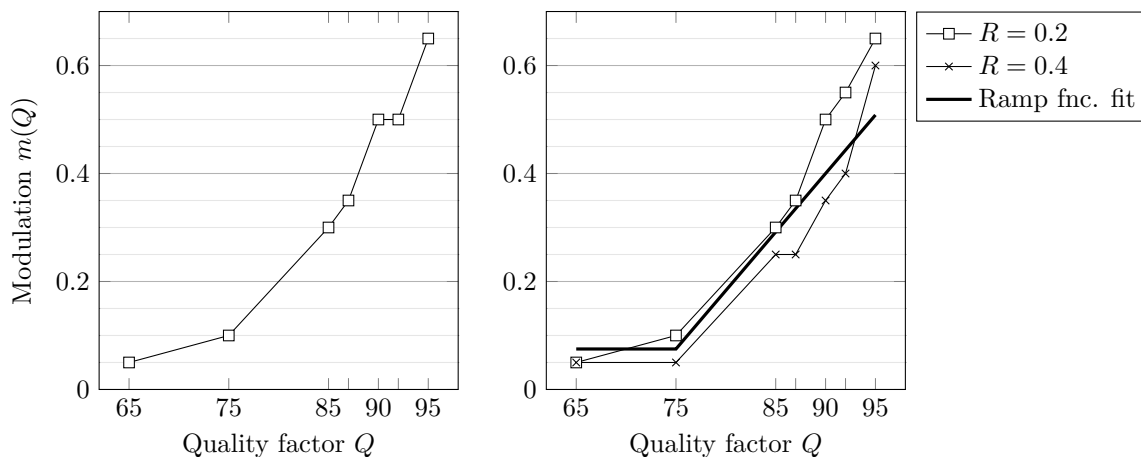


Figure 7.2.2: Optimal modulation factor $m(Q)$ as a function of the JPEG quality factor Q . Left: BOSSbase 1.01 images with simulated acquisition noise. Right: BURSTbase.

cover elements) is as small as possible. Syndrome-trellis codes [28] can achieve this goal near the corresponding rate–distortion bound.

The costs of changing the quantized JPEG coefficient $x_{ij}^{(a,b)}$ by $+1$ and -1 will be denoted $\rho_{ij}^{(a,b)}(+1)$ and $\rho_{ij}^{(a,b)}(-1)$, respectively. The total cost (distortion) of embedding is $D(\mathbf{X}, \mathbf{Y}) = \sum_{x_{ij} \neq y_{ij}} \rho_{ij}(y_{ij} - x_{ij})$, where $y_{ij} \in \{x_{ij} - 1, x_{ij}, x_{ij} + 1\}$ are quantized DCT coefficients from the stego image. An embedding scheme operating at the rate–distortion bound (with minimal D) embeds a payload of R bits by modifying the DCT coefficients with probabilities [28]:

$$\beta_{ij}^{\pm} = \mathbb{P}\{y_{ij} = x_{ij} \pm 1\} = \frac{e^{-\lambda \rho_{ij}(\pm 1)}}{1 + e^{-\lambda \rho_{ij}(+1)} + e^{-\lambda \rho_{ij}(-1)}} \quad (7.2.1)$$

where λ is determined from the payload constraint

$$R = \sum_{ij} h_3(\beta_{ij}^+, \beta_{ij}^-), \quad (7.2.2)$$

with $h_3(x, y) = -x \log_2 x - y \log_2 y - (1 - x - y) \log_2(1 - x - y)$ the ternary entropy function in bits.

One of the most secure schemes for JPEG images called J-UNIWARD [51] uses symmetric costs $\rho_{ij}(+1) = \rho_{ij}(-1)$ for all i, j . Alice can prohibit the embedding from modifying x_{ij} , e.g., by $+1$, by setting $\rho_{ij}(+1) = C_{\text{wet}}$, where C_{wet} is a very large number, the so-called “wet cost” [38].

Side-informed steganography relates to embedding schemes where the sender has some additional information that is used to adjust the costs. For JPEG steganography, the side-information may be in the form of an uncompressed image or, equivalently, the unquantized precover values u_{ij} . Since u_{ij} are not available to the Warden, Alice has a fundamental advantage. As shown in [35], u_{ij} partially compensates for the lack of knowledge of the cover model when it is highly non-stationary.

While it is currently not known how to use side-information in an optimal fashion for embedding, numerous heuristic schemes were proposed in the past [61, 91, 52, 43, 18, 77, 51]. Typically, the rounding error $e_{ij} = u_{ij}/q_{ij} - x_{ij}$, $-1/2 \leq e_{ij} \leq 1/2$, is used to modulate the embedding costs ρ_{ij} by $1 - 2|e_{ij}| \in [0, 1]$. In SI-UNIWARD [51], for example, the costs are:

$$\rho_{ij}(\text{sign}(e_{ij})) = (1 - 2|e_{ij}|)\rho_{ij}^{(J)} \quad (7.2.3)$$

$$\rho_{ij}(-\text{sign}(e_{ij})) = C_{\text{wet}}, \quad (7.2.4)$$

where $\rho_{ij}^{(J)}$ are J-UNIWARD costs. In other words, SI-UNIWARD is a binary embedding scheme that either leaves a DCT coefficient unmodified (rounds u_{ij}/q_{ij} to x_{ij}) or rounds it to the “other side” in

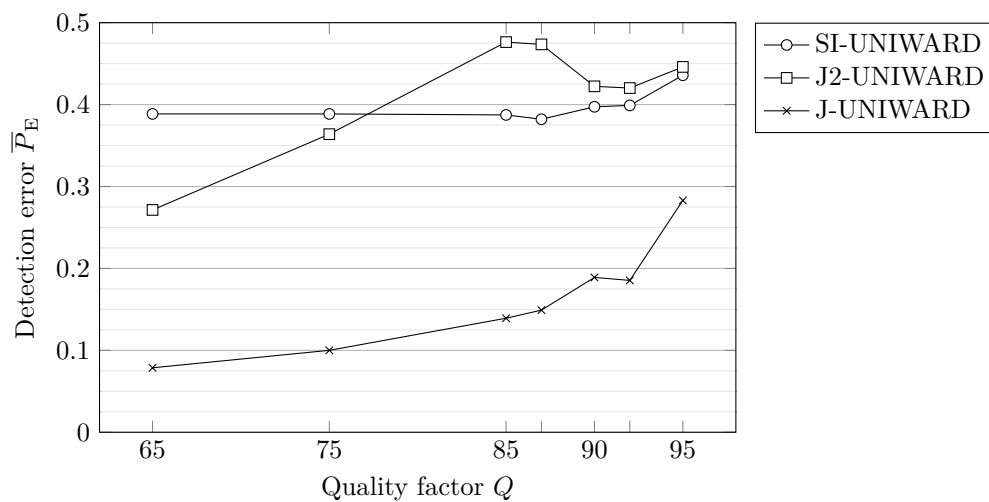


Figure 7.2.3: Empirical security, \bar{P}_E , as a function of the JPEG quality factor for relative payload $R = 0.4$ bpnzac for J2-UNIWARD, J-UNIWARD, and SI-UNIWARD. BOSSbase with simulated acquisition noise, low-complexity linear classifier trained with GFR.

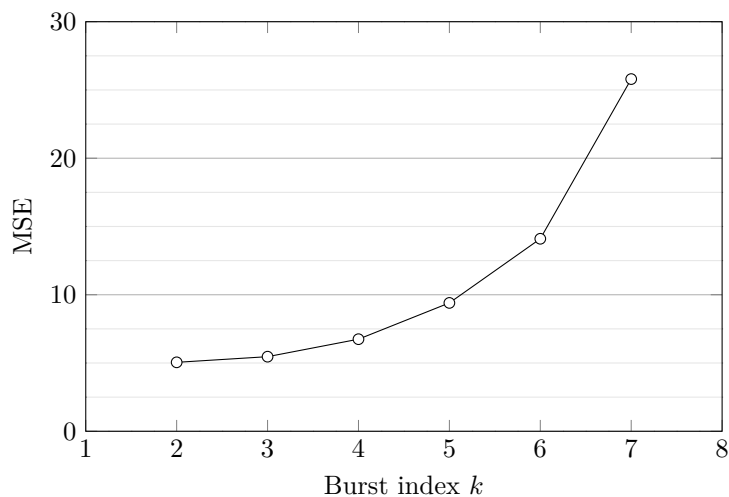


Figure 7.2.4: MSE between $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(k)}$, $k = 2, \dots, 7$ from each burst averaged over all 9,310 bursts from BURSTbase. See Section 7.5 for notation and further details.

the direction of x_{ij} , in which case the J-UNIWARD cost associated with this change is modulated. The intuition behind the modulation is clear: when $|e_{ij}| \approx 1/2$, a small perturbation could cause u_{ij}/q_{ij} to be rounded to the other side. Such coefficients are thus assigned a proportionally smaller cost. On the other hand, the costs are unchanged when $e_{ij} \approx 0$, as it takes a larger perturbation to change the rounded value.

In [18], a ternary version of SI-UNIWARD was studied where the authors argued that, as the rounding error e_{ij} becomes small, the embedding rule should be allowed to change the coefficient both ways. This ternary version of SI-UNIWARD uses the following costs:

$$\rho_{ij}(\text{sign}(e_{ij})) = (1 - 2|e_{ij}|)\rho_{ij}^{(J)} \quad (7.2.5)$$

$$\rho_{ij}(-\text{sign}(e_{ij})) = \rho_{ij}^{(J)}. \quad (7.2.6)$$

7.3 Steganography with multiple JPEGs

In this section, we describe the proposed scheme for embedding in JPEG images when the sender possesses more than one acquisition of (approximately) the same scene. We start with the embedding algorithm for two acquisitions and then discuss the possibilities for its generalization to more than two acquisitions. The main embedding algorithm is explained with a pseudo-code to allow faster understanding of the main concept and ease the implementation for practitioners.

Before we start, we wish to discuss some important philosophical issues. In reality, it is in principle impossible to obtain two independent samplings of one object (Heraclitus’ “You could not step twice into the same river” by Plato in *Cratylus*, 402a) because of small differences in exposure time, physical shaking of the camera, and small differences in the scene itself, e.g., due to wind and the amount and direction of illumination. In this chapter, for brevity we nevertheless abuse the language a little while being aware of the fact that in reality the images will inevitably contain differences other than those due to acquisition noise. One mission of this chapter is to investigate whether, despite these obvious limitations, it is possible to make use of the other acquisitions to improve steganographic security.

The proposed method can be applied to any cost-based scheme that embeds in quantized DCT coefficients of a JPEG file. In fact, it is not limited to the JPEG format and could be applied to other lossy formats, such as the JPEG 2000. We restrict ourselves to JPEG images in this chapter because it is by far the most ubiquitous image format in current use.

7.3.1 Two exposures

First, we describe the embedding algorithm when two JPEG versions of the cover image are available. We denote the quantized DCT coefficients in both images by $x_{ij}^{(1)}$ and $x_{ij}^{(2)}$ and pronounce, for example, the first image as the cover JPEG and consider $x_{ij}^{(2)}$ as side-information.

Pronouncing $x_{ij}^{(1)}$ as cover and $x_{ij}^{(2)}$ as side-information, the sender first computes from $x_{ij}^{(1)}$ the costs of changing the ij th DCT coefficient by -1 and $+1$: $\rho_{ij}(-1)$ and $\rho_{ij}(+1)$. The costs can be computed using, e.g., an existing cost-based embedding scheme, such as J-UNIWARD or one of the versions of UED. The proposed embedding scheme keeps these costs when $x_{ij}^{(1)} = x_{ij}^{(2)}$ and modulates the costs otherwise. This can be explained by finding the new costs $\rho_{ij}^{(\text{SI})}(\pm 1)$ via the following two-step procedure:

$$\text{Step 1 : set } \rho_{ij}^{(\text{SI})}(\pm 1) = \rho_{ij}(\pm 1) \quad (7.3.1)$$

$$\text{Step 2 : } x_{ij}^{(1)} \neq x_{ij}^{(2)} \Rightarrow \rho_{ij}^{(\text{SI})}(s_{ij}) = m(Q)\rho_{ij}(s_{ij}), \quad (7.3.2)$$

$$\text{where } s_{ij} = \text{sign}(x_{ij}^{(2)} - x_{ij}^{(1)}) \quad (7.3.3)$$

where $m(Q) \in [0, 1]$ is a modulation factor that depends on the quality factor $1 \leq Q \leq 100$. To ease the understanding of the embedding method and its implementation, Algorithm 7.1 shows the pseudo-code for the embedding algorithm.

The value of the modulation factor $m(Q)$ will be determined experimentally for each tested quality factor Q and cover source by a search over $m(Q) \in [0, 1]$ to obtain the smallest minimal total probability of error, $P_E = \min_{P_{FA}} (P_{MD} + P_{FA})/2$, where P_{MD} and P_{FA} are missed-detection and false-alarm rates of a detector implemented using a low-complexity linear classifier [14] with the Gabor Filter Residual (GFR) features [85] on the training set. The GFR features were selected for the design because they are known to be highly effective against modern JPEG steganography, including J-UNIWARD and all versions of UED [42, 43]. Experiments show that $m(Q)$ should generally be increasing in Q . The experimental Sections 7.4 and 7.6 and the appendix contain further details on the specific form $m(Q)$.

Our final note of this section concerns a naming convention. An embedding scheme with two JPEGs with J-UNIWARD (UED-JC) costs will be abbreviated as J2-UNIWARD (and UED2-JC).

7.3.2 Multiple exposures

In this subsection, we discuss several possibilities for extending the embedding algorithm to the case when Alice acquires $k > 2$ JPEG images of the same scene, $x_{ij}^{(1)}, \dots, x_{ij}^{(k)}$.

With increased k , it may become possible to obtain a more accurate estimate of the noise-free scene T_{ij} (Section 7.1), for example, as a maximum-likelihood $\hat{t}_{ij}^{(ML)} = (x_{ij}^{(1)} + \dots + x_{ij}^{(k)})/k$ or a MAP estimate by leveraging a prior on $x_{ij}^{(u,v)}$, $1 \leq i, j \leq 8$, with u, v the 8×8 block index, estimated for the given source. The estimates, however, will likely be biased since spatial misalignment between exposures and differences other than due to the acquisition noise will likely increase with k , making it not clear whether the additional exposures are an asset.

Moreover, it is not clear how the embedding should incorporate such estimates. Using \hat{t}_{ij} as a “high-quality precover” and applying standard side-informed steganography, such as SI-UNIWARD, is questionable because the rounded values $[\hat{t}_{ij}]$ form a different source with a suppressed acquisition noise. On the other hand, using \hat{t}_{ij} as a “high-quality precover” for one of the JPEGs, e.g., $x_{ij}^{(1)}$, would lead to “rounding errors” $e_{ij} = \hat{t}_{ij} - x_{ij}^{(1)}$ out of the range $[-1/2, 1/2]$ and would thus require a revisit of the established cost modulation (7.2.3) and (7.2.5).

In the end, and based on our experiments in Subsections 7.6.1 and 7.6.2, it appears that the best way to use multiple images in practice is to simply select a pair of two closest images among the k exposures and apply the algorithm described in the previous subsection.

7.4 Study with simulated acquisition noise

Our first experimental evaluation involves tests on images with simulated acquisition noise. These are included because they constitute the “ideal” (and unachievable in practice) situation when no other differences between the exposures exist besides a very simple form of the acquisition noise. These results will be contrasted with real multiple exposures.

The mother database was BOSSbase 1.01 [7] containing 10,000 8-bit grayscale 512×512 PGM images. Two different realizations of Gaussian noise $\mathcal{N}(0, 1)$ were added to the images, producing two simulated acquisitions $p_{ij}^{(l)}$, $l = 1, 2$, which were subsequently compressed with a range of JPEG quality factors to obtain the values of rounded DCT coefficients $x_{ij}^{(l)}$, $l = 1, 2$, for each image in the database. Each JPEG image $x_{ij}^{(1)}$ was then embedded with relative payload $R = 0.4$ bits per

```

1: Input: Two quality factor  $Q$  JPEG images with quantized DCT coefficients  $x_{ij}^{(1)}$  and
    $x_{ij}^{(2)}$ ,  $1 \leq i \leq M$ ,  $1 \leq j \leq N$ 
2: Output: Stego JPEG image with DCT coefficients  $y_{ij}^{(1)}$ 
3: Compute costs  $\rho_{ij}^{(0)}(-1), \rho_{ij}^{(0)}(+1)$  of DCT coefficients from JPEG  $x_{ij}^{(1)}$  (the cover)
4: for  $i = 1, \dots, M$  do
5:   for  $j = 1, \dots, N$  do
6:      $\rho_{ij}(\pm 1) = \rho_{ij}^{(0)}(\pm 1)$ 
7:      $s_{ij} = \text{sign}(x_{ij}^{(2)} - x_{ij}^{(1)})$ 
8:     IF  $x_{ij}^{(1)} \neq x_{ij}^{(2)}$  THEN  $\rho_{ij}(s_{ij}) = m(Q)\rho_{ij}^{(0)}(s_{ij})$ 
9:   end for
10: end for
11: Embed message in  $x_{ij}^{(1)}$  using costs  $\rho_{ij}$  using STCs to obtain stego JPEG file with
    DCT coefficients  $y_{ij}$ 
12: Recipient reads the secret message using STCs from the stego JPEG file  $y_{ij}$ 

```

Algorithm 7.1: Pseudo-code for side-informed embedding with two JPEGs.

non-zero AC DCT coefficient (bpnzac) using J2-UNIWARD. The values of the optimal modulation factor $m(Q)$ as a function of Q for this source are shown in Figure 7.2.2 left.

Figure 7.2.3 shows \bar{P}_E , which is the detection error P_E averaged over ten random splits of the database into training and testing parts as a function of the JPEG quality factor. We do not show the statistical spread of the detection error as it is very small and in most cases covered by the markers. In all experiments in this manuscript, the largest encountered standard deviation of the detection error was 0.0122 and the average was 0.0042. The classifier was a low-complexity linear classifier [14] and the feature set is the Gabor Filter Residual (GFR) [85] rich model known to be highly effective against modern steganographic schemes. For comparison, the figure also contains the detection error for J-UNIWARD (with $x_{ij}^{(1)}$ as covers) and SI-UNIWARD (with $p_{ij}^{(1)}$ as side-information). For a simulated acquisition noise, the side-information in the form of two JPEG images significantly increases empirical security w.r.t. embedding with a single JPEG (J-UNIWARD). It seems even more valuable for quality factors $Q \gtrsim 80$ than non-rounded DCT coefficients (SI-UNIWARD). We next shed some light on why this is the case.

The value $p_{ij}^{(2)}$ can only be useful to Alice when $p_{ij}^{(2)} \neq p_{ij}^{(1)}$, which will happen increasingly more often with smaller quantization steps q_{ij} (larger JPEG quality). This type of side-information is different from the non-rounded values $u_{ij}^{(1)}$. It informs Alice about the direction along which the costs should be modulated and less about the magnitude of the rounding error $e_{ij}^{(1)} = u_{ij}^{(1)}/q_{ij} - x_{ij}^{(1)}$. To better understand the difference between these two types of side-information, we conducted the following experiment.

A generalized Gaussian model $\mathcal{G}(0, 0.4, 0.1)$ was adopted for the distribution of DCT coefficients of the noise-free scene \mathbf{T} . These parameters roughly correspond to medium spatial frequencies in BOSSbase 1.01 [7] images. Then, we generated $2 \times N_{\text{MC}}$ independent realizations from $\mathcal{G}(0, 0.4, 0.1)$, $t_k^{(1)}$ and $t_k^{(2)}$, $k \in \{1, \dots, N_{\text{MC}}\}$, $N_{\text{MC}} = 10^6$. Next, N_{MC} independent realizations from $\mathcal{N}(0, 1)$ were added to both vectors,¹ divided by $q \in \{1, \dots, 15\}$ and rounded to integers, $p_k^{(l)} = t_k^{(l)} + n_k^{(l)}$, $x_k^{(l)} = \lfloor p_k^{(l)}/q \rfloor$, $l = 1, 2$. We then counted how often the different side-information correctly informed us about the sign of the rounding error (direction of the stego changes).

We will say that side-information $p_k^{(1)}$ correctly determines the direction of steganographic changes with respect to the noise-free scene if the embedding modifies the quantized cover value $x_k^{(1)}$ towards

¹ $\sigma_a = 1$ approximately corresponds to acquisition noise with 1/60th sec. exposure at 100 ISO with Canon 6D.

the noise-free scene $t_k^{(1)}$, which will happen when the rounding error $e_k^{(1)} = p_k^{(1)}/q - x_k^{(1)}$ has the same sign as $t_k^{(1)}/q - x_k^{(1)}$, or when $((p_k^{(1)} + n_k^{(1)})/q - x_k^{(1)})(t_k^{(1)}/q - x_k^{(1)}) > 0$. It determines the direction incorrectly if this product is negative.² Similarly, we will say that side-information $x_k^{(2)}$ determines the correct direction with respect to the noise-free scene if $(x_k^{(2)} - x_k^{(1)})(t_k^{(1)}/q - x_k^{(1)}) > 0$. When this product is negative, it determines the direction incorrectly. When it is zero ($x_k^{(2)} = x_k^{(1)}$), the side-information is not useful.

Figure 7.2.1 shows the relative number of correctly and incorrectly determined embedding directions based on side-information in the form of one non-quantized DCT coefficient $c_k^{(1)}$ (Precover) and two quantized coefficients $x_k^{(1)}$ and $x_k^{(2)}$ (Two JPEGs). The most interesting part of the figure is for small values of q . Two quantized images are much more conservative in the sense that they determine the direction incorrectly much less frequently than from one non-rounded value. On the other hand, with increasing q , the two quantized images find fewer correct directions. For small values of $q = 1, 2, 3$ (more generally, for large values of σ_a/q), two JPEG images provide more useful side-information about the preferred changes compared to the non-rounded DCTs. This is in qualitative agreement with Figure 7.2.3 that shows that J2-UNIWARD indeed outperforms SI-UNIWARD for high quality factors (small q).

Note that for side-information with precover values $p_k^{(1)}$, the sum of the relative number of correctly and incorrectly determined directions is one while this is not the case for two quantized coefficients because “ties” $x_k^{(1)} = x_k^{(2)}$ occur with non-zero probability.

7.5 Datasets for experiments

In general, it is difficult to acquire two images of the same scene because the camera position may slightly change between the exposures even when mounted on a tripod due to vibrations caused by the shutter. Another potential source of differences is slightly varying exposure time and changing light conditions between exposures. To test the real-life performance of the proposed side-informed steganography in Section 7.6, we prepared two new datasets: BURSTbase with images obtained with a camera mounted on a tripod and BURSTbaseH with images shot from hand.

7.5.1 BURSTbase

To eliminate possible impact of flicker of artificial lights, all images were acquired in daylight, both indoor and outdoor, and without a flash. Canon 6D, a DSLR camera with a full-frame 20 MP CMOS sensor, set to ISO 200 was used in a burst mode. The shutter was operated with a two-second self-timer to further minimize vibrations due to operating the camera. To prevent the camera from changing the settings during the burst, it was used in manual mode. All images were acquired in the RAW CR2 format and then exported from Lightroom 5.7 to 24-bit TIFF format with no other processing applied.

We acquired 133 bursts, each containing 7 images. To increase the number of images for experiments, the 5472×3648 TIFF images were cropped into 10×7 equidistantly positioned tiles with 512×512 pixels. This required a slight overlap between neighboring tiles (7 pixels horizontally and 35 pixels vertically). These $70 \times 133 = 9,310$ smaller images were then converted to grayscale in Matlab using `'rgb2gray'` and saved in a lossless raster format to facilitate experiments with a range of JPEG quality factors. We call this database of $7 \times 9,310$ uncompressed grayscale images 'BURSTbase'.

For each pair of different images from each burst, we computed the mean square error (MSE) between them and then selected the pair with the smallest MSE, denoting one of them randomly as $x_{ij}^{(1)}$ and

²We can ignore the zero-probability event $t_k^{(1)}/q = x_k^{(1)}$.

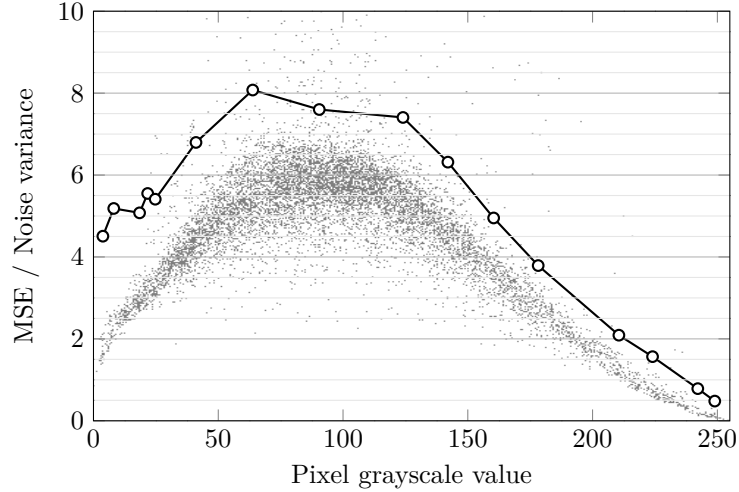


Figure 7.5.1: Gray dots: $\text{MSE}(\mathbf{X}^{(1)}, \mathbf{X}^{(2)})$ vs. average grayscale of $\mathbf{X}^{(1)}$ across images from BURSTbase. Circles: acquisition noise variance estimated from images of gray wall. Both at ISO 200.

the other $x_{ij}^{(2)}$. The remaining five images from the burst were denoted $x_{ij}^{(k)}$, $k = 3, \dots, 7$, so that the MSE between $x_{ij}^{(1)}$ and $x_{ij}^{(k)}$ forms a non-decreasing sequence in k . We analyzed images from BURSTbase sorted in this manner to determine how much the differences between images are due to acquisition noise or slight spatial misalignment. Figure 7.2.4 shows the MSE between $x_{ij}^{(1)}$ and $x_{ij}^{(k)}$, $k = 2, \dots, 7$, averaged over the entire BURSTbase. For the closest pairs, $\text{MSE}(\mathbf{X}^{(1)}, \mathbf{X}^{(2)}) \approx 5$, which would correspond to $\sigma_a^2 = 5$ if the differences were solely due to AWG noise with variance σ_a^2 . This closely matches the variance estimated from a single image of content-less scenes with medium gray. This reasoning indicates that $\mathbf{X}^{(2)}$ and $\mathbf{X}^{(3)}$ are on average reasonably well aligned with $\mathbf{X}^{(1)}$ while $\mathbf{X}^{(k)}$, $k \geq 4$, are increasingly more affected by small spatial shifts.

To obtain additional evidence that the differences between the two closest images from each burst are due to acquisition noise rather than slight spatial misalignment, we conducted another experiment in which we studied the MSE as a function of luminance. This was done to capture the dependence of the acquisition noise variance on luminance – it follows the heteroscedastic model further modified by tonal curve adjustment. To map out the dependence, we took RAW images of a uniform gray wall in the exposure priority mode with a wide range of exposures while all other settings were kept unchanged (at ISO 200). These flat-field images were then exported from Lightroom to 24-bit TIFF images, converted to grayscale using Matlab’s `rgb2gray`, and cropped to the central 512×512 region. To isolate only the acquisition noise, a third-degree polynomial fit for each pixel on a sliding 32×32 block was subtracted from the pixels to remove any leftover gradual fall-off of luminance towards the image edges due to vignetting. Figure 7.5.1 shows the MSE as a function of the average image grayscale across BURSTbase, with the circles corresponding to variance–grayscale pairs from images of gray wall. The data is in qualitative agreement with the maximum variance for pixels with grayscale around 100. The decreased variance for grayscales below 100 is most likely due to the tonal adjustment done by cameras to avoid magnifying noise in underexposed areas.

7.5.2 BURSTbaseH

Since most casual photographers do not shoot from a tripod, we prepared a second dataset with images shot from hand to see whether the proposed modulation of costs still provides a boost under this more realistic and less ideal conditions. A different set of images was acquired using the same Canon 6D camera on a different day, this time with the camera being hand-held instead of mounted on a tripod. A total of 154 bursts of 7–13 images were obtained that were processed and then

γ	1	0.5	0.2	0.1
max MSE	3790	100.1	25.42	12.94
avg MSE	254.23	39.10	13.32	7.81

Table 7.1: Maximum and average MSE between two closest exposures from each burst in BURSTbaseH when constraining it to a fraction γ of best bursts.

cropped into 10,780 smaller 512×512 images in the same manner as described in the previous subsection. To distinguish this source from BURSTbase, we call this database BURSTbaseH (H as in **H**and-held).

The average MSE between the two closest images from each burst was 254.23, which is significantly larger than for BURSTbase (5.05). This tells us that the images are on average misaligned by a large amount, which is likely to have a significant impact on the security of the proposed scheme. The steganographer, however, can reject bad bursts and/or take another one and only embed in images from bursts that are not grossly misaligned. In fact, many mobile devices today are capable of taking bursts, such as for HDR photography or to reduce high-ISO noise. The authors envision a mobile app that would leverage this capability for the purpose of increasing the security of steganographic communication. Another possibility to obtain well-aligned multiple exposures is to extract consecutive frames from short M-JPEG video clips. This, too, could be achieved with a mobile app.

Based on the considerations spelled out in the previous paragraph, in the next section we experiment with subsets of BURSTbaseH consisting of a fraction $\gamma \in [0, 1]$ of images with the smallest MSE for the closest pair. For example, in BURSTbaseH with $\gamma = 0.5$, we selected $10,780/2 = 5,390$ bursts with the smallest MSE, eliminating thus half of the bursts with the worst misalignment. Table 7.1 shows the average MSE between the closest pair of images when constraining BURSTbaseH to the fraction of $\gamma \in \{0.1, 0.2, 0.5, 1\}$ best bursts. Note that the average MSE between the two closest exposures from each burst in BURSTbaseH with $\gamma = 0.1$ is rather close to the MSE between the closest images of BURSTbase.

7.6 Experiments

In this section, we first study the empirical security of J2-UNIWARD on BURSTbase across a range of quality factors and payloads and contrast it with J-UNIWARD and SI-UNIWARD. We also assess how the security boost of the second exposure changes with increased differences between exposures. In the second round of experiments, we assess the performance of the proposed scheme in more realistic conditions when the bursts are taken with a hand-held camera instead of mounted on a tripod (BURSTbaseH). On tests with J2-UNIWARD and UED2-JC, we show that when bad bursts are rejected embedding with two JPEGs still provides a significant performance boost with respect to embedding in single JPEGs despite rather large spatial misalignments.

Since the feedback from a detector utilizing the GFR feature set was used to determine the modulation factor, it is essential that we test J2-UNIWARD with other feature sets to evaluate its security. Thus, all experiments in this section were executed with a low-complexity linear classifier trained with the merger of the GFR features, the spatial rich model (SRM) [39], and the cartesian-calibrated JPEG Rich Model (ccJRM) [63].

7.6.1 BURSTbase

The modulation factor $m(Q)$ (7.3.3) found experimentally as described in Section 7.3 is shown in Figure 7.2.2 right. All our experiments in this subsection were executed with $m(Q)$ approximated

by a following ramp function:

$$m(Q) = \max\{0.075, 0.02167 \times Q - 1.55\}. \quad (7.6.1)$$

The appendix contains a simple qualitative argument explaining why the modulation factor follows a ramp function.

Figure 7.6.1 left shows \bar{P}_E as a function of the JPEG quality factor for payload 0.2 bpnzac together with the results for J-UNIWARD (with $x_k^{(1)}$ as covers) and SI-UNIWARD (with $c_k^{(1)}$ as side-information). For real acquisitions, the side-information in the form of two JPEG images significantly increases empirical security w.r.t. embedding with a single JPEG (J-UNIWARD). In contrast with the experiments with simulated acquisition noise, however, the empirical security is not better than when non-rounded DCT coefficients are used as side-information (SI-UNIWARD). For completeness, in Figure 7.6.1 right we report the detection error as a function of the quality factor for five payloads and in Table 7.2 we report all numerical values, including the results obtained with STCs with constraint height $h = 10$ rather than with an embedding simulator to see the coding loss.

To assess how sensitive J2-UNIWARD is w.r.t. small differences between exposures, we implemented it with $x_{ij}^{(1)}$ as cover and $x_{ij}^{(k)}$, $k = 3, \dots, 7$ as side-information, essentially using the second closest ($k = 3$), the third closest ($k = 4$), etc., image instead of the closest image. As apparent from Figure 7.2.4, with increasing k the MSE increases and thus the security boost should start diminish. Figure 7.6.3 shows \bar{P}_E as a function of the quality factor across $k = 2, \dots, 7$ together with the value of J-UNIWARD. While the gain of the second image indeed decreases with increased MSE, this decrease is rather gradual and very small for higher quality factors. This experiment proves that the second exposure provides useful side-information even when small spatial shifts are present opening thus the possibility to improve steganography even when multiple exposures are acquired with a hand-held camera rather than mounted on a tripod, a topic studied in the next subsection.

7.6.2 BURSTbaseH

To investigate the security of the proposed technique under more realistic setting, we experimented with J2-UNIWARD and UED2-JC on BURSTbaseH with $\gamma \in \{0.1, 0.2, 0.5, 1\}$ for a range of quality factors and payloads. For J2-UNIWARD, we reused the modulation factor $m(Q)$ determined on BURSTbase (Eq. (7.6.1)). Although we did perform a search for the best modulation factor for UED2-JC, the detection error was rather insensitive to $m(Q)$ as long as it was sufficiently small. In all our experiments with UED2-JC, the modulation factor was set as $m(Q) = 0.01$ for all tested payloads and quality factors.

Figure 7.6.4 shows the detection error \bar{P}_E for two payloads for JPEG quality factor 75 for all four values of γ for J-UNIWARD, J2-UNIWARD, and SI-UNIWARD with the same steganalysis detector as in the previous subsection. Figure 7.6.5 contains the detection error for the same three embedding schemes as a function of JPEG quality factor for $\gamma = 0.1$. Both figures demonstrate a substantial gain in security of J2-UNIWARD w.r.t. J-UNIWARD. While this gain is understandably smaller for the images of BURSTbaseH, it becomes substantial in comparison with embedding with a single JPEG image as the number of rejected bursts increases. The numerical values of \bar{P}_E of all experiments are provided in Table 7.3.

In Figure 7.6.6, we display the detection error as a function of γ for two payloads and two quality factors for the UED-JC embedding algorithm. Here, the bad burst rejection is even more effective than for J2-UNIWARD. For quality factor 95, UED2-JC even outperforms UED informed by the precover (SI-UED-JC) for all $\gamma < 1$. Substantial security gain is observed even for $\gamma = 0.5$, e.g., when every other burst is rejected on average, across all payloads and quality factors.

R	\mathcal{M}	Quality factor Q						
		65	75	85	87	90	92	95
0.1	SI	0.4991	0.4973	0.4897	0.4892	0.4952	0.4984	0.4525
	J	0.3508	0.3541	0.3766	0.4892	0.4121	0.4087	0.4421
	J2	0.4897	0.4659	0.4610	0.4633	0.4560	0.4523	0.4433
	J2c	0.4550	0.4591	0.4326	0.4289	0.4149	0.4138	0.4155
0.2	SI	0.4815	0.4811	0.4761	0.4753	0.4812	0.4811	0.4498
	J	0.1946	0.1953	0.2258	0.2301	0.2840	0.2787	0.3622
	J2	0.4620	0.4275	0.4178	0.4128	0.4161	0.4100	0.3796
	J2c	0.4146	0.4186	0.4179	0.4119	0.4103	0.3959	0.3695
0.3	SI	0.4501	0.4456	0.4406	0.4437	0.4506	0.4520	0.4200
	J	0.1010	0.0975	0.1179	0.1256	0.1771	0.1660	0.2647
	J2	0.4245	0.3827	0.3729	0.3723	0.3733	0.3560	0.3196
	J2c	0.3740	0.3709	0.3626	0.3524	0.3569	0.3346	0.2990
0.4	SI	0.4056	0.3989	0.3976	0.3963	0.4118	0.4037	0.4201
	J	0.0528	0.0469	0.0592	0.0627	0.0980	0.0906	0.1776
	J2	0.3734	0.3394	0.3144	0.3084	0.3218	0.2932	0.2647
	J2c	0.3356	0.3244	0.2949	0.2862	0.2976	0.2649	0.2380
0.5	SI	0.3552	0.3446	0.3392	0.3361	0.3571	0.3491	0.3779
	J	0.0280	0.0234	0.0289	0.0291	0.0506	0.0444	0.1076
	J2	0.3062	0.2989	0.2501	0.2383	0.2569	0.2168	0.2043
	J2c	0.2777	0.2815	0.2210	0.1991	0.2231	0.1848	0.1779

Table 7.2: Empirical security \bar{P}_E of embedding schemes, \mathcal{M} , J-UNIWARD (J), J2-UNIWARD (J2), J2-UNIWARD implemented using STCs (J2c), and SI-UNIWARD (SI) on BURSTbase for a range of payloads, R , and quality factors.

R	\mathcal{M}	Quality factor Q						
		65	75	85	87	90	92	95
0.2	SI	0.4788	0.4706	0.4744	0.4697	0.4736	0.4739	0.4541
	J	0.2596	0.2600	0.2729	0.2769	0.2769	0.2996	0.3887
	J2	0.3786	0.3963	0.4084	0.4163	0.4250	0.4176	0.4260
0.4	SI	0.4372	0.4305	0.4186	0.4275	0.4442	0.4541	0.4363
	J	0.1267	0.1131	0.1000	0.1043	0.1356	0.3887	0.2399
	J2	0.2583	0.0075	0.3020	0.2956	0.3274	0.4260	0.3518

Table 7.3: Empirical security \bar{P}_E of embedding schemes, \mathcal{M} , J-UNIWARD (J), J2-UNIWARD (J2) and SI-UNIWARD (SI) on BURSTbaseH for a range of payloads, R , and quality factors for $\gamma = 0.1$.

R	\mathcal{M}	Quality factor Q	
		75	95
0.2	SI	0.2185	0.2893
	U	0.0462	0.1318
	U2	0.1995	0.3547
0.4	SI	0.0970	0.2477
	U	0.0250	0.0706
	U2	0.1032	0.1884

Table 7.4: Empirical security \bar{P}_E of embedding schemes, \mathcal{M} , UED-JC (U), UED2-JC (U2), and SI-UED-JC (SI) on BURSTbaseH for two payloads and two JPEG quality factors for $\gamma = 0.1$.

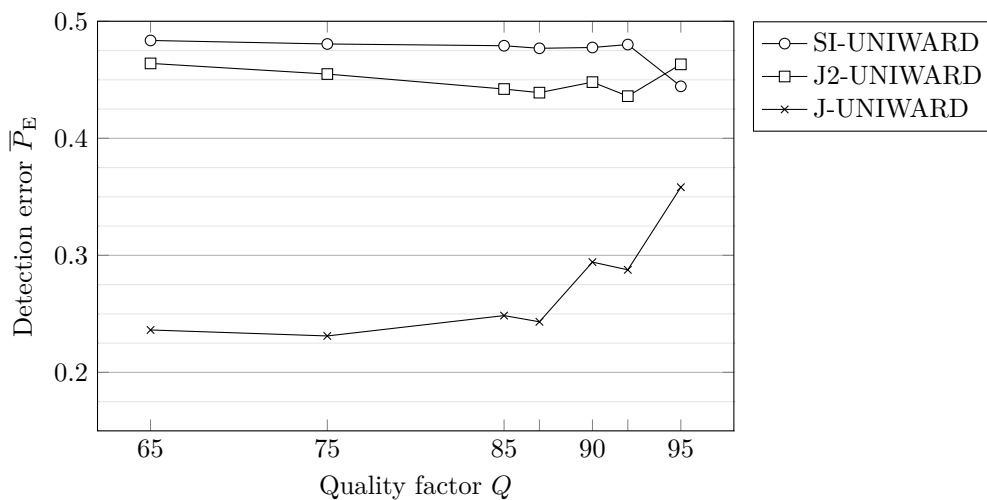


Figure 7.6.1: Empirical security \bar{P}_E of J2-UNIWARD as a function of the JPEG quality factor Q on BURSTbase. Comparison with previous art for $R = 0.2$ bpnzac.

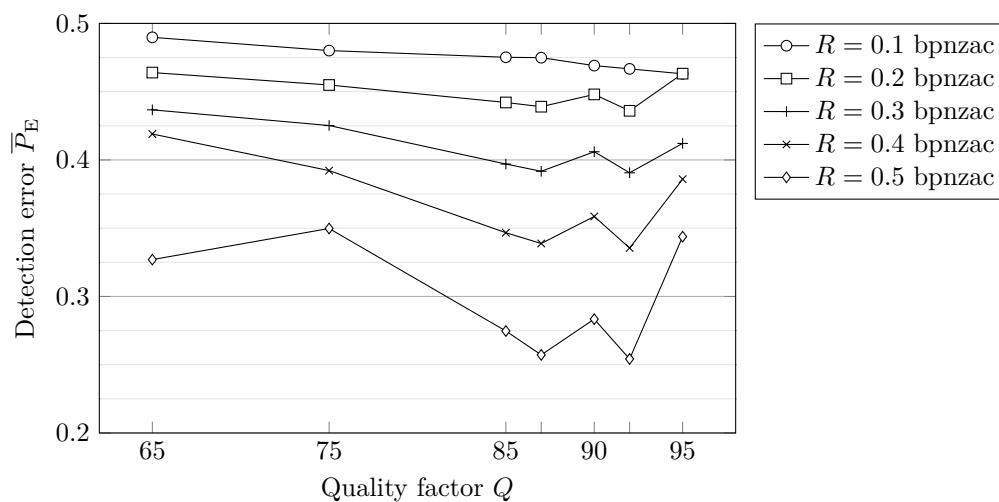


Figure 7.6.2: Empirical security \bar{P}_E of J2-UNIWARD as a function of the JPEG quality factor Q on BURSTbase. J2-UNIWARD \bar{P}_E for $R \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$ bpnzac, embedding simulated at rate-distortion bound.

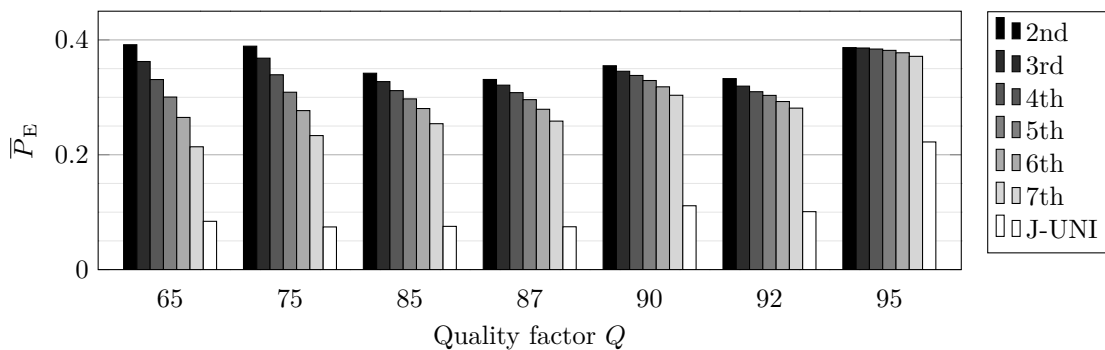


Figure 7.6.3: Empirical security \bar{P}_E of J2-UNIWARD when the k th closest image from each burst from BURSTbase was used as side-information. Payload $R = 0.4$ bpnzac.

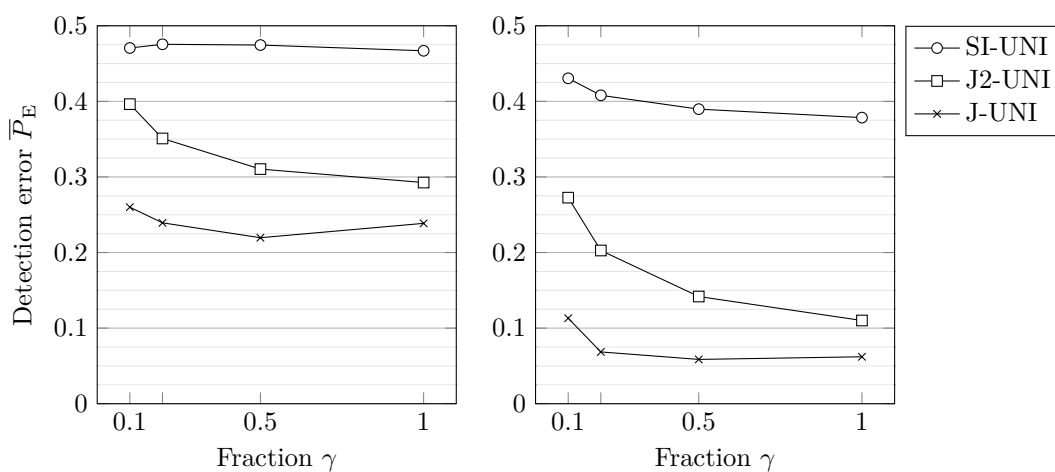


Figure 7.6.4: Empirical security \bar{P}_E of J-UNIWARD, J2-UNIWARD, and SI-UNIWARD as a function of γ best bursts from BURSTbaseH. JPEG quality factor 75, left column 0.2 bpnzac, right column 0.4 bpnzac.

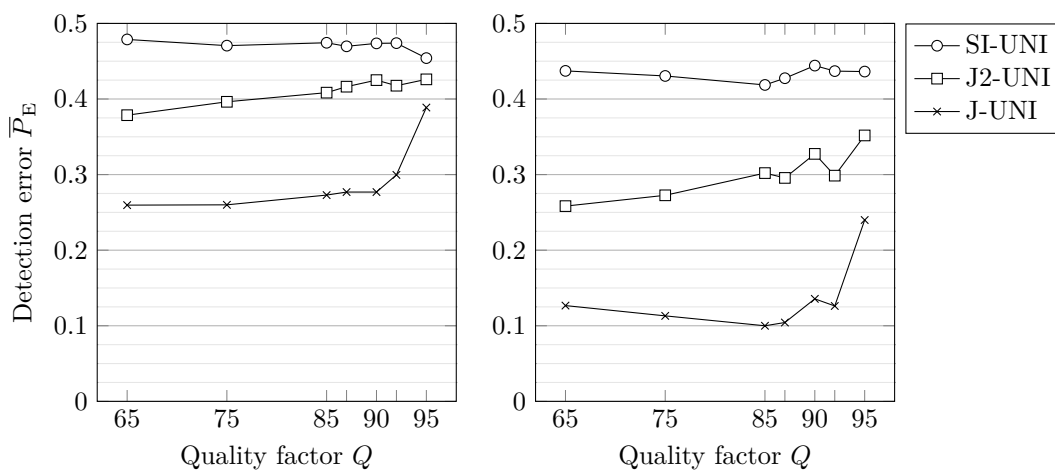


Figure 7.6.5: Empirical security \bar{P}_E of J-UNIWARD, J2-UNIWARD, and SI-UNIWARD as a function of JPEG quality factor Q for $\gamma = 0.1$ best bursts from BURSTbaseH. Left column 0.2 bpnzac, right column 0.4 bpnzac.

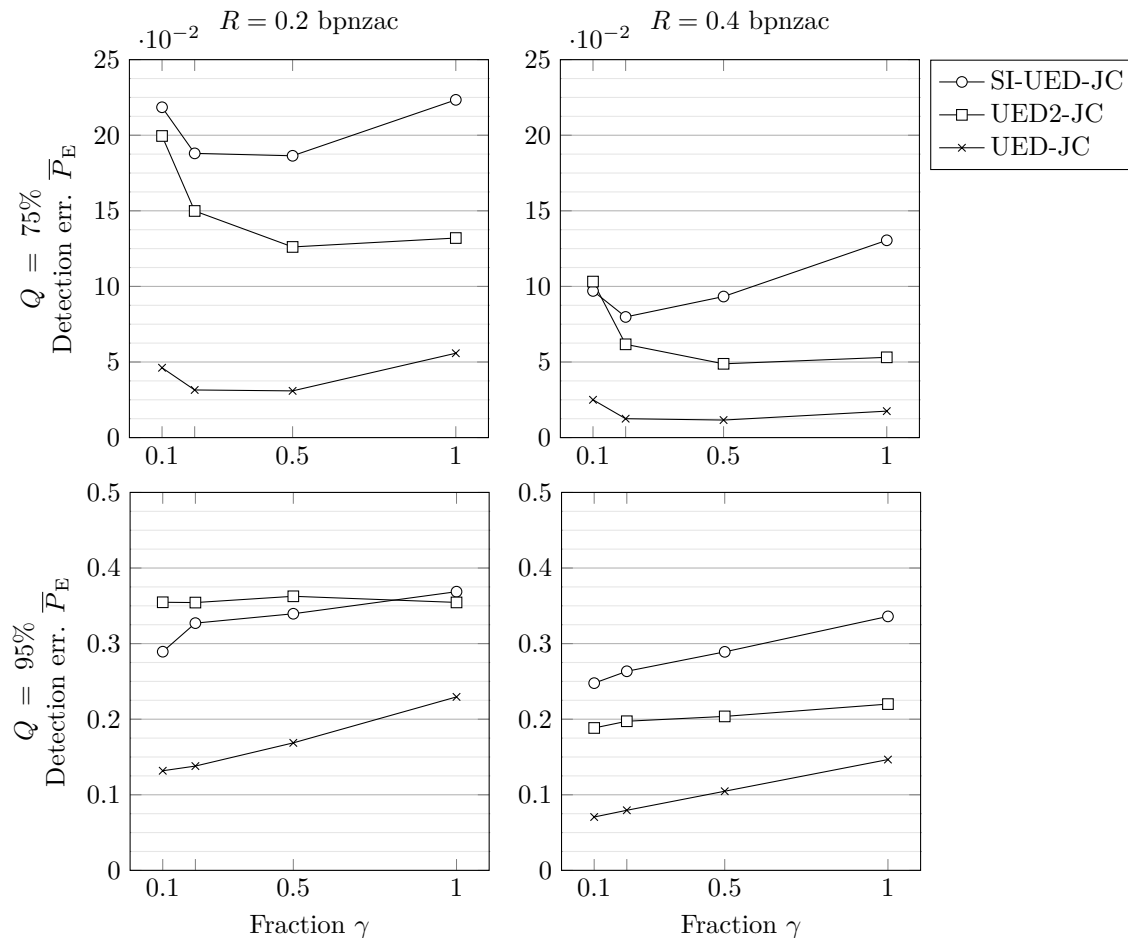


Figure 7.6.6: Empirical security \bar{P}_E of UED-JC, UED2-JC, and SI-UED-JC as a function of γ best bursts from BURSTbaseH for two JPEG quality factors and two payloads.

7.7 Conclusions

We introduce a novel steganographic method with side-information at the sender in the form of a second JPEG image of the same scene. The second exposure is used to infer the preferred direction of steganographic embedding changes in the first exposure (cover). This information is incorporated in any cost-based steganography by decreasing the embedding costs of such preferred changes with a multiplicative modulation factor.

The proposed methodology is first studied on J-UNIWARD costs with multiple exposures simulated by adding AWG noise to BOSSbase 1.01 images. This experiment revealed that, under such ideal conditions, the proposed method with two JPEG images of the same scene exhibits empirical security comparable with and sometimes even better than SI-UNIWARD informed by the uncompressed precover. This observation was attributed to the fact that for larger quality factors two JPEGs better inform the sender about the preferred embedding change direction than one uncompressed image.

To evaluate the proposed method in real-life conditions, we created two new datasets: BURSTbase with multiple exposures obtained by a tripod-mounted camera and BURSTbaseH with images shot with a hand-held camera. Detailed analysis of the differences between the two closest exposures from BURSTbase confirmed that they differ mostly by the acquisition noise, while images from BURSTbaseH are generally significantly much more spatially misaligned due to camera shake.

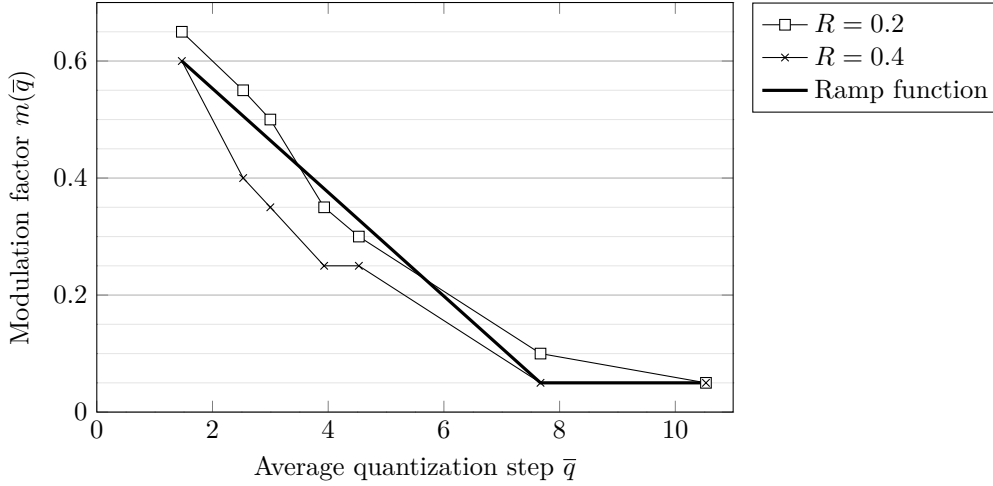


Figure 7.6.7: Modulation factor versus average quantization step \bar{q} (real acquisitions).

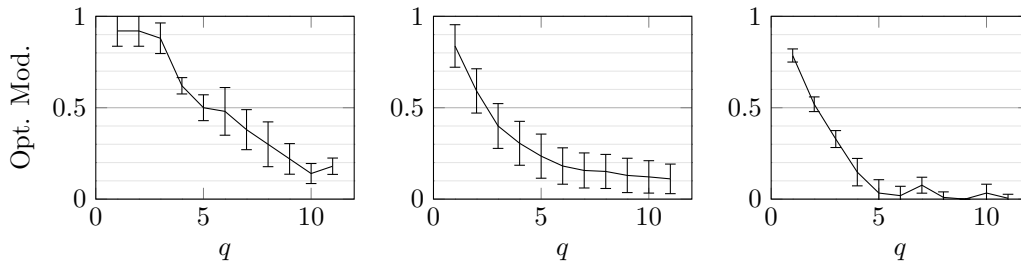


Figure 7.7.1: Optimal modulation factor $m_{ij}(q, R)$ as a function of the quantization step q for relative payload $R = 0.4$ determined by minimizing the Bhattacharyya distance between cover and stego distributions on generalized Gaussian models of DCT coefficients. Left: low frequency DCT modes (i, j) , $3 \leq i + j \leq 4$ (second and third minor diagonal), Middle: medium frequency DCT modes (i, j) , $5 \leq i + j \leq 10$, Right: high frequency DCT modes (i, j) , $11 \leq i + j \leq 16$.

For BURSTbase, we observed a quite significant increase in empirical security with respect to steganography with a single cover image that gracefully decreased with increased spatial misalignment between images. On the other hand, because of the comparatively larger misalignments between images shot with a hand-held camera the security improvement on BURSTbaseH was understandably smaller. However, we demonstrated for both J-UNIWARD and UED-JC, that the sender can still significantly gain on empirical security by rejecting a portion of “bad bursts”, which testifies about the practicality of the proposed embedding scheme.

Finally, the dependence of the experimentally determined modulation factor on the quality factor is justified using Monte Carlo simulations by adopting generalized Gaussian model for DCT coefficients and measuring the impact of cost modulation on statistical detectability in terms of the Bhattacharyya distance between cover and stego distributions. Optimal modulation derived from this model qualitatively matches the modulation obtained experimentally on real multiple exposures.

Further improvement is likely possible by optimizing the embedding cost modulation for the average grayscale of the DCT block because the acquisition noise amplitude depends on luminance. We plan to further study how the embedding should utilize more than two (quantized and unquantized) acquisitions of the same scene, possibly by extending the approach proposed in [22]. We anticipate that the proposed methodology will also work with multiple exposures obtained as consecutive frames from video clips. Finally, we note that the proposed approach is not limited to JPEG domain and will likely work for side-informed embedding in other domains [18].

Chapter 8

Natural steganography

Recently, Natural Steganography (NS), that relies on the concept of cover-source switching, has showed a great promise for constructing practical secure steganographic systems [5, 4]. The author showed that a high-capacity steganographic scheme with a rather low empirical detectability can be built when the developing process of a RAW sensor capture is sufficiently simplified, e.g., after gamma correction, bilinear downsampling, and 8-bit quantization of RAW images coming from a monochrome sensor. The impact of embedding is masked as an increased level of photonic (shot) noise due to a larger sensor gain (ISO setting). This is possible because in the raw domain the distribution of the shot noise is well approximated with the heteroscedastic model independently distributed on each photo-site. For a sufficiently simple developer, one can thus arrange the statistical properties of the stego signal to mimic the increased heteroscedastic noise and make the stego image statistically resemble an image taken at a higher ISO setting (a switch in the cover source). The feasibility of this concept was shown in [4] with raw images taken with a Leica M Monochrome Type 230 camera. In a follow-up work [5], the same author extended NS to more complex developers that involved gamma correction and bilinear downsampling as these processes allowed analytic derivation of the acquisition noise properties in the developed domain. In this chapter, we make NS more practical by introducing JPEG compression and also by treating the developer as a black box. Similarly to [20, 19], we use multiple instances of developed images in order to design our embedding strategy for each DCT coefficient.

8.1 Natural steganography in JPEG domain

In this section, we introduce the heteroscedastic noise model and study its properties after applying block Discrete Cosine Transform (DCT) as in JPEG compression.

8.2 Model in the spatial domain

In natural steganography, the stego signal added to the cover image acquired at ISO_1 is constructed to mimic the additional shot noise to make the stego image look like it was acquired at $ISO_2 > ISO_1$.

The shot noise values in the spatial domain are assumed to be independent realizations of random variables $N_{i,j}$ that follow the heteroscedastic model

$$N_{i,j}^{(1)} \sim \mathcal{N}(0, a_1 \mu_{i,j} + b_1) \quad (8.2.1)$$

where $\mu_{i,j}$ is the noiseless photo-site value at photo-site i, j , while (a_1, b_1) only depend on the ISO_1 sensitivity and the specific sensor.

The acquired photo-site sample $x_{i,j}^{(1)}$ is thus a realization

$$x_{i,j}^{(1)} = \mu_{i,j} + n_{i,j}^{(1)}, \quad (8.2.2)$$

of a Gaussian variable

$$X_{i,j}^{(1)} \sim \mathcal{N}(\mu_{i,j}, a_1\mu_{i,j} + b_1). \quad (8.2.3)$$

Because the sum of two independent normally distributed random variables is also normally distributed with the mean and variance the sum of means and variances of both variables, we can write that at ISO_2 the photo-site value is given by $x_{i,j}^{(2)} = x_{i,j}^{(1)} + s_{i,j}$ where $S_{i,j}$ is a random variable representing the stego signal necessary to mimic the image captured at ISO_2 :

$$S_{i,j} \sim \mathcal{N}(0, (a_2 - a_1)\mu_{i,j} + b_2 - b_1). \quad (8.2.4)$$

Assuming that the observed photo-site is close to its expectation, $\mu_{i,j} \approx x_{i,j}^{(1)}$, the photo-site of the stego image is distributed as:

$$\begin{aligned} Y_{i,j} &\sim \mathcal{N}(\mu_{i,j}, a_1\mu_{i,j} + b_1 + (a_2 - a_1)\mu_{i,j} + b_2 - b_1) \\ &\sim X_{i,j}^{(2)}. \end{aligned} \quad (8.2.5)$$

The distribution of the stego signal in the continuous domain takes into account the statistical model of the shot noise estimated for two ISO settings, ISO_1 and ISO_2 , using the procedure described in [4]. The work presented in [5, 4] shows that for monochrome sensors, this model in the spatial domain can be used to derive the distribution of the stego signal in the spatial domain after quantization, gamma correction, and image downsampling using bilinear kernels. We next study the properties of the acquisition noise after DCT.

8.2.1 Model in the DCT domain

We now compute the joint-distribution of the heteroscedastic noise in the DCT domain. This mathematical derivation can be used for a specific practical scenario of image development when a RAW image coming from a monochrome sensor is directly transformed into a JPEG image. To a certain extent, the derivations are also valid when gamma correction is performed before the DCT transform.

Since the stego signal in the spatial domain follows the normal distribution 8.2.4 and since the DCT is linear, the stego signal in the DCT domain, S_{DCT} , follows a 64-dimensional multivariate normal distribution

$$S_{\text{DCT}} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_{\text{DCT}}). \quad (8.2.6)$$

In order to compute the covariance matrix \mathbf{C}_{DCT} of the stego signal \mathbf{S} , it is convenient to use vector notation by transforming the matrix $\mathbf{S} \in \mathbb{R}^{8 \times 8}$ into a vector $\mathbf{s} \in \mathbb{R}^{64}$ by concatenating the columns. The transpose operation \mathbf{S}^t is then equivalent to the multiplication $\mathbf{T}\mathbf{s}$, by \mathbf{T} given by:

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & \dots & & & & & & & \\ 0 & \dots & \dots & 1 & 0 & \dots & & & & \\ & & & & & & 1 & 0 & \dots & \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & & \\ 0 & 1 & \dots & & & & & & & \\ & & & & 0 & 1 & \dots & & & \\ & & & & & & & 0 & 1 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}. \quad (8.2.7)$$

Consequently, the 8×8 matrix DCT transformation \mathbf{A} (1.5.4) is transformed into a 64×64 matrix \mathbf{A}_v given by:

$$\mathbf{A}_v = \begin{bmatrix} \mathbf{A} & 0 & \cdots & & & & & \\ 0 & \mathbf{A} & 0 & \cdots & & & & \\ \vdots & 0 & \mathbf{A} & 0 & \cdots & 0 & & \\ & \cdots & 0 & \mathbf{A} & 0 & \vdots & & \\ & & \vdots & 0 & \mathbf{A} & 0 & \vdots & \\ & & 0 & \cdots & 0 & \mathbf{A} & 0 & \vdots \\ & & & & \cdots & 0 & \mathbf{A} & 0 \\ & & & & & \cdots & 0 & \mathbf{A} \end{bmatrix}. \quad (8.2.8)$$

The vector form of the DCT (1.5.3) finally becomes

$$\text{DCT}_v(\mathbf{s}) = \mathbf{A}_v \mathbf{T} \mathbf{A}_v \mathbf{T} \mathbf{s} = \mathbf{B} \mathbf{s}, \quad (8.2.9)$$

where $\mathbf{B} = \mathbf{A}_v \mathbf{T} \mathbf{A}_v \mathbf{T}$.

Now

$$\mathbf{C}_{\text{DCT}} = \mathbf{E}[\mathbf{B} \mathbf{S} \mathbf{S}^t \mathbf{B}^t] = \mathbf{B} \text{Cov}(\mathbf{S}) \mathbf{B}^t, \quad (8.2.10)$$

and $\text{Cov}(\mathbf{S})$ denotes a diagonal matrix with diagonal elements equal to $\text{Var}(s_{i,j}) = (a_2 - a_1)x_{i,j} + b_2 - b_1$.

8.2.2 Discussion

Even though the stego signal (the sensor noise) is independent in the spatial domain, it follows a general multivariate normal distribution in the DCT domain. Thus, ideally the embedding should take into account dependencies that exist between DCT modes within each 8×8 block. Note that in this setting, no dependencies exist between DCT blocks. This model consequently enables us to explicitly compute the variance of the stego signal for each DCT mode and the covariance between DCT modes.

To better understand the nature of the dependencies between DCT coefficients, we sample the stego signal directly in the DCT domain and observe the dependencies before and after JPEG quantization.

In Figure 8.2.1, we visually compare sampled blocks before (even columns) and after quantization (odd columns) with the standard JPEG quantization matrix corresponding to quality factor 95. Note that the quantization process is here $\text{quant}(x) = \Delta \times \text{round}(x/\Delta)$, where Δ is the quantization step.

For different spatial cover blocks represented in the first row, blocks of stego signals are sampled in the DCT domain (the second row, S) using (8.2.6) or in the spatial domain (the third row, S^s) using (8.2.4) and then transformed.

While the first two spatial blocks with horizontal/vertical directions produce vertical/horizontal correlations in the DCT domain, neither the checkerboard or the constant block produce significant correlations (for the constant block, the signal must be i.i.d. since it is the DCT of an i.i.d. signal). The diagonal blocks produce slightly correlated stego signals which are more pronounced for the minor-diagonal block. Comparing the second and third rows enables us to verify that the sampling either in the spatial domain or directly in the DCT domain exhibits similar dependencies. The odd columns illustrate the effect of JPEG quantization, which tends to reduce the dependencies between coefficients by nullifying high frequencies.

This experiment also shows that the dependencies in the DCT domain, contrary to the spatial domain (see [17, 67]), heavily depend on the cover block content. However, we shall see in Section 8.5 that for large quantization regimes not taking into account the dependencies does not significantly impact the detectability of embedding.



Figure 8.2.1: First row: spatial 8×8 blocks. The second and third rows are samples where, for the purpose of comparison, the signal S is sampled directly in the DCT domain by sampling a 64-dimensional multivariate Gaussian distribution while S^s is sampled in the spatial domain and then converted to DCT coefficients.

8.3 Overview of the algorithms

We remind the reader that our goal is to develop a NS method capable of embedding messages in JPEG images by utilizing a cover source switch from ISO_1 to a larger ISO_2 . The first step in building such a steganographic method is to estimate the parameters of the heteroscedastic sensor noise for the specific camera that will be used for communication and for both ISO settings: (a_1, b_1) and (a_2, b_2) . This has been executed by taking images of a gray gradient as explained in [5, 4] and in Section 8.4.2. Having estimated these four parameters, from Eqs. (8.2.2)–(8.2.5) the stego photo-site is obtained from the cover photo-site $X_{i,j}^{(1)} \sim \mathcal{N}(\mu_{i,j}, a_1\mu_{i,j} + b_1)$ by adding to it a realization of $S_{i,j} \sim \mathcal{N}(0, \sigma_{i,j}^2)$, where $\sigma_{i,j}^2 = (a_2 - a_1)\mu_{i,j} + b_2 - b_1 \approx (a_2 - a_1)x_{i,j} + b_2 - b_1$.

In order to perform a cover-source switch on a raw pre-cover image, we adopt special rules for photo-sites saturated at 2^r , where r is the dynamic range of the sensor, typically 12 or 14 bits. Our strategy is similar to the one presented in [83]. The photo-site value $y_{i,j}^{(2)}$ after the cover-source switch mimicking sensitivity ISO_2 is:

$$y_{i,j}^{(2)} = \begin{cases} 2^r & \text{if } x_{i,j}^{(1)} = 2^r \text{ or } x_{i,j}^{(2)} > 2^r \\ 0 & \text{if } x_{i,j}^{(2)} < 0, \\ x_{i,j}^{(2)} & \text{else.} \end{cases}, \quad (8.3.1)$$

To obtain a better insight into the role of our simplifying assumptions and the effect of estimating the noise in the DCT domain and to establish upper bounds on the detection error, we investigate five different approaches explained below. The security of these approaches is evaluated by building a classifier distinguishing between the images from both classes after JPEG compression. To get closer to a practical scheme, we added the case when the developer is treated as a black box and the noise distribution is estimated from quantized DCT coefficients in the developed domain using Monte-Carlo (MC) sampling.

1. **Pure, unmodified images.** As a baseline experiment, no modifications were introduced to

the ISO_1 images and a classifier was trained to recognize between these and ISO_2 images after JPEG compression.

2. **Simulated noise.** Before any developing, we added a simulated heteroscedastic noise to the raw images at ISO_1 using the clipping rule 8.3.1. Provided the parameters of the heteroscedastic noise at each ISO setting were precisely estimated and under the assumption that we steganalyze using the best possible detector, this result could serve as an upper bound on the security of the method – the detection error. Note, however, that it does not correspond to any practical embedding.
3. **Monte-Carlo estimate of the variance.** We add 300 independent realizations of the heteroscedastic noise estimated as in Approach 2 to the raw pre-cover image acquired at ISO_1 , $x_{i,j}^{(1)}$, employing again 8.3.1, develop the images, and apply the DCT. Then, we independently estimate the mean and the variance of each DCT coefficient from the MC samples. To obtain the stego JPEG file, we add independent realizations of such random variables to the unquantized DCT coefficients of the developed pre-cover and round to integers to obtain the JPEG DCT coefficients of the stego image.
4. **Monte-Carlo estimate of the pmf.** To remove the Gaussianity assumption, we use the MC samples to directly estimate the probability mass function of each *rounded* DCT coefficient. Then, we sampled from this distribution for each coefficient to obtain the final quantized DCT coefficient from the stego image.
5. **SI-UNIWARD.** For comparison with the current state of the art, we embedded all images also with SI-UNIWARD [51] with the same average embedding rate or lower if the embedding rate was over 1 bit per DCT coefficient, the maximal payload of SI-UNIWARD.

We now proceed with a formal description of the NS method that hides messages in the JPEG file given a pre-cover in a RAW format. The sender basically uses the pre-cover in the RAW format to *estimate* the Gaussian variance from MC samples (Approach 3) and then *compute* the pmf of the quantized stego DCT coefficient, or to *directly estimate the pmf* of each quantized DCT coefficient from the stego image (Approach 4). The advantage of Approach 4 is that it can be applied for realistic (i.e., complicated) developers that output more complex (non-Gaussian) shot noise distribution.

Denoting the pmf of a fixed quantized stego DCT coefficient as q_k , the payload that could embed at this coefficient is

$$-\sum_k q_k \log_2 q_k \text{ bits}, \quad (8.3.2)$$

the entropy of the pmf. For Approach 3, given the variance ω^2 of a specific unquantized stego DCT coefficient with quantization step Δ , q_k corresponds to the k th bin in a quantized Gaussian distribution $\mathcal{N}(0, \omega^2/\Delta^2)$: In contrast, in Approach 4 the pmf q_k is estimated directly from the 300 MC samples by computing an empirical histogram.

The actual message embedding can be implemented in practice using the multi-layered version of syndrome-trellis codes [28], which essentially allow embedding payload close to the entropy (8.3.2) at each DCT coefficient. We would also like to stress that the total payload that can be embedded is determined by the two ISO values and is equal to the sum of entropies (8.3.2) over all DCT coefficients in the JPEG image. The payload size also depends on the JPEG quality factor and the content of the image. Should the sender need to embed a shorter payload, the message could be padded with random bits. Alternatively, the sender could also switch to a smaller value of ISO_2 . On the contrary, if the payload to be embedded is larger than the admissible payload offered by the cover-source switch, the sender would have to use a larger value of ISO_2 or split the payload across multiple images.

Note that the proposed NS method may, depending on the ISO settings, embed more bits in an image than SI-UNIWARD. For a fair comparison, for SI-UNIWARD we therefore embedded the same relative payload in each image (in bits per pixel rather than per non-zero AC DCT coefficient) obtained by averaging the payload embedded by Approach 3 over the whole database.



Figure 8.4.1: Comparison between distributions of shot noise coming from different sensors. Histograms are computed from the photo-site values of one given channel (for color sensors) on a uniform patch. Dash lines represent Gaussian distributions with the same mean and variance as the histogram.

8.4 Database acquisition and shot noise distribution

In this section, we describe how we acquired the image databases needed to benchmark Natural Steganography, and discuss the statistical properties of the photonic noise distribution for different sensors.

8.4.1 Acquisition process

In contrast to the widely used BOSSBase [1] used in steganography and steganalysis, for benchmarking Natural Steganography the datasets need to be built with special care. Because the goal of the embedding is to mimic a shot noise at ISO_2 from images captured at sensitivity $ISO_1 < ISO_2$, two sets of images have to be acquired: one set at ISO_1 that will be used for the embedding and another set at ISO_2 that will represent the set of cover images. The steganalyst will then compare stego images coming from the set at ISO_1 and cover images acquired at ISO_2 . We assume here that the sender will modify or remove the ISO setting from the stego images because the steganalyst could potentially utilize the discrepancy between the noise level in stego images and the ISO setting.

It is important to mention that in order to build a classifier that will only detect the steganographic

embedding, the two sets of images have to represent identical content.

During our acquisition campaign, we consequently paid attention to use constant acquisition parameters: the same focus, the same scene with the use of a tripod, the same white balance, and the same aperture to have only the sensitivity and the exposure time fluctuating. We realized the importance of this step when at one point we slightly modified the focus between the two sets, which resulted in increased classification accuracy due to the ability of the classifier to distinguish between content sharpness rather than the steganographic changes.

To alleviate the labor associated with the acquisition of these databases, we took around 200 raw images¹ at each setting and subsequently cropped each picture to non-overlapping 512×512 images to generate around 10,000 crops in each set. The development of the raw image was done using the 'dcraw' Linux command line with the parameters '-k 0' to obtain the same darkness level for each set, '-g 1 1' to disable gamma correction, '-W' to obtain the same white balance for each set and '-6' to generate 16-bit ppm/pgm images instead of 8-bit images.

We ran these acquisition campaigns on three sensors: one monochrome CCD sensor from the Leica M Monochrome Type 230 camera, one color CCD sensor from the Leica M9 camera, and one CMOS sensor from the Z CAM E1 action camera. Note that the Leica M Monochrome and Leica M9 cameras have identical sensors but the Monochrome does not have a Bayer CFA. The databases built from Leica cameras are images from different scenes, shot using a tripod at different ISO settings (320, 1000, and 1250 for the Monochrome), the databases from the E1 sensor have been captured using a rotating platform in a room filled with different objects.

8.4.2 Shot noise distribution

We were very surprised to notice that, using exactly the switch 8.3.1, the detectability of the M9 sensor was extremely high compared to the detectability of the Monochrome sensor. After some investigations, we noticed that the shot noise on the M9 sensor does not have a Gaussian distribution at high ISO sensitivities.

This phenomenon is illustrated in Figure 8.4.1, where we compare the shot noise distributions for both different sensors. To estimate the distribution of the sensor noise, we used here a simple but robust technique: we shot a white wall at a distance of 1m from the sensor and out of focus in order to obtain an image with average constant illumination. We then computed the histogram from the RAW image, using the photo-site values² of a 100×100 patch centered on the image to avoid vignetting for one given color channel (we checked that our observations were consistent for all channels and for different patches). While the histogram of the noise taken at ISO 160 (Figure 8.4.1a) corresponds to a Gaussian signal, as soon as the ISO sensitivity is increased, the shot noise distribution becomes strongly non-Gaussian. For example, at ISO 2000 (Figure 8.4.1b) the distribution has heavy tails and the Gaussian assumption is rejected. The sensor capture from another M9 camera (see Figure 8.4.1c) shows that this artifact does not come from the specific camera. It is also not specific to the manufacturer since Figure 8.4.1e depicts a Gaussian distribution for the Leica Type M262 CMOS sensor. What is even more surprising is the fact that the Leica M Monochrome Type 230 camera (an M9 version without Bayer CFA) does not exhibit this artifact (Figure 8.4.1d).

In the end, we selected two sensors:

1. Leica M Monochrome camera to directly acquire grayscale images because this sensor does not have demosaicking applied during the development process and it also exhibits normally distributed shot noise,

¹The exact number depends of the sensor resolution.

²They are, for example, easily accessible using the Rawkit Python module [72].

2. Z CAM E1 to acquire color images because of a Gaussian shot noise distribution at high ISO (Figure 8.4.1f). This action camera has a time-lapse mode, which enables fast acquisition of new images.

Note that the M9 Leica was the only camera for which we observed a rather peculiar non-Gaussian shot noise.

8.5 Experiments

In this section, we subject the proposed natural steganography algorithms to tests on images taken with the CCD sensor from the Leica M Monochrome Type 230 and the CMOS sensor from the Z CAM E1 action camera. Images coming from the monochrome sensor are named MonoBase and are composed of 10,320 512×512 images in 16-bit PGM format developed using the command "`dcraw -k 0 -6 -W -g 1 1`". Since there is no demosaicking on this sensor, this format is very close to the RAW format. Images coming from the E1 sensor are named E1Base and are generated from 200 DNG images that are developed and cropped to provide 10,800 512×512 images. Both E1Base and MonoBase can be downloaded from [2] and [6].

The switches used for MonoBase are from ISO 320 to ISO 1000 and for E1Base from ISO 100 to ISO 200. Because the E1 sensor is smaller and cheaper, the power of the stego signal for both sensors is of the same order of magnitude. The parameters (a, b) used to realize the switches are $(4.3, 3801)$ for the MonoBase and $(0.9, -800.0)$ for the E1Base. Note, however, MonoBase images use values coded between $[0; 2^{16} - 1]$ due to the PGM format while E1Base values are between $[0; 2^{14} - 1]$ due to the sensor dynamic range. We believe that the negative value of b for E1Base is due to a bias correction that is ISO dependent and coded inside the chipset.

The detection error is evaluated as the minimal total classification error probability under equal priors, $P_E = \min_{P_{FA}} \frac{1}{2}(P_{FA} + P_{MD})$, with P_{FA} and P_{MD} standing for the false-alarm and missed-detection rates, using a low complexity linear classifier [14]. The JPEG images were steganalyzed with the SRM [39], GFR [85], DCTR [49] and cc-JRM [62] feature sets. For improved readability, we report only the best detection (lowest error) over these four feature sets. All reported errors are averaged over ten different splits of the database into equal sized training and testing sets. The largest measured standard deviation over the ten splits was 0.0097.

8.5.1 Results on MonoBase

In Figure 8.5.1, we show the detection errors for Approach 1, 2, and 5 (Pure, Simulated, and SI-UNIWARD) as a function of the JPEG quality factor. SI-UNIWARD embeds into each image the same payload obtained as the average payload over the whole database for Approach 4 using Eq. (8.3.2) (see Table 8.1).

In Table 8.2, we list the detection errors for all five approaches. The fact that the detection errors for Approach 3 and 4 are very close to the errors of Approach 2, the simulated acquisition noise that should preserve all dependencies among DCT coefficients, validates the simplifications of ignoring the dependencies during embedding used by Approach 3 and 4.

Our method clearly has a great promise, particularly w.r.t. the current state of the art in side-informed steganography, the SI-UNIWARD. For a JPEG QF of 95, the practical security of NS using MC-pmf (Approach 4) leads to $P_E \simeq 40\%$ for an average embedding rate of 2.36 bpnzac when $P_E \simeq 0\%$ for SI-UNIWARD. Note, however, that when making this comparison, it should be taken into account that NS needs the RAW file while SI-UNIWARD only needs the non-rounded values of DCT coefficients that are computed from the developed image, which is a substantially less extensive side-information.

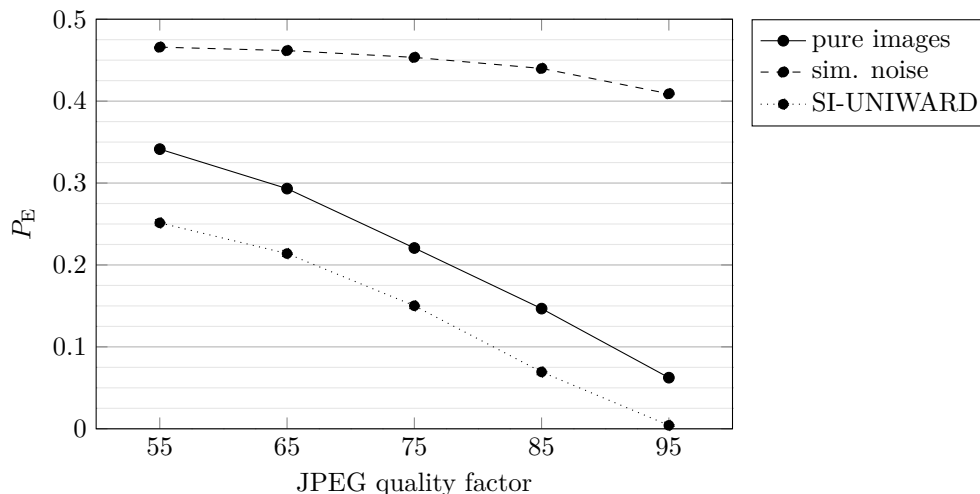


Figure 8.5.1: Detection error P_E for the pure, simulated noise, and SI-UNIWARD (Approaches 1, 2, and 6) for a switch from ISO 320 to ISO 1000 on MonoBase as a function of the JPEG quality factor. Approaches 3–5 exhibit security that is approximately equal to that of Approach 2 (simulated noise), see Table 8.2.

QF	Average embedding rate (bpp)	Average embedding rate (bpnzac)
55%	0.0158	0.2315
65%	0.0277	0.3474
75%	0.0462	0.4734
85%	0.1127	0.8607
95%	0.5300	2.3650

Table 8.1: Average payload in bits per pixel per MonoBase image embedded by Approach 3.

It is also interesting to note that taking into account the dependencies between DCT coefficients within the same block has virtually no impact on the empirical security for this particular sensor. This is probably due to the fact that the DCT tends to generate uncorrelated coefficients whose dependencies are rather weak and/or not captured by the employed steganalysis features in this case. The analysis performed in Subsection 8.2.1 also shows that dependencies have to be taken into account only when the content inside a block is structured (edges, patterns). Such blocks will not be as common in high-resolution images investigated in this subsection.

Another important point is that Approach 4, which treats the developing process as a black box and has access only to rounded DCT coefficients, is basically as secure as the other approaches. This indicates a path that can be taken for other, more advanced and more realistic developers for this sensor.

8.5.2 Results on E1Base

We now evaluate the empirical security of NS in the JPEG domain for images coming from a color sensor. In contrast to monochrome sensors, after development the stego signal becomes dependent due to demosaicking.

Table 8.3 contains the detection results for all five embedding approaches for the E1 sensor. Compared to images from the monochrome sensor, the empirical security of Approach 2 (Simulated

QF	Pure	Sim.	MC var.	MC pmf	SI-UNI
App	1	2	3	4	5
55%	.3414	.4659	.4672	.4716	0.2514
65%	.2932	.4617	.4610	.4601	0.2139
75%	.2207	.4534	.4511	.4486	0.1502
85%	.1467	.4399	.4449	.4438	0.0694
95%	.0624	.4090	.4112	.4093	0.0042

Table 8.2: Minimum detection error when steganalyzing the NS in MonoBase with SRM, GFR, DCTR, and cc-JRM feature sets for five different approaches and a range of JPEG quality factors for a switch from ISO 320 to ISO 1000.

Noise) decreased by about 10% but the P_E remained above 30% for all QFs. However, the security of Approaches 3 and 4 ('MC var' and 'MC pmf'), is much lower especially for high QFs. We recommend using NS with Approaches 3 and 4 only for quality factors lower than 65 for which $P_E \geq 25\%$ while the average embedding rate is still high with 2.5 bpnzac, see Table 8.4. The fact that the empirical security of 'MC var' is slightly larger than for 'MC pmf' is probably due to the fact that the number of samples used during Monte Carlo sampling (300), is not enough to accurately estimate the theoretical pmfs.

Note that Approach 1 (pure images) is also more detectable than for the monochrome sensor despite the gap between the two ISO sensitivities being similar. This is likely due to the dependencies introduced by demosaicking for images from the E1 sensor.

Comparing the embedding rates in bpnzac for the two databases (see Tables 8.1 and 8.4), while for MonoBase the rates increase from 0.2 to 2.36 bpnzac with increasing QF, they are nearly constant for the E1Base and always larger than 2 bpnzac. This can be explained by the fact that the demosaicking applied to E1 images increases the number of small DCT coefficients before quantization, especially in high frequencies. Thus, after quantization the number of non-zero coefficients is larger for MonoBase than for E1Base and the rate in bpnzac is correspondingly smaller.

QF	Pure	Sim.	MC var.	MC pmf	SI-UNI
App	1	2	3	4	5
65%	0.1168	0.3426	0.2433	0.1920	0.1168
75%	0.0937	0.3385	0.1757	0.1473	0.0957
85%	0.0732	0.3350	0.0881	0.0715	0.0752
95%	0.0595	0.3077	0.0056	0.0023	0.0032

Table 8.3: Detection error P_E when steganalyzing the NS in E1Base with SRM, GFR, DCTR, and cc-JRM feature sets for different approaches and JPEG quality factors for ISO switch 100 to 200. Note that the embedding capacity of SI-UNIWARD is limited to 1 bpnzac.

8.5.3 Discussion

In this subsection, we attempt to explain the striking difference in empirical security of NS (Approach 4) when applied to the monochrome sensor and the color sensor. As analyzed in Subsection 8.2.1, depending on the block content, intra-block dependencies exist between DCT coefficients of the stego-signal. Furthermore, inter-block dependencies also exist between DCT coefficients from neighboring blocks due to the demosaicking process. Note, however, that the natural dependencies among

QF	Average embedding rate (bpp)	Average embedding rate (bpnzac)
65	0.0330	2.5556
75	0.0618	2.2849
85	0.1493	2.3336
95	0.5671	2.8488

Table 8.4: Average embedding rate for Approach 4 (MC pmf), E1Base.



Figure 8.5.2: Experiment with a synthetic RAW image: co-occurrences of pixel pairs of adjacent pixels belonging either to adjacent blocks (a), to the same block (b), to adjacent blocks (c) or same block (d) after simulating noise that preserves dependencies only at the block-wise level.

neighboring pixels do not impact *per se* the dependency of the stego signal since the shot noise is independent from the photo-site values.

In order to determine whether the loss of security is due to not preserving intra or inter-block dependencies among DCT coefficients, we conducted two experiments:

Experiment 1: We generated the stego images to preserve intra-block dependencies of the stego noise in each each DCT block. In particular, each block came from one specific realization of Approach 2 but different blocks came from different realizations. Calling this strategy 'Sim block-wise', its practical security is compared with Approach 2 in Table 8.5, which shows that the empirical security is even lower than the security of 'MC-pmf' (Approach 4). This means that the loss of security of 'MC-pmf' and 'MC-var' must be due to violating inter-block dependencies rather than not preserving intra-block dependencies.

Experiment 2: To confirm this hypothesis, we next used a synthetic RAW image with all photo-site values from even columns equal to 8192 and the values of odd columns equal to 5461 (see [41]). This content was selected purposely with harsh high-frequency discontinuities in order to magnify the errors the interpolation algorithm will introduce. The demosaicking has to predict the missing color components. After adding stego noise with arbitrary (a, b) parameters, we then apply both Approach 2 and Approach 'Sim block-wise' without considering the JPEG quantization step. Since co-occurrence matrices are sensitive to steganographic embedding – they are for example the basis of SPAM or SRM feature sets [39, 74] – we plot in Figure 8.5.2 the co-occurrence of the red color

component of adjacent pixels after development. These sets of pairs of adjacent pixels are either located on the boundaries of two adjacent DCT blocks for Approach 2 (Figure 8.5.2a) and for Approach 'Sim block-wise' (Figure 8.5.2c), or in the middle of one DCT block for Approach 2 (Figure 8.5.2b) and for Approach 'Sim block-wise' (Figure 8.5.2d). Note that the demosaicking process has a profound effect on inter-block dependencies. After Approach 2, which does not violate the demosaicking step, the co-occurrences are nearly identical for different pixel locations but if we compare Approach 2 and Approach 'Sim block-wise' for pixels located across the boundaries of DCT blocks (Figure 8.5.2a vs Figure 8.5.2c) the co-occurrences become very different because 'Sim block-wise' only preserves intra-block dependencies.

QF	Sim.	MC pmf	Sim block-wise
65	0.3426	0.1920	0.1511
75	0.3385	0.1473	0.1093
85	0.3350	0.0715	0.0274
95	0.3077	0.0023	0.0005

Table 8.5: Comparison between Approach 2 (simulated noise), which preserves intra-block dependencies, Approach 4 (independent embedding at each DCT coefficient), and simulated noise sampled independently for each DCT block.

We conclude from these two experiments that the low empirical security of Approach 'MC pmf' is due to the fact that it does not preserve inter-block dependencies between DCT coefficients. This conclusion is supported by the fact that preserving intra-block dependencies but not inter-block dependencies does not improve security (Experiment 1), and also by the fact that discrepancies form in co-occurrences of adjacent pixels from neighboring blocks (see Experiment 2).

8.6 Conclusions and perspectives

Natural steganography is an embedding paradigm in which sensor noise is added to a RAW (cover) image capture to embed the secret message, making thus the stego image look as if it was acquired at a higher ISO setting. The novel idea explored in this chapter is extending NS to allow embedding of the message in quantized DCT coefficients in a JPEG file and with more complex RAW format developers. The most promising embedding algorithms studied in this chapter estimate the distribution of quantized stego DCT coefficients using Monte-Carlo sampling by adding sensor noise to the RAW cover capture, developing the images, and then JPEG compressing. This approach is free of any modeling assumptions on the distribution of stego image DCT coefficients and can also be used with more complex (e.g., more realistic) developers.

Our findings can be summarized as follows:

- For images acquired by monochrome sensors, such as the Leica M Monochrome Type 230, when adopting a linear development, NS can embed large payloads (more than 2 bpnzac) with high empirical security ($P_E > 0.4$) for a wide range of JPEG quality factors. We experimentally verified that making independent embedding changes to DCT coefficients does not significantly impact the security.
- When the same strategy (independent embedding in each DCT coefficient, linear development) is applied to images from a color sensor, the empirical security of NS becomes low. Further analysis showed that this loss of security can be attributed to the failure of the embedding algorithm to preserve inter-block dependencies between DCT coefficients introduced by the demosaicking process.

In our future work, we plan to address the problem of statistically modeling and better preserving inter-block dependencies between DCT coefficients for color sensors and move towards more advanced development pipelines. To this end, generative models, such as the PCA or the Generative Adversarial Networks using strategies similar to [88], could be used.

Chapter 9

Conclusion

Modern steganography does not hide information only in the non-deterministic part of digital images. This is evidenced by the success of content-adaptive approaches that directly exploit steganalyst's inability to model the signal well enough, and hide a part of the message in sufficiently complex but deterministic signal.

This dissertation gives some power back to the steganalyst. The same local complexity estimator used to hide the message can be used to focus rich-model style feature sets on the areas most likely affected by steganography and improve the detection. While the idea itself is quite straightforward, the proper implementation is not. The steganalyst should correctly propagate the information about the selection channel to the same domain where the detection statistic is computed – the residual domain. This can prove computationally expensive but improves the detection of content-adaptive schemes. The success of the statistics describing the impact of embedding on residuals proposed in Chapters 3 and 4 is proven not only in the experiments presented in this dissertation, but also in the fact the state-of-the-art convolutional network based detectors [93, 10] also use these statistics.

The absence of a quality model is one of the most significant challenges for not only steganalysts but also for steganographers. New schemes that introduce novel, better performing models are rare and difficult to find. In this dissertation, we introduce several ways how to utilize different types of private side-information to improve the models and also show that the previously popular cost modulation is not, in fact, optimal and propose new, model justified approach.

Appendix A

SI-UNIWARD as published

In this appendix, we point out that the costs of SI-UNIWARD as defined in [51] are incorrect, state how the costs are computed in the SI-UNIWARD implementation, how they should be defined, and how they are related to the costs of J-UNIWARD.

Denoting the block-wise DCT with J , the (non-rounded) DCT coefficients of the uncompressed precover image will be denoted as $J(\mathbf{P}) = \mathbf{U} \in \mathbb{R}^{n_1 \times n_2}$. The 2D array of quantized DCT coefficients is $\mathbf{X} = Q_{11}(\mathbf{U})$. Eqs. 5 and 6 in [51] define the cost of changing X_{ij} to $Y_{ij} = X_{ij} + \text{sign}(e_{ij})$ as

$$\rho_{ij}^{(\text{SI})}(\mathbf{X}) = D^{(\text{SI})}(\mathbf{X}, \mathbf{X}_{\sim ij} Y_{ij}), \quad (\text{A.0.1})$$

where $\mathbf{X}_{\sim ij} Y_{ij}$ stands for the matrix \mathbf{X} with only X_{ij} changed to Y_{ij} and J^{-1} is the block-wise inverse DCT (without rounding to \mathcal{I}_8). The non-additive distortion $D^{(\text{SI})}$ is defined as

$$D^{(\text{SI})}(\mathbf{X}, \mathbf{Y}) = D(\mathbf{P}, J^{-1}(\mathbf{Y})) - D(\mathbf{P}, J^{-1}(\mathbf{X})), \quad (\text{A.0.2})$$

$$D(\mathbf{A}, \mathbf{B}) = \sum_{b=1}^3 \sum_{u,v=1}^{n_1, n_2} \frac{|W_{uv}^{(b)}(\mathbf{A}) - W_{uv}^{(b)}(\mathbf{B})|}{\sigma + |W_{uv}^{(b)}(\mathbf{A})|}, \quad (\text{A.0.3})$$

where $W_{uv}^{(b)}(\mathbf{X})$ is the uv -th wavelet coefficient in subband b in image \mathbf{X} , σ is a stabilizing constant, and $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n_1 \times n_2}$. Using the wavelet kernel in b th subband, $\mathbf{W}^{(b)}$, the wavelet coefficients are computed using a convolution, $W_{uv}^{(b)}(\mathbf{A}) = (\mathbf{W}^{(b)} \star \mathbf{A})_{uv}$.

According to this definition, and in contrast with the claims made in [51], the costs (A.0.1) defined this way may become negative, which can be easily verified by implementing the formulas. The implementation of SI-UNIWARD available from the authors' web site uses a different formula, which always gives non-negative costs. The formula that exactly corresponds to the implementation of SI-UNIWARD should have been

$$\rho_{ij}^{(\text{SI})}(\mathbf{X}) = D^{(\text{SI})}(\mathbf{U}_{\sim ij} X_{ij}, \mathbf{U}_{\sim ij} Y_{ij}). \quad (\text{A.0.4})$$

We now show that the costs defined this way follow the paradigm introduced in Chapter 5, where we propose to modulate by the factor $1 - 2|e_{ij}|$ the costs of an additive scheme, which in this case is J-UNIWARD computed using the precover. Recalling that $X_{ij} = U_{ij} - e_{ij}$ and $Y_{ij} = U_{ij} + \text{sign}(e_{ij}) - e_{ij}$, we use the Dirac delta δ_{ij} to express

$$J^{-1}(\mathbf{U}_{\sim ij} X_{ij}) = J^{-1}(\mathbf{U} - \delta_{ij} e_{ij}) = \mathbf{P} + e_{ij} J^{-1}(\delta_{ij}), \quad (\text{A.0.5})$$

$$J^{-1}(\mathbf{U}_{\sim ij} Y_{ij}) = \mathbf{P} + (\text{sign}(e_{ij}) - e_{ij}) J^{-1}(\delta_{ij}). \quad (\text{A.0.6})$$

The linearity of convolution allows us to write (A.0.4)

$$\rho_{ij}^{(\text{SI})}(\mathbf{X}) = \sum_{b,u,v} \frac{|\text{sign}(e_{ij}) - e_{ij}| |(\mathbf{W}^{(b)} \star J^{-1}(\delta_{ij}))_{uv}|}{\sigma + |(\mathbf{W}^{(b)} \star \mathbf{P})_{uv}|} - \frac{|e_{ij}| |(\mathbf{W}^{(b)} \star J^{-1}(\delta_{ij}))_{uv}|}{\sigma + |(\mathbf{W}^{(b)} \star \mathbf{P})_{uv}|} \quad (\text{A.0.7})$$

$$= (1 - 2|e_{ij}|) \sum_{b,u,v} \frac{|(\mathbf{W}^{(b)} \star J^{-1}(\delta_{ij}))_{uv}|}{\sigma + |(\mathbf{W}^{(b)} \star \mathbf{P})_{uv}|} \quad (\text{A.0.8})$$

$$= (1 - 2|e_{ij}|) \rho_{ij}^{(\text{J})}(\mathbf{P}), \quad (\text{A.0.9})$$

which is the cost of changing the ij th DCT coefficient in J-UNIWARD with the precover (rather than cover) in the denominator modulated by $1 - 2|e_{ij}|$. When the denominator uses the cover \mathbf{X} instead of the precover, we will denote the J-UNIWARD costs with $\rho_{ij}^{(\text{J})}(\mathbf{X})$.

Appendix B

Cost modulation as a function of quality factor

In this appendix, we provide some insight into why the experimentally-found optimal modulation factor follows the ramp function (7.6.1) depicted in Figure 7.2.2. First, in Figure 7.6.7 we redraw the modulation factor shown in Figure 7.2.2 right as a function of the average quantization step $\bar{q} = 1/15 \sum_{i+j \leq 5} Q_{ij}$ instead of the quality factor Q . We only average the first five diagonals of the quantization matrix because this is where the vast majority of differences between two JPEG files occur ($x_{ij}^{(1)} \neq x_{ij}^{(2)}$). *This figure tells us that the modulation factor should be smaller for larger quantization steps and vice versa.* This important observation is validated via the following experiment.

A total of 100 random images from BOSSbase 1.01 were selected. A generalized Gaussian distribution (7.1.1) was fitted using the method of moments [70] to each AC DCT mode (i, j) across all 100 images, obtaining thus 63 values of the shape and width parameters a_{ij}, b , $1 \leq i, j \leq 8, i + j > 2$. For each AC DCT mode (i, j) and for each quantization step q , we twice generated $N_{MC} = 10^8$ independent realizations from $\mathcal{G}(0, a_{ij}, b_{ij})$, denoting them $t_k^{(1)}$ and $t_k^{(2)}$, $k \in \{1, \dots, N_{MC}\}$, and N_{MC} independent realizations $\xi_k^{(1)}$ and $\xi_k^{(2)}$ from $\mathcal{N}(0, 1)$, the acquisition noise. The non-rounded DCT coefficients and their rounded values were computed and denoted $c_k^{(l)} = (t_k^{(l)} + \xi_k^{(l)})/q$ and $x_k^{(l)} = \lfloor c_k^{(l)} \rfloor$, $l = 1, 2$. Next, we simulated J2-UNIWARD with $x_k^{(1)}$ as the cover and $x_k^{(2)}$ as the side-information with $\rho_{ij}^{(j)} = 1$ for all i, j modulated as in (6.2.14). The embedding was simulated with change probabilities as explained in Section 7.2 for a fixed relative payload $R = 0.4$ measured w.r.t. the number of non-zero coefficients, $N_0 = |\{k | x_k^{(1)} \neq 0\}|$, giving us the stego object $y_k \in \{x_k^{(1)} - 1, x_k^{(1)}, x_k^{(1)} + 1\}$. The impact of embedding on the cover model was measured by computing the complement of the Bhattacharyya coefficient¹ between the sample cover and stego distributions, $\mathbf{p}^{(x)}, \mathbf{p}^{(y)}$:

$$B(\mathbf{p}^{(x)}, \mathbf{p}^{(y)}) = 1 - \sum_r \sqrt{p_r^{(x)} p_r^{(y)}} \quad \text{where} \quad (\text{B.0.1})$$

$$p_r^{(x)} = \frac{1}{N_{MC}} \sum_{k=1}^{N_{MC}} [x_k^{(1)} = r], r \in \mathbb{Z} \quad (\text{B.0.2})$$

$$p_r^{(y)} = \frac{1}{N_{MC}} \sum_{k=1}^{N_{MC}} [y_k = r], r \in \mathbb{Z}. \quad (\text{B.0.3})$$

¹Since the Bhattacharyya distance is $B_{\text{dist}} = -\log(1 - B)$, B reaches its minimum exactly when B_{dist} does.

Above, $[P]$ denotes the Iverson bracket, $[P] = 1$ when P is true and 0 when P is false. The exact range of index r depends on the specific realizations generated. The Bhattacharyya coefficient was selected for its good numerical stability w.r.t. unpopulated bins.

Since the quantized cover and stego DCT coefficients $x_k^{(1)}$ and y_k depend on the DCT mode (i, j) , the quantization step q , and relative payload R , the sample distributions $\mathbf{p}^{(x)}$, $\mathbf{p}^{(y)}$ and thus $B(\mathbf{p}^{(x)}, \mathbf{p}^{(y)})$ also depend on these parameters. The optimal value of the modulation parameter, $m_{ij}(q, R)$, was determined for each DCT mode (i, j) by minimizing $B(\mathbf{p}^{(x)}, \mathbf{p}^{(y)})$ over $m \in [0, 1]$:

$$m_{ij}(q, R) = \arg \min_{m \in [0, 1]} B(\mathbf{p}^{(x)}, \mathbf{p}^{(y)}). \quad (\text{B.0.4})$$

The optimal values of the modulation parameter as a function of the quantization step q are shown in Figure 7.7.1 for low, mid, and high-frequency DCT modes for payload $R = 0.4$. The error bars are across the DCT modes from the frequency band. We observe that the modulation mainly depends on q and stays approximately constant over DCT modes for each frequency band. The dependence on the quantization step q is qualitatively and quantitatively similar to Figure 7.6.7, validating thus our design choice.

Bibliography

- [1] BOSSbase 1.01. <http://agents.fel.cvut.cz/stegodata/>.
- [2] P. Bas. Monobase. <http://patrickbas.ec-lille.fr/MonoBase/>, July 2016.
- [3] P. Bas. Steganography via cover-source switching. In *IEEE International Workshop on Information Forensics and Security*, Abu Dhabi, December 4–7 2016.
- [4] P. Bas. Steganography via Cover-Source Switching. IEEE Workshop on Information Forensics and Security (WIFS), 2016.
- [5] P. Bas. An embedding mechanism for Natural Steganography after down-sampling. IEEE ICASSP, 2017.
- [6] P. Bas. E1base. <http://patrickbas.ec-lille.fr/E1Base/>, January 2018.
- [7] P. Bas, T. Filler, and T. Pevný. Break our steganographic system – the ins and outs of organizing BOSS. In T. Filler, T. Pevný, A. Ker, and S. Craver, editors, *Information Hiding, 13th International Conference*, volume 6958 of Lecture Notes in Computer Science, pages 59–70, Prague, Czech Republic, May 18–20, 2011.
- [8] R. Böhme. Weighted stego-image steganalysis for JPEG covers. In K. Solanki, K. Sullivan, and U. Madhow, editors, *Information Hiding, 10th International Workshop*, volume 5284 of Lecture Notes in Computer Science, pages 178–194, Santa Barbara, CA, June 19–21, 2007. Springer-Verlag, New York.
- [9] R. Böhme. *Advanced Statistical Steganalysis*. Springer-Verlag, Berlin Heidelberg, 2010.
- [10] M. Boroumand, M. Chen, and J. Fridrich. Deep residual network for steganalysis of digital images. under review.
- [11] C. Cachin. An information-theoretic model for steganography. In D. Aucsmith, editor, *Information Hiding, 2nd International Workshop*, volume 1525 of Lecture Notes in Computer Science, pages 306–318, Portland, OR, April 14–17, 1998. Springer-Verlag, New York.
- [12] M. Carnein, P. Schöttler, and R. Böhme. Predictable rain? Steganalysis of public-key steganography using wet paper codes. In S. Katzenbeisser, R. Kwitt, and A. Piva, editors, *The 2nd ACM Workshop on Information Hiding and Multimedia Security*, pages 97–108, Salzburg, Austria, June 11–13, 2014.
- [13] R. Cogranne and F. Retraint. Application of hypothesis testing theory for optimal detection of LSB matching data hiding. *Signal Processing*, 93(7):1724–1737, July, 2013.
- [14] R. Cogranne, V. Sedighi, T. Pevný, and J. Fridrich. Is ensemble classifier needed for steganalysis in high-dimensional feature spaces? In *IEEE International Workshop on Information Forensics and Security*, Rome, Italy, November 16–19 2015.

- [15] T. Denemark, M. Boroumand, and J. Fridrich. Steganalysis features for content-adaptive JPEG steganography. *IEEE Transactions on Information Forensics and Security*, 11:1736–1746, April 20 2016.
- [16] T. Denemark and J. Fridrich. Detection of content-adaptive LSB matching (game theory approach). In A. Alattar, N. D. Memon, and C. Heitzenrater, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2014*, volume 9028, pages 04 1–12, San Francisco, CA, February 3–5, 2014.
- [17] T. Denemark and J. Fridrich. Improving steganographic security by synchronizing the selection channel. In A. Alattar, J. Fridrich, N. Smith, and P. Comesana Alfaro, editors, *The 3rd ACM Workshop on Information Hiding and Multimedia Security*, Portland, OR, June 17–19, 2015.
- [18] T. Denemark and J. Fridrich. Side-informed steganography with additive distortion. In *IEEE International Workshop on Information Forensics and Security*, Rome, Italy, November 16–19 2015.
- [19] T. Denemark and J. Fridrich. Steganography with multiple JPEG images of the same scene. *IEEE Transactions on Information Forensics and Security*, 12(10):2308–2319, October 2017.
- [20] T. Denemark and J. Fridrich. Steganography with two JPEGs of the same scene. In *IEEE ICASSP*, New Orleans, March 5–9 2017.
- [21] T. Denemark, V. Sedighi, V. Holub, R. Cogramne, and J. Fridrich. Selection-channel-aware rich model for steganalysis of digital images. In *IEEE International Workshop on Information Forensics and Security*, Atlanta, GA, December 3–5, 2014.
- [22] Tomáš Denemark and Jessica Fridrich. Model based steganography with precover. *Electronic Imaging*, pages 56–66, 2017.
- [23] S. Dumitrescu and X. Wu. LSB steganalysis based on higher-order statistics. In A. M. Eskicioglu, J. Fridrich, and J. Dittmann, editors, *Proceedings of the 7th ACM Multimedia & Security Workshop*, pages 25–32, New York, NY, August 1–2, 2005.
- [24] S. Dumitrescu, X. Wu, and N. D. Memon. On steganalysis of random LSB embedding in continuous-tone images. In *Proceedings IEEE, International Conference on Image Processing, ICIP 2002*, pages 324–339, Rochester, NY, September 22–25, 2002.
- [25] S. Dumitrescu, X. Wu, and Z. Wang. Detection of LSB steganography via Sample Pairs Analysis. In F. A. P. Petitcolas, editor, *Information Hiding, 5th International Workshop*, volume 2578 of Lecture Notes in Computer Science, pages 355–372, Noordwijkerhout, The Netherlands, October 7–9, 2002. Springer-Verlag, New York.
- [26] L. Fillatre. Adaptive steganalysis of least significant bit replacement in grayscale images. *IEEE Transactions on Signal Processing*, 60(2):556–569, 2011.
- [27] T. Filler and J. Fridrich. Gibbs construction in steganography. *IEEE Transactions on Information Forensics and Security*, 5(4):705–720, 2010.
- [28] T. Filler, J. Judas, and J. Fridrich. Minimizing additive distortion in steganography using syndrome-trellis codes. *IEEE Transactions on Information Forensics and Security*, 6(3):920–935, September 2011.
- [29] T. Filler, T. Pevný, and P. Bas. BOSS (Break Our Steganography System). <http://www.agents.cz/boss>, July 2010.
- [30] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian. Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing*, 17(10):1737–1754, Oct. 2008.

- [31] E. Franz. Steganography preserving statistical properties. In F. A. P. Petitcolas, editor, *Information Hiding, 5th International Workshop*, volume 2578 of Lecture Notes in Computer Science, pages 278–294, Noordwijkerhout, The Netherlands, October 7–9, 2002. Springer-Verlag, New York.
- [32] E. Franz. Embedding considering dependencies between pixels. In E. J. Delp, P. W. Wong, J. Dittmann, and N. D. Memon, editors, *Proceedings SPIE, Electronic Imaging, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, volume 6819, pages D 1–12, San Jose, CA, January 27–31, 2008.
- [33] E. Franz and A. Schneidewind. Pre-processing for adding noise steganography. In M. Barni, J. Herrera, S. Katzenbeisser, and F. Pérez-González, editors, *Information Hiding, 7th International Workshop*, volume 3727 of Lecture Notes in Computer Science, pages 189–203, Barcelona, Spain, June 6–8, 2005. Springer-Verlag, Berlin.
- [34] J. Fridrich. *Steganography in Digital Media: Principles, Algorithms, and Applications*. Cambridge University Press, 2009.
- [35] J. Fridrich. On the role of side-information in steganography in empirical covers. In A. Alattar, N. D. Memon, and C. Heitzentrater, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2013*, volume 8665, pages 0I 1–11, San Francisco, CA, February 5–7, 2013.
- [36] J. Fridrich and R. Du. Secure steganographic methods for palette images. In A. Pfitzmann, editor, *Information Hiding, 3rd International Workshop*, volume 1768 of Lecture Notes in Computer Science, pages 47–60, Dresden, Germany, September 29–October 1, 1999. Springer-Verlag, New York.
- [37] J. Fridrich, M. Goljan, and D. Soukal. Perturbed quantization steganography using wet paper codes. In J. Dittmann and J. Fridrich, editors, *Proceedings of the 6th ACM Multimedia & Security Workshop*, pages 4–15, Magdeburg, Germany, September 20–21, 2004.
- [38] J. Fridrich, M. Goljan, D. Soukal, and P. Lisoněk. Writing on wet paper. In E. J. Delp and P. W. Wong, editors, *Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents VII*, volume 5681, pages 328–340, San Jose, CA, January 16–20, 2005.
- [39] J. Fridrich and J. Kodovský. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3):868–882, June 2011.
- [40] J. Fridrich and J. Kodovský. Multivariate Gaussian model for designing additive distortion for steganography. In *Proc. IEEE ICASSP*, Vancouver, BC, May 26–31, 2013.
- [41] Q. Giboulot, R. Cogranne, and P. Bas. Steganalysis into the wild: How to define a source? In A. Alattar and N. D. Memon, editors, *Proceedings IS&T, Electronic Imaging, Media Watermarking, Security, and Forensics 2016*, San Francisco, CA, January 29–February 2, 2018.
- [42] L. Guo, J. Ni, and Y.-Q. Shi. An efficient JPEG steganographic scheme using uniform embedding. In *Fourth IEEE International Workshop on Information Forensics and Security*, Tenerife, Spain, December 2–5, 2012.
- [43] L. Guo, J. Ni, and Y. Q. Shi. Uniform embedding for efficient JPEG steganography. *IEEE Transactions on Information Forensics and Security*, 9(5):814–825, May 2014.
- [44] S. Hetzl and P. Mutzel. A graph-theoretic approach to steganography. In J. Dittmann, S. Katzenbeisser, and A. Uhl, editors, *Communications and Multimedia Security, 9th IFIP TC-6 TC-11 International Conference, CMS 2005*, volume 3677 of Lecture Notes in Computer Science, pages 119–128, Salzburg, Austria, September 19–21, 2005.

- [45] V. Holub and J. Fridrich. Designing steganographic distortion using directional filters. In *Fourth IEEE International Workshop on Information Forensics and Security*, Tenerife, Spain, December 2–5, 2012.
- [46] V. Holub and J. Fridrich. Random projections of residuals as an alternative to co-occurrences in steganalysis. In A. Alattar, N. D. Memon, and C. Heitznerater, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2013*, volume 8665, pages OL 1–11, San Francisco, CA, February 5–7, 2013.
- [47] V. Holub and J. Fridrich. Random projections of residuals for digital image steganalysis. *IEEE Transactions on Information Forensics and Security*, 8(12):1996–2006, December 2013.
- [48] V. Holub and J. Fridrich. Challenging the doctrines of JPEG steganography. In A. Alattar, N. D. Memon, and C. Heitznerater, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2014*, volume 9028, pages 02 1–8, San Francisco, CA, February 3–5, 2014.
- [49] V. Holub and J. Fridrich. Low-complexity features for JPEG steganalysis using undecimated DCT. *IEEE Transactions on Information Forensics and Security*, 10(2):219–228, February 2015.
- [50] V. Holub and J. Fridrich. Phase-aware projection model for steganalysis of JPEG images. In A. Alattar and N. D. Memon, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2015*, volume 9409, pages 0T 1–11, San Francisco, CA, February 8–12, 2015.
- [51] V. Holub, J. Fridrich, and T. Denemark. Universal distortion function for steganography in an arbitrary domain. *EURASIP Journal on Information Security (Revised Selected Papers of ACM IH and MMS 2013)*, 2013.
- [52] F. Huang, J. Huang, and Y.-Q. Shi. New channel selection rule for JPEG steganography. *IEEE Transactions on Information Forensics and Security*, 7(4):1181–1191, August 2012.
- [53] J. R. Janesick. *Scientific Charge-Coupled Devices*, volume Monograph PM83. Washington, DC: SPIE Press - The International Society for Optical Engineering, January 2001.
- [54] A. D. Ker. Improved detection of LSB steganography in grayscale images. In J. Fridrich, editor, *Information Hiding, 6th International Workshop*, volume 3200 of Lecture Notes in Computer Science, pages 97–115, Toronto, Canada, May 23–25, 2004. Springer-Verlag, Berlin.
- [55] A. D. Ker. A general framework for structural analysis of LSB replacement. In M. Barni, J. Herrera, S. Katzenbeisser, and F. Pérez-González, editors, *Information Hiding, 7th International Workshop*, volume 3727 of Lecture Notes in Computer Science, pages 296–311, Barcelona, Spain, June 6–8, 2005. Springer-Verlag, Berlin.
- [56] A. D. Ker. Fourth-order structural steganalysis and analysis of cover assumptions. In E. J. Delp and P. W. Wong, editors, *Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents VIII*, volume 6072, pages 25–38, San Jose, CA, January 16–19, 2006.
- [57] A. D. Ker. A fusion of maximal likelihood and structural steganalysis. In T. Furon, F. Cayre, G. Doërr, and P. Bas, editors, *Information Hiding, 9th International Workshop*, volume 4567 of Lecture Notes in Computer Science, pages 204–219, Saint Malo, France, June 11–13, 2007. Springer-Verlag, Berlin.
- [58] A. D. Ker. Optimally weighted least-squares steganalysis. In E. J. Delp and P. W. Wong, editors, *Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, volume 6505, pages 6 1–6 16, San Jose, CA, January 29–February 1, 2007.

-
- [59] A. D. Ker. Steganalysis of embedding in two least significant bits. *IEEE Transactions on Information Forensics and Security*, 2:46–54, 2007.
- [60] A. D. Ker, T. Pevný, and P. Bas. Rethinking optimal embedding. In F. Perez-Gonzales, F. Cayre, and P. Bas, editors, *4th ACM IH&MMSec. Workshop*, Vigo, Spain, June 20–22, 2016.
- [61] Y. Kim, Z. Duric, and D. Richards. Modified matrix encoding technique for minimal distortion steganography. In J. L. Camenisch, C. S. Collberg, N. F. Johnson, and P. Sallee, editors, *Information Hiding, 8th International Workshop*, volume 4437 of Lecture Notes in Computer Science, pages 314–327, Alexandria, VA, July 10–12, 2006. Springer-Verlag, New York.
- [62] J. Kodovský and J. Fridrich. Steganalysis in high dimensions: Fusing classifiers built on random subspaces. In A. Alattar, N. D. Memon, E. J. Delp, and J. Dittmann, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security and Forensics III*, volume 7880, pages OL 1–13, San Francisco, CA, January 23–26, 2011.
- [63] J. Kodovský and J. Fridrich. Steganalysis of JPEG images using rich models. In A. Alattar, N. D. Memon, and E. J. Delp, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2012*, volume 8303, pages 0A 1–13, San Francisco, CA, January 23–26, 2012.
- [64] J. Kodovský, J. Fridrich, and V. Holub. Ensemble classifiers for steganalysis of digital media. *IEEE Transactions on Information Forensics and Security*, 7(2):432–444, 2012.
- [65] M. Kwan. Gifshuffle. <http://www.darkside.com.au/gifshuffle>.
- [66] B. Li, M. Wang, and J. Huang. A new cost function for spatial image steganography. In *Proceedings IEEE, International Conference on Image Processing, ICIP*, Paris, France, October 27–30, 2014.
- [67] B. Li, M. Wang, X. Li, S. Tan, and J. Huang. A strategy of clustering modification directions in spatial image steganography. *IEEE Transactions on Information Forensics and Security*, 10(9):1905–1917, September 2015.
- [68] P. Lu, X. Luo, Q. Tang, and L. Shen. An improved sample pairs method for detection of LSB embedding. In J. Fridrich, editor, *Information Hiding, 6th International Workshop*, volume 3200 of Lecture Notes in Computer Science, pages 116–127, Toronto, Canada, May 23–25, 2004. Springer-Verlag, Berlin.
- [69] 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, June 2010.
- [70] S. Meignen and H. Meignen. On the modeling of DCT and subband image data for compression. *IEEE Transactions on Image Processing*, 4(2):186–193, February 1995.
- [71] S. Nadarajah and S. Kotz. Exact distribution of the max-min of two Gaussian random variables. *IEEE Transactions on VLSI Systems*, 16(2):210–212, February 2008.
- [72] Rawkit package. <https://rawkit.readthedocs.io/en/latest/>. Python package.
- [73] K. Petrowski, M. Kharrazi, H. T. Sencar, and N. D. Memon. Psteg: Steganographic embedding through patching. In *Proceedings IEEE, International Conference on Acoustics, Speech, and Signal Processing*, pages 537–540, Philadelphia, PA, March 18–23, 2005.
- [74] T. Pevný, P. Bas, and J. Fridrich. Steganalysis by subtractive pixel adjacency matrix. In J. Dittmann, S. Craver, and J. Fridrich, editors, *Proceedings of the 11th ACM Multimedia & Security Workshop*, pages 75–84, Princeton, NJ, September 7–8, 2009.

- [75] T. Pevný, T. Filler, and P. Bas. Using high-dimensional image models to perform highly undetectable steganography. In R. Böhme and R. Safavi-Naini, editors, *Information Hiding, 12th International Conference*, volume 6387 of Lecture Notes in Computer Science, pages 161–177, Calgary, Canada, June 28–30, 2010. Springer-Verlag, New York.
- [76] N. Provos. Defending against statistical steganalysis. In *10th USENIX Security Symposium*, pages 323–335, Washington, DC, August 13–17, 2001.
- [77] V. Sachnev, H. J. Kim, and R. Zhang. Less detectable JPEG steganography method based on heuristic optimization and BCH syndrome coding. In J. Dittmann, S. Craver, and J. Fridrich, editors, *Proceedings of the 11th ACM Multimedia & Security Workshop*, pages 131–140, Princeton, NJ, September 7–8, 2009.
- [78] P. Sallee. Model-based methods for steganography and steganalysis. *International Journal of Image Graphics*, 5(1):167–190, 2005.
- [79] A. Sarkar, K. Solanki, and B. S. Manjunath. Further study on YASS: Steganography based on randomized embedding to resist blind steganalysis. In E. J. Delp, P. W. Wong, J. Dittmann, and N. D. Memon, editors, *Proceedings SPIE, Electronic Imaging, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, volume 6819, pages 16–31, San Jose, CA, January 27–31, 2008.
- [80] P. Schöttle and R. Böhme. A game-theoretic approach to content-adaptive steganography. In M. Kirchner and D. Ghosal, editors, *Information Hiding, 14th International Conference*, volume 7692 of Lecture Notes in Computer Science, pages 125–141, Berkeley, California, May 15–18, 2012.
- [81] P. Schöttle, S. Korff, and R. Böhme. Weighted stego-image steganalysis for naive content-adaptive embedding. In *Fourth IEEE International Workshop on Information Forensics and Security*, Tenerife, Spain, December 2–5, 2012.
- [82] V. Sedighi, R. Cogranne, and J. Fridrich. Content-adaptive steganography by minimizing statistical detectability. *IEEE Transactions on Information Forensics and Security*, 11(2):221–234, 2015.
- [83] V. Sedighi and J. Fridrich. Effect of saturated pixels on security of steganographic schemes for digital images. In *Image Processing (ICIP), 2016 IEEE International Conference on*, pages 2747–2751. IEEE, 2016.
- [84] V. Sedighi, J. Fridrich, and R. Cogranne. Content-adaptive pentary steganography using the multivariate generalized Gaussian cover model. In A. Alattar and N. D. Memon, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2015*, volume 9409, San Francisco, CA, February 8–12, 2015.
- [85] X. Song, F. Liu, C. Yang, X. Luo, and Y. Zhang. Steganalysis of adaptive JPEG steganography using 2D Gabor filters. In A. Alattar, J. Fridrich, N. Smith, and P. Comesana Alfaro, editors, *The 3rd ACM Workshop on Information Hiding and Multimedia Security, IH&MMSec '15*, Portland, OR, June 17–19, 2015.
- [86] J. Fridrich T. Denemark and V. Holub. Further study on security of s-uniward. In A. Alattar, N. D. Memon, and C. Heitzenrater, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2014*, volume 9028, San Francisco, CA, February 3–5, 2014.
- [87] W. Tang, H. Li, W. Luo, and J. Huang. Adaptive steganalysis against WOW embedding algorithm. In S. Katzenbeisser, R. Kwitt, and A. Piva, editors, *The 2nd ACM Workshop on Information Hiding and Multimedia Security*, pages 91–96, Salzburg, Austria, June 11–13, 2014.

- [88] W. Tang, S. Tan, B. Li, and J. Huang. Automatic steganographic distortion learning using a generative adversarial network. *IEEE Signal Processing Letters*, 24(10):1547–1551, 2017.
- [89] T. H. Thai, R. Cogranne, and F. Retraint. Camera model identification based on the heteroscedastic noise model. *Image Processing, IEEE Transactions on*, 23(1):250–263, Jan 2014.
- [90] D. Upham. Steganographic algorithm jsteg. Software available at <http://zoooid.org/~paul/crypto/jsteg>.
- [91] C. Wang and J. Ni. An efficient JPEG steganographic scheme based on the block-entropy of DCT coefficients. In *Proc. of IEEE ICASSP*, Kyoto, Japan, March 25–30, 2012.
- [92] A. Westfeld. High capacity despite better steganalysis (F5 – a steganographic algorithm). In I. S. Moskowitz, editor, *Information Hiding, 4th International Workshop*, volume 2137 of Lecture Notes in Computer Science, pages 289–302, Pittsburgh, PA, April 25–27, 2001. Springer-Verlag, New York.
- [93] J. Ye, J. Ni, and Y. Yi. Deep learning hierarchical representations for image steganalysis. *IEEE Transactions on Information Forensics and Security*, pages 2545–2557, 2017.
- [94] C. Zitzmann, R. Cogranne, F. Retraint, I. Nikiforov, L. Fillatre, and P. Cornu. Statistical decision methods in hidden information detection. In T. Filler, T. Pevný, A. Ker, and S. Craver, editors, *Information Hiding, 13th International Conference*, Lecture Notes in Computer Science, pages 163–177, Prague, Czech Republic, May 18–20, 2011.