Binary Codes for Enhancing the Most Significant Bit-Based Selective Encryption

Binary Codes for Enhancing the Most Significant Bit-Based Selective Encryption

By Mehrshad Kafi, M.Sc.,

A Thesis Submitted to the School of Graduate Studies in the Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy

McMaster University

Doctor of Philosophy (2022)

Hamilton, Ontario (Department of Electrical and Computer Engineering)

TITLE: Binary Codes for Enhancing the Most Significant Bit-Based Selective Encryption

AUTHOR: Mehrshad Kafi, M.Sc. (McMaster University)

SUPERVISOR: Dr. Sorina Dumitrescu

NUMBER OF PAGES: xvii, 158

To my wife, Sahar, who is always a source of support and patience for me

and to him

Abstract

Selective encryption (SE) methods for images often encrypt the sign bits, i.e., the most significant bits (MSBs) of the codewords corresponding to key syntax elements (in compressed images) or to pixel intensities (for uncompressed images). Our work is motivated by the key observation that the binary code (BC) used for these representations has an impact on the quality of the reconstruction at the eavesdropper (Eve), which can be regarded as a measure of the degree of security of the encryption (the lower the quality, the higher the level of security). Therefore, we address the design of BCs that enhance the security of MSB-based SE by increasing the degradation at Eve's side when she uses a simple replacement attack (replacing all MSBs either by 0s or by 1s).

We first consider the scenario of fixed-length BCs, i.e., where all the codewords have the same length. We formulate the design problem as an optimization problem with the objective of maximizing the distortion at Eve's without any constraint or with a constraint on the entropy of the MSBs in order to shorten the size of the MSB stream to reduce the computational overhead of encryption. We show that the problem can be cast as a binary integer linear program equivalent to a weighted non-bipartite graph matching problem, for which polynomial-time solution algorithms exist. We empirically assess the performance of the optimized BCs on a Mixed Gaussian source, as well as on Gaussian and Laplacian sources, the latter two being commonly used to model the distribution of transform coefficients and prediction residuals. Our experiments lead to the conclusion that MSB-based SE schemes could benefit from the proposed BC designs.

For the case of uncompressed images, we also propose a family of structured BCs for the pixel intensity values. These BCs are constructed such that intensity values that are close have reconstructions that are far apart. As a result, the reconstruction with the replacement attack significantly destroys the smooth areas and blurs the edges, therefore increasing the degree of security.

Next, we focus on the construction of variable-length BCs for the non-zero quantized AC coefficients in JPEG compressed images. For this, we first formulate the BC design problem as the problem of maximizing the distortion at Eve's side with a constraint on the entropy. This problem can also be cast as a weighted non-bipartite graph matching problem and, therefore, can be solved efficiently. Furthermore, by gaining insights from the optimization results, a simpler and faster method for BC design is devised, which consists of only swapping a few codewords in the original code used in JPEG. We assess the practical performance of the proposed BCs for the SE method of JPEG images that encrypts only the MSBs of the non-zero quantized AC coefficients, along with the full encryption of the DC coefficients. Our experimental results show that high visual security can be achieved with only a small sacrifice in compression efficiency. In addition, the proposed BCs can be tuned to achieve various levels of degradation at Eve's side, a property that is convenient for certain applications.

Acknowledgements

I would like to extend my extreme gratitude and respect to Professor Sorina Dumitrescu for her continuous support, immense knowledge, and especially her patience throughout this process. Without her invaluable guidance and dedication, none of the thesis work would have been materialized. Great appreciations are also expressed to my committee members Dr. Shahram Shirani and Dr. Jun Chen, for their valuable guidance and comments on my research works. Moreover, I appreciate the help and moral support from Ms. Cheryl Gies the graduate administrative assistant of Department of Electrical and Computer Engineering.

I cannot forget to thank my mother and father for all their unconditional support in these years. Last but not least, I would like to thank my wife, Sahar Motamedi, for her unwavering support and patience throughout my PhD journey.

Contents

List of Figures

List of Tables

Abbreviations

1BPE	1 Bit Plane Encryption
2BPE	2 Bit Plane Encryption
ACC	AC Coefficient
ANBC	Alternate NBC
BC	Binary Code
CtE	Compression then Encryption
DCC	DC Coefficient
DCCA	DC Category Attack
DCorr	Diagonal Correlation
DCT	Discrete Cosine Transform
DPCM	Differential Pulse Code Modulation
EAC	Energy of AC coefficients attack
ECOptBC	Entropy-constrained Optimized BC for QACs
EffCA	Efficient Codeword Assignment
EOB	End of Block
EONB	Entropy-optimized NBC
EtC	Encryption then Compression
Eve	Eavesdropper
FastCA	Faster Codeword Assignment
FBC	Folded Binary Code
FCE	Format-compliant Encryption
FE	Full Encryption
HCorr	Horizontal Correlation
i.i.d.	Identically and Independently Distributed

IDCT	Inverse Discrete Cosine Transform
INCC	Improved Non-zero Coefficient Count attack
LP	Linear Program
LSB	Least Significant Bit
MSB	Most Significant Bit
MSB-SE	SE methods with MSB encryption
MSE	Mean Squared Error
MVD	Motion Vector Differences
NBC	Natural Binary Code
NZC	Non-zero Coefficients Count
OPT	Optimized Binary Code
OPTA	Optimized Binary Code in scenario A
OPTB	Optimized Binary Code in scenario B
OptBC	Optimized BC for QACs without Entropy Constraint
OrigBC	Original JPEG BC
pdf	Probability Density Function
PSNR	Peak Signal-to-Noise Ratio
QAC	Non-Zero Quantized AC Coefficient
QF	Quality Factor
RNBC	Reversed Natural Binary Code
RS	Run of Zeros, Size
RV	Random Variable
SBC	Spaced BCs
SE	Selective Encryption
SI	Side Information

SSIM	Structural Similarity Index
SwapBC	Swap-based BC
VCorr	Vertical Correlation
VLC	Variable-length Code
VLI	Variable-length Integer

Chapter 1

Introduction

Nowadays, high-quality images and videos can be captured simply by typical smartphones, and are transmitted over the Internet and wireless networks. The large-size multimedia data can be in high quality without any lossy compression or be compressed due to the limited network bandwidth and high storage costs. In addition, data security is crucial and ensuring privacy against eavesdropping is becoming increasingly a major concern, especially when transmitting sensitive information. Therefore, uncompressed or compressed multimedia data should be encrypted efficiently. First, this chapter explains the limitations of traditional encryption methods called Full Encryption. Next, it introduces the Selective Encryption methods developed to overcome the challenges of Full Encryption. A brief review of these methods developed for various compression standards and uncompressed images is given. Section 1.3 describes our contributions and the thesis organization.

1.1 Full Encryption

One method that guarantees security is first to compress the data and afterwards apply a standard encryption algorithm to the obtained bitstream. This method is referred to as Full Encryption (FE). However, FE increases the latency and the computational costs at both the transmitter and receiver. Thus, FE is less appealing in real-time or fast communication applications such as video conferencing, TV broadcasting, and telemedicine of electronic health records (e.g. MRI and X-ray Images) [14, 39, 17, 26, 1] and/or in applications involving resource-constrained devices like smartphones or wireless sensors. Moreover, FE encrypts the format-related bits too, and it not only could cause the secret key information leakage due to the known structures of these bits [61], but it could also make impossible any processing of the ciphered data that accepts only specific format files, such as browsing, cutting, or storing [20, 42]. Thus, the adaptation and transmission error recovery will become harder or even impossible [66]. On the other hand, data can be encrypted completely before compression, and then a standard compression is applied to the encrypted data. This approach is not efficient too, because the encryption can affect the characteristics of multimedia data which are often used for adequate compression, such as high correlations between the pixels and frames or the proper order of quantized transformed coefficients.

1.2 Selective Encryption

Selective encryption (SE) (also referred to as partial or soft encryption [6, 44, 21]) methods have been developed to avoid the aforementioned drawbacks of FE. Such methods encrypt only content-related bits, such as the bitstream of the quantized

transform coefficients in transform-based compression algorithms, and further select only some parts of these bitstreams for encryption. Generally, SE methods divide the data into important and less important sections, such that if only the former part is ciphered, eavesdropper cannot achieve useful information, even if the latter part remains unencrypted.

Selective encryption significantly reduces the amount of data to be encrypted, hence, reduce the computational cost of encryption in comparison to FE [61, 64]. Additionally, they can provide more flexibility in terms of security since not all applications require the same degree of secrecy. For instance, in military applications or telemedicine the highest confidentiality (termed "confidential security") is needed, while in entertainment applications, it is sufficient to guarantee only "perceptual security", i.e., to make sure that any illegitimate user can only recover a degraded version which makes the content non-consumable [21, 67, 19, 78]. Actually, the latter feature, i.e., being able to discern some low quality information about the content from the encrypted data, is even desirable in certain cases, for instance when advertising to potential clients[21].

Furthermore, usually, the data is ciphered by a secret key and the secret key itself has to be ciphered by a more complex public key, for being protected during transmission to the authorized user. As the public key is more complex, encryption of the whole data using the public key would be too time-consuming. However, in SE, the data subjected to encryption is so small that the public key can be applied to this part and thus the need for a secret key is eliminated totally [6]. Therefore, again the time and complexity requirements of the system are reduced while still providing security. Finally, SE can preserve the functionality of the bitstream too,

by keeping the file format compliant [21].

As a result, SE can meet the requirements of applications where both time and security are essential, such as pay TV, subscription limited social networks, process and transmission of medical images, Internet banking transactions, confidential video conferencing, corporate communications, online video gaming and so on. Moreover, with SE, the power consumption needed in the encryption process is drastically decreased in comparison with FE and hence it is convenient for power constrained systems like mobile devices.

SE methods have been devised for various image and video compression standards [76, 44, 13]. For instance, SE methods [55, 32, 49, 69, 35, 33, 80, 52, 47, 8, 82, 7, 18, 37, 25, 62, 36, 45, 46, 63, 48, 3, 15, 54, 65, 64, 39, 27, 68, 4, 77, 34, 53, 22] have been developed for the JPEG standard, which is accredited as the most popular image lossy compression standard [20, 11]. SE method for SPIHT images was designed by encrypting the significance information related to pixels or sets in the two highest pyramid levels [6]. For JPEG 2000, one approach for SE uses randomized arithmetic coding [16]. Another approach for the SE of JPEG 2000 images is to encrypt the most significant layer of low resolutions bands for hard visual degradation [43]. Video data has been subjected to SE too, due to its size, real-time applications, the necessity of format-compliance, and multi-user different quality demands. A widespread strategy is the encryption of the sign bits of key syntax elements such as the quantized transform coefficients and/or the motion vector differences (MVD). For instance, the MSBs of quantized transform coefficients in JPEG images are considered for encryption [45, 39]. This strategy was used for the SE of MPEG-compressed video [61, 75, 83, 34], for the H.264/AVC

standard [38, 81, 71, 72, 2] and for HEVC-compressed video [60, 5, 58]. More recently, the encryption of the prediction modes for intra blocks for H.264/AVC and intra units for HEVC was considered too [5].

Besides the SE of compressed images and videos, recent years have witnessed a growing interest in the development of SE techniques for uncompressed images [79, 28, 59, 41, 56]. In the aforementioned works, the encryption is applied only to the higher order bitplanes and the focus is on designing the encryption algorithms based on chaotic systems and DNA computing.

1.3 Contributions and Thesis Organization

This work is motivated by the observation that many SE approaches proposed to date for compressed images and videos and uncompressed images use the encryption of the sign bits of some syntax elements such as the quantized transform coefficients or motion vector differences, possibly in conjunction with the encryption of other portions of the bitstream. Since the sign bit can be interpreted as the Most Significant Bit (MSB) in the binary representation of the syntax element, encrypting the sign bits corresponds to encrypting the stream of MSBs. We will use the acronym MSB-SE to refer to SE methods that include the encryption of the MSB stream.

Our key insight is that the way of assigning binary codewords to these syntax elements must influence the performance of the SE method. Prior work on the SE of compressed images and videos based on sign bit encryption uses the assignment

of binary codewords to the syntax elements that is specified by the respective compression algorithm or standard.

In this work, we consider the possibility of going beyond the conventional binary codes and applying a different binary code (BC) to represent these syntax elements. Since the purpose of SE is to degrade the reconstruction at the eavesdropper's side, we are interested in designing judiciously a BC such that this degradation is maximized. Another goal of SE is to decrease the amount of the encrypted portion which is already achieved by selecting only a portion of the whole data stream for encryption. Furthermore, if the selected portion for encryption is compressed before being encrypted, the BC could also affect the length of this bitstream. Therefore, besides achieving the high security at the eavesdropper, the reduction of the length of the bitstream to be encrypted can be considered in the design of the new BCs leading to low encryption complexity.

Therefore, in Chapter 2, we address the problem of optimal fixed-length BC design for SE. Ideally, the goal is to simultaneously maximize the eavesdropper's distortion and minimize the length of the compressed MSB stream. Since these two objectives are conflicting in general, we formulate the problem as the maximization of a weighted sum of the eavesdropper's distortion and of the probability of the MSB being 0. We cast the problem as a binary integer linear program equivalent to a weighted non-bipartite graph matching problem, which has a polynomial-time solution algorithm. We show that, when the source to be quantized and the quantizer are symmetric, the problem can be converted to a linear program of smaller size, for a family of distortion metrics. Next, we proceed to assess the practical performance of the optimized BCs in comparison with the BCs commonly used,

namely the Natural Binary Code (NBC) and the Folded Binary Code (FBC). This comparison can also be regarded as assessing how effective the conventional codes are relative to the optimum in ensuring secrecy and lowering the encryption complexity. To this end, we perform experiments on three sources, namely a Gaussian source and a Laplacian source, which have been used to model the distribution of transform coefficients and of prediction residuals [31], as well as on a Gaussian mixture. These analyses and their results have been presented and published in [23] and [24] too.

Although the proposed design is for the scenario when the data stream is compressed before encryption, the concept can also be applied to the SE of uncompressed images. In Chapter 3, the optimized BC developed in Chapter 2 is applied to obtain a binary representation of each pixel value. We present experimental results of the new BCs and show that higher visual secrecy can be obtained in comparison with the conventional approach. Extensive experiments assess the performance of the optimized BC in comparison with existing approaches. The results reveal that certain existing selective encryption schemes could benefit from the proposed design. These results have also been published in [24]. Furthermore, in this chapter we develop structured BCs for uncompressed images that increase the security of SE of MSB encryption, but without the computational overhead of the optimization. Our construction exploits the intrinsic property of natural images of having similar intensities for pixels that are spatially close. The key idea of our design is to construct the BC in such a way that if the original pixel values are close to one another, when their MSBs are encrypted then they will be far away from each other inducing degradation into the image.

In Chapter 4, we are concerned with the design of variable-length BCs tailored for MSB-SE of images compressed using the JPEG standard. We demonstrate that applying MSB encryption to DCT coefficients of the JPEG-compressed images with their baseline coding does not ensure a sufficient degree of confidentiality, even when combined with the full encryption of the DC coefficients. Therefore, again we address this shortcoming by leveraging the idea that the binary codewords used to represent the syntax elements can influence the quality of the reconstruction at the eavesdropper. We focus on the SE method that encrypts only the MSBs of non-zero quantized AC coefficients, which we name QACs, along with the full encryption of the DC differential bitstream, and propose novel binary code representations for the QACs that achieve high security. Since our measure of security is the degree of degradation at eavesdropper's, our first technique is based on solving an optimization problem that aims at maximizing the distortion at eavesdropper's while keeping small the increase in bitrate of the compressed bitstream. Our results show that high security can be achieved with only a small sacrifice in compression efficiency in most cases. In addition, the trade-off between secrecy and bitrate can be controlled by varying a certain parameter, thus allowing for more flexibility in the choice of the security level according to the application. Further, by exploiting the insights gained from the analysis of the structure of the new BC, we propose a simpler and faster method for BC design by only swapping a few codewords in the original code assignment. The swap-based approach can achieve high levels of security at bitrates that are comparable or even smaller than for the optimization-based technique, and it has a smaller computational cost. These developed methods and their results has been composed in [22].

Chapter 2

Fixed-Length BC Optimized for SE

Prior work on the SE of uncompressed images and compressed images and videos based on sign bit encryption uses the assignment of conventional binary codewords to pixel values of uncompressed images or BC of the syntax elements that is specified by the respective compression algorithm or standard. In this chapter, we consider the possibility of applying a different BC to represent these syntax elements for more efficient SE. Since the purpose of SE is to degrade the reconstruction at the eavesdropper's side, we are interested in designing a binary code such that this degradation is maximized. Another goal of SE is to decrease the amount of the encrypted portion. This is already partly achieved by selecting only a portion of the whole data stream for encryption. However, in situations where the portion chosen for encryption is further compressed before being encrypted, the BC could also affect the length of this compressed bitstream. In such a situation it might be of interest to further decrease the length of the bitstream to

be encrypted (for low encryption complexity) while maintaining a high enough degradation at the eavesdropper (i.e., high secrecy).

This chapter is organized into six sections. The following section describes the general framework and contributuons of the chapter. Section 2.2 introduces the definitions, notations, and the problem set-up. Section 2.3 formulates the problem of optimizing the binary code for SE and shows that it is equivalent to a weighted non-bipartite graph matching program. Section 2.4 demonstrates that the problem can be equivalently cast as linear program in the case of a symmetric source with the symmetric quantizer. In Section 2.5, experimental results for i.i.d. sources are presented and discussed. Finally, Section 2.6 concludes the chapter.

2.1 General Framework

In order to investigate this problem, we consider a general framework where a signal is first applied a scalar quantizer. Each quantized level is converted into a binary codeword consisting of b bits, where b is a fixed integer. Let N be the number of signal samples. Thus, at the end of the process, a sequence of N binary codewords is obtained, which are concatenated to produce a codestream. We consider two scenarios, A and B. For simplicity, we will refer to the eavesdropper as Eve, while Alice is the sender, and Bob is the legitimate receiver, as is common practice in the literature on information security. In scenario A, Alice encrypts the most significant bits (MSB) of all codewords, while the remaining portions are not encrypted. Then Alice sends the resulted bitstream to Bob. To perform the encryption, the MSBs are concatenated and this stream is applied a strong enough cipher, for instance by utilizing a standard encryption algorithm such as

AES. We will assume that the cyphertext contains the same number of bits as the input sequence of MSBs. After that the MSB of each codeword is replaced by the corresponding bit in the cyphertext.

In scenario B, the MSB is separated from the rest of the codeword. The sequence of MSBs is further compressed using an entropy coder and the resulted bitstream is further encrypted. The sequence of remaining $(b-1)$-bit portions of the codewords is further compressed, but not encrypted. Alice appends the resulted codestream after the cyphertext and transmits it to Bob.

Note that scenario A is encountered in the SE of JPEG-compressed images where the sign bits of the non-zero quantized AC coefficients are encrypted. The non-zero values are divided into categories and the values within each category are represented by a fixed length binary code (i.e., where all codewords have the same number of bits). The MSB of each codeword represents the sign bit. Thus, for each category, the SE process falls under scenario A. Another case is the encryption of the sign bits of the MVDs in video streams compressed with H.264/AVC. In the baseline profile, the non-zero MVDs are divided into the same categories as for JPEG and the MVDs within each category are coded using a fixed-length binary code. By considering the sign bit as the MSB, this process also falls under scenario A. In addition, wavelet-based compression algorithms tend to use bitplane coding. A potential SE method is to encrypt the MSB plane after it is entropy-coded. Such an approach falls under scenario B. The encoding and encryption of scenarios A and B and the method of reconstructions at legitimate user and eavesdropper are presented in Figure 2.1.

We will assume that Eve is not able to break the encryption, but she is able to obtain a degraded reconstruction of the signal and we will measure the secrecy of the scheme by the distortion achieved at Eve's side. Thus, in scenario A, we are interested in designing a BC that maximizes the distortion at Eve's. In scenario B, we will use the entropy of the random variable S_0 that represents the MSB as a measure of the complexity of the encryption. This is accurate when the input signal samples are are i.i.d. (identically and independently distributed) and entropy coding is used to compress the MSB plane. We address the problem of designing the BC such that to maximize the secrecy, while keeping the encryption complexity as low as possible. These two objectives are generally conflicting requirements, thus we will resort to maximizing a utility function which accounts for both objectives, namely the weighted sum of the distortion at Eve's side and the probability of the MSB being 0.

We show that for both scenarios the optimization problem can be cast as a maximum weight matching problem, which is known to have a solution algorithm that runs in $O(M^3)$ time, where M is the total number of quantizer bins. We additionally prove that, when the source probability density function (pdf) and the quantizer are symmetric, the problem can be formulated as a linear program of a smaller size for a class of distortion measures that includes the squared error distortion. The latter formulation allows for more efficient and for a larger variety of solution algorithms to be used.

Next, we proceed to assess the practical performance of the optimized binary codes in comparison with the binary codes commonly used, namely the natural binary code (NBC) and the folded binary code (FBC). This comparison can also

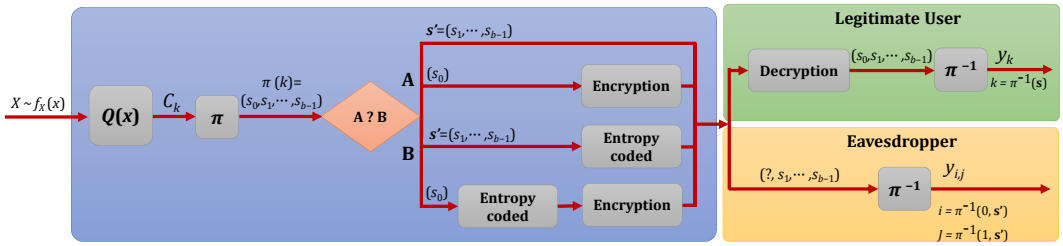


FIGURE 2.1: Block diagram of the encoding and encryption processes.

be regarded as assessing how effective the conventional codes are relative to the optimum in ensuring secrecy and lowering the encrythe encryption. To this end, we perform experiments on three sources, namely a Gaussian source, a Laplacian source, and a Gaussian mixture. The first two sources are chosen since they both have been used to model the distribution of transform coefficients and of prediction residuals [31]. Our results indicate that existing SE methods based on sign bit encryption could benefit from the optimization of the binary code.

2.2 Preliminaries

Let X denote the random variable (RV) representing the values to be quantized, with continuous pdf $f(x)$, $x \in \mathbf{R}$, mean μ and finite variance σ^2. The scalar quantizer Q is specified by the encoder partition thresholds $b_1 < b_2 < \cdots < b_{M-1}$, and by the reconstruction values $y_0, \cdots, y_{M-1}$, where $M = 2^b$, for some positive integer b. The quantizer cells are denoted $C_0, C_1, \cdots, C_{M-1}$, with $C_k = (b_k, b_{k+1}]$, for $0 \leq k \leq M-2$, and $C_{M-1} = (b_{M-1}, b_M)$, where $b_0 = -\infty$ and $b_M = \infty$. For $0 \leq k \leq M-1$, let $P(C_k) = \int_{C_k} f(x)\, dx$. As illustrated in Figure 2.1, the encoding proceeds as follows. For each input sample x, the quantizer determines the cell C_k such that $x \in C_k$. Next the binary code $\pi : \{0, \cdots, M-1\} \rightarrow \{0,1\}^b$ assigns

to the integer k a b-bit binary index $\pi(k)$. Let $\pi(k) = (s_0, s_1, \cdots, s_{b-1})$, where s_0 denotes the MSB. Next, in scenario A, the MSBs of all output binary codewords are encrypted, while the remaining $(b-1)$-bit portions are left unchanged. The resulted bitstream is sent to the destination. In scenario B, the MSB s_0 and the remaining $(b-1)$-bit index $(s_1, \cdots, s_{b-1})$ are losslessly encoded separately. Specifically, the sequence of MSBs is encoded using an entropy coder, thus achieving a rate close to the entropy of S_0, denoted by $H(S_0)$. This bitstream is further encrypted. The remaining $(b-1)$-bit indexes are also encoded using some encoding mechanism. The encrypted and the unencrypted portions are concatenated and transmitted to the destination.

The legitimate receiver, Bob, is able to decrypt the cypher and therefore he can recover the transmitted b-bit index $\mathbf{s}$. Then he applies the inverse mapping π^{-1} to find the label $\pi^{-1}(\mathbf{s})$ of the quantizer cell the original sample belongs to, and uses $y_{\pi^{-1}(\mathbf{s})}$ as reconstruction.

The eavesdropper, Eve, intercepts the communication between Alice and Bob, but she is not able to decipher the encrypted portion. Thus, she can only recover the least significant $(b-1)$-bit portion $\mathbf{s}' = (s_1, \cdots, s_{b-1})$ of each encoded index, but not the MSB s_0. Eve knows the binary code π since this information is not encrypted. As there are only two possibilities for the value of s_0, she concludes that the transmitted index is either $i = \pi^{-1}((0, \mathbf{s}'))$ or $j = \pi^{-1}((1, \mathbf{s}'))$, i.e., the original coefficient belongs to $C_i \cup C_j$. Based on her knowledge about the source statistics and the quantizer, Eve can select different strategies to decode the received bits. The strategies we consider in this work are listed below.

Strategy 1: When Eve knows the quantizer partition and the source pdf ($f(x)$) she can use as reconstruction the value which minimizes the distortion, i.e.,

$$y_{i,j}^{(1)} = \arg\min_{y\in\mathbb{R}} \int_{C_i\cup C_j} d(x,y)f(x)\ dx, \tag{2.1}$$

where $d : \mathbb{R}\times\mathbb{R} \to [0,\infty)$ denotes the distortion function, chosen such that the integral in (2.1) exists. Note that when $d(x,y) = (x-y)^2$, the integral exists since σ^2 is finite.

Strategy 2: If Eve does not know the source pdf, but she knows the reconstruction values y_i and y_j, a reasonable strategy is to use as reconstruction their average, i.e.,

$$y_{i,j}^{(2)} = \frac{y_i + y_j}{2}. \tag{2.2}$$

Strategy 3: There is also the possibility that Eve is not aware that encryption took place. In this case, she just assumes that the recovered b-bit index is correct and uses the reconstruction of the corresponding quantizer cell.

Let us evaluate now the distortion that Eve achieves. We will denote it by $D_E^{(l)}$, where $l = 1,2,3$, indicates the decoding strategy that she uses. Then the following holds, for $l = 1,2,3$,

$$D_E^{(l)} = \sum_{\mathbf{s}'\in\{0,1\}^{b-1}} D_{\pi^{-1}((0,\mathbf{s}')),\pi^{-1}((1,\mathbf{s}'))}, \tag{2.3}$$

where, for any pair (i,j), $0 \le i < j \le M-1$, $D_{i,j}^{(l)}$ is defined as follows

$$D_{i,j}^{(l)} = \int_{C_i\cup C_j} d(x, y_{i,j}^{(l)})f(x)\ dx, \text{ for } l = 1,2, \tag{2.4}$$

$$D_{i,j}^{(3)} = 0.5 \int_{C_i \cup C_j} d(x, y_i) f(x)\ dx + 0.5 \int_{C_i \cup C_j} d(x, y_j) f(x)\ dx. \tag{2.5}$$

It should be noted that equation (2.5) holds under the assumption that the encryption randomly flips 0s and 1s with probability 0.5.

Finally, another strategy that Eve can use is the so-called *replacement* or *error-concealment attack*. Namely, Eve is aware that encryption of the MSBs took place, and to decode the codestream she either replaces all MSBs with 0 (the MSB0 attack) or replaces all MSBs with 1 (the MSB1 attack). The replacement attack is simple and it does not require additional software that the attacker would need to purchase or skill that the attacker would need to acquire. Such attacks have also been assumed in prior work on SE of multimedia data [6, 77, 67]. The arithmetic average of the distortions achieved for the two reconstructions of the MSB0 attack and the MSB1 attack equals $D_E^{(3)}$. Therefore, we will also use $D_E^{(3)}$ as a measure of secrecy for this decoding strategy.

By analyzing the distortion formulas for strategies 2 and 3 when $d(x, y) = (x - y)^2$, we find that they are closely related. More specifically, it can be shown that for any pair i, j

$$\begin{aligned} D_{i,j}^{(2)} =& (y_j - y_i) \left(\int_{C_j} (x - y_j) f(x)\ dx - \int_{C_i} (x - y_i) f(x)\ dx \right) \\ &+ \int_{C_i} (x - y_i)^2 f(x)\ dx + \int_{C_j} (x - y_j)^2 f(x)\ dx \\ &+ 1/4 (y_j - y_i)^2 (P(C_i) + P(C_j)), \end{aligned}$$

$$D_{i,j}^{(3)} = D_{i,j}^{(2)} + 1/4 (y_j - y_i)^2 (P(C_i) + P(C_j)).$$

When $|y_j - y_i|$ is large enough, the dominant term in $D_{i,j}^{(2)}$ is the last term, thus, $D_{i,j}^{(2)} \approx 1/4(y_j - y_i)^2(P(C_i) + P(C_j))$. This observation further implies that if $|y_j - y_i|$ is sufficiently large for all connected pairs then $D_E^{(3)} \approx 2D_E^{(2)}$. We conclude that the difference in terms of secrecy between two binary codes under strategies 2 and 3 is similar. For this reason, in the sequel, we will focus our attention on strategies 1 and 3 only.

The definitions in this section can be extended in a straightforward manner to the case of finite-alphabet sources. A particular case is when each cell consists of only one element, i.e., $C_k = \{y_k\}$. We will say that the *trivial* quantizer is applied. In this situation, we have for each pair i, j

$$D_{i,j}^{(3)} = 1/2(y_j - y_i)^2(P(C_i) + P(C_j)). \tag{2.6}$$

We will say that two quantizer cells C_i and C_j, $0 \leq i < j \leq M-1$, are *connected* by the binary code π (alternatively, we say that integers i and j are connected by the binary code) if their b-bit binary representations $\pi(i)$ and $\pi(j)$ differ only in the MSB. Thus, the binary code π connects each cell with exactly one other cell. An important observation made by analyzing the definition (2.3) is that the distortion at Eve's side only depends on the way the cells are connected. Thus, all binary codes which result in the same connections lead to the same value of $D_E^{(l)}$.

The example in Figure 2.2 illustrates what a high difference the binary code can make in terms of Eve's distortion. Here, strategy 1 is assumed, which is the worst case in terms of security since it leads to the best reconstruction at Eve's side (i.e., the reconstruction which minimizes the distortion). A Gaussian source

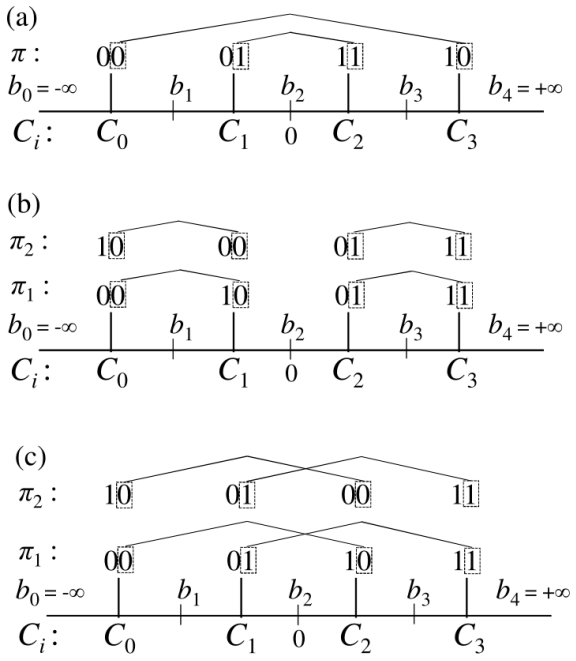


FIGURE 2.2: Different binary codes for a 4-level quantizer.

with $\mu = 0$ and $\sigma^2 = 1$ is used. The squared error is the distortion measure. The quantizer is the optimal 4-level uniform quantizer for the aforementioned source. The distortion incurred at Eve's in case (a) equals the variance of the source, thus it is as worse as if no information where available. This is the highest distortion achieved with an optimized decoder. On the other hand, in case (b) $D_E^{(1)}$ is much smaller (0.3634), while in case (c) $D_E^{(1)}$ is only slightly lower than the maximum (0.9697).

As pointed out earlier, all binary codes that generate the same set of connected pairs of quantizer cells lead to the same $D_E^{(l)}$. The way the MSB is assigned to the two cells in each pair affects the entropy $H(S_0)$. In scenario B, we are interested in reducing $H(S_0)$. Therefore, we will assign the MSB 0 to the cell with the higher probability. In this way, we obtain a binary code that maximizes $P_0 = \mathbb{P}(S_0 = 0)$

(hence minimizes $H(S_0)$) among all binary codes with the same set of connected pairs. We will say that such a binary code is an *entropy-optimized* binary code and we will understand that it is entropy-optimized for the given set of connected pairs. Let us analyze the binary codes in Figure 2.2 from the point of view of P_0 and $H(S_0)$.

For the 4-level quantizer in Figure 2.2, $P(C_0) = P(C_3) \approx 0.16$ and $P(C_1) = P(C_2) \approx 0.34$. In case (a) π is entropy-optimized for the given set of connected pairs and has $P_0 = 0.5$, hence $H(S_0) = 1$. In cases (b) and (c), π_1 and π_2 have the same set of connected pairs but only π_2 is entropy-optimized achieving $P_0 = 0.68$ and $H(S_0) = 0.9037$, while for π_1, $H(S_0) = 1$.

2.3 Problem Formulation

In this section, we formulate the problem of optimal binary code design for SE and show that it is equivalent to a maximum weight matching problem.

In scenario A the goal is to maximize the security by maximizing the distortion at Eve's side. In scenario B, we are also interested in reducing the entropy of the MSB as much as possible. As the example in Figure 2.2(a) illustrates, these two objectives are conflicting in general. Therefore, we need to settle for a trade-off between them. Such a trade-off can be achieved by choosing as an objective function to be maximized a weighted sum of $D_E^{(l)}$ and P_0.

Consequently, we formulate the problem of designing the binary code optimized for SE as

$$\max_{\pi} \; D_E^{(l)} + \lambda P_0, \tag{2.7}$$

where $\lambda = 0$ in scenario A, while $\lambda \geq 0$ in scenario B.

Next, we show that problem (2.7) can be formulated as a binary integer linear program. To this end, notice first that when $\lambda > 0$ the solution to the problem (2.7) has to be an entropy-optimized binary code for the given set of connected cells. For such a binary code, the value of the objective function depends only on the set of pairs of connected cells, observation that also holds when $\lambda = 0$. Thus, problem (2.7) can be recast as the optimization of the set of connected pairs. Let us consider, for each pair (i, j), $0 \leq i < j \leq M-1$, a binary variable $x_{i,j}$ that equals 1 if the pair (C_i, C_j) is connected by π, and equals 0 otherwise. Then the distortion $D_E^{(l)}$ can be written as

$$D_E^{(l)} = \sum_{i=0}^{M-2} \sum_{j=i+1}^{M-1} D_{i,j}^{(l)} x_{i,j}. \tag{2.8}$$

In addition, for an entropy-optimized binary code, if C_i and C_j are connected, then the cell assigned the MSB 0 must be the one with higher probability. Thus, $P_0 = \sum_{i=0}^{M-2} \sum_{j=i+1}^{M-1} P_{0,i,j} x_{i,j}$, where $P_{0,i,j} = \max(P(C_i), P(C_j))$. Consequently, the objective function in (2.7) becomes

$$D_E^{(l)} + \lambda P_0 = \sum_{i=0}^{M-2} \sum_{j=i+1}^{M-1} D_{i,j}^{(l)} x_{i,j} + \lambda \sum_{i=0}^{M-2} \sum_{j=i+1}^{M-1} P_{0,i,j} x_{i,j} = \sum_{i=0}^{M-2} \sum_{j=i+1}^{M-1} w_{i,j}^{(l)} x_{i,j}, \tag{2.9}$$

where $w_{i,j}^{(l)} = D_{i,j}^{(l)} + \lambda P_{0,i,j}$, for $l = 1, 2, 3$.

Further, for each cell C_i, the constraint that it occurs in exactly one connected pair is equivalent to the constraint that all $x_{i,j}, i < j \leq M-1$, and all $x_{k,i}, 0 \leq k < i$, equal 0, except for one of them, which equals 1. Given that the variables are binary, the aforementioned condition is equivalent to the equality $\sum_{j=i+1}^{M-1} x_{i,j} + \sum_{k=0}^{i-1} x_{k,i} = 1$. Consequently, problem (2.7) can be formulated as the following binary integer linear program,

$$\begin{aligned} \max_{(x_{i,j})_{i,j}} \quad & \textstyle\sum_{i=0}^{M-2} \sum_{j=i+1}^{M-1} w_{i,j}^{(l)} x_{i,j} \\ \text{subject to} \quad & x_{i,j} \in \{0,1\},\ 0 \leq i < j \leq M-1 \\ & \textstyle\sum_{j=i+1}^{M-1} x_{i,j} + \sum_{k=0}^{i-1} x_{k,i} = 1,\ 0 \leq i \leq M-1. \end{aligned} \tag{2.10}$$

The above problem is a weighted non-bipartite graph matching problem and can be solved in $O(M^3)$ time [50].

2.4 Linear Programming Formulation

In this section, we consider the case of a symmetric source with a symmetric quantizer and show that, for certain distortion measures, problem (2.10) can be formulated as a linear program (LP) of approximately half the size. Specifically, we will show that the above claim holds for $l = 1, 3$, when $d(x, y) = \rho(|x - y|)$, for any nondecreasing function ρ, while for $l = 2$, it holds when $d(x, y) = (x - y)^2$ and some additional conditions are satisfied.

The idea is to show that the optimal binary code has connections only from the

left side to the right side. Then we can formulate the problem with this constraint from the start, which is a maximum weight bipartite graph matching problem. It is known that the LP relaxation of the latter problem necessarily has an integer solution (to be exact, any basic feasible solution has integer components) [50], leading to the conclusion that problem (2.7) can be cast as an LP.

Let $\mathcal{I} = \{0, 1, \cdots, M/2 - 1\}$ and $\bar{\mathcal{I}} = \{M/2, \cdots, M - 1\}$. Consider now the problem formulation where connections are allowed only from the left side to the right side, i.e., $i \in \mathcal{I}, j \in \bar{\mathcal{I}}$. To this end, for each pair (i, j) with $i \in \mathcal{I}$ and $j \in \bar{\mathcal{I}}$, let $z_{i,j}$ be a binary variable that takes the value 1 if the pair (C_i, C_j) is connected by π and takes the value 0 otherwise. In this case, the problem can be formulated as follows

$$\begin{aligned} \max_{(z_{i,j})_{i\in\mathcal{I},j\in\bar{\mathcal{I}}}} \quad & \sum_{i\in\mathcal{I}} \sum_{j\in\bar{\mathcal{I}}} w_{i,j}^{(l)} z_{i,j} \\ \text{subject to} \quad & z_{i,j} \in \{0,1\},\ i \in \mathcal{I}, j \in \bar{\mathcal{I}},\ \sum_{j\in\bar{\mathcal{I}}} z_{i,j} = 1,\ i \in \mathcal{I}, \sum_{i\in\mathcal{I}} z_{i,j} = 1,\ j \in \bar{\mathcal{I}}. \end{aligned} \tag{2.11}$$

By dropping the integrality constraint in problem (2.11) we obtain the following linear program

$$\begin{aligned} \max_{(z_{i,j})_{i\in\mathcal{I},j\in\bar{\mathcal{I}}}} \quad & \sum_{i\in\mathcal{I}} \sum_{j\in\bar{\mathcal{I}}} w_{i,j}^{(l)} z_{i,j} \\ \text{subject to} \quad & z_{i,j} \geq 0,\ i \in \mathcal{I}, j \in \bar{\mathcal{I}},\ \sum_{j\in\bar{\mathcal{I}}} z_{i,j} = 1,\ i \in \mathcal{I}, \sum_{i\in\mathcal{I}} z_{i,j} = 1,\ j \in \bar{\mathcal{I}}. \end{aligned} \tag{2.12}$$

It is known that problem (2.12) has a solution that satisfies the integer constraint $z_{i,j} \in \{0, 1\},\ i \in \mathcal{I}, j \in \bar{\mathcal{I}}$, and therefore is also a solution to problem (2.11) [50].

Clearly, any feasible solution $\mathbf{z} = (z_{i,j})_{i\in\mathcal{I},j\in\bar{\mathcal{I}}}$ to problem (2.11) can be augmented to a feasible solution $\mathbf{x}(\mathbf{z}) = (x(z)_{i,j})_{0\leq i<j\leq M-1}$ to problem (2.10), where $x(z)_{i,j} = z_{i,j}$ for $i \in \mathcal{I}, j \in \bar{\mathcal{I}}$ and $x(z)_{i,j} = 0$ otherwise. The fact that $\mathbf{x}(\mathbf{z})$ is a feasible solution to problem (2.10) can be easily checked. Likewise, it follows that $F_1(\mathbf{x}(\mathbf{z})) = F_2(\mathbf{z})$, where F_1 and F_2 denote the cost functions of problems (2.10) and (2.11), respectively.

In order to present the main result of this section, we need to establish some new terminology and notations. We say that source X (or the pdf $f(x)$) is symmetric about μ if $f(x) = f(\mu - x)$ for each $x \in \mathbf{R}$. For each $i, 0 \leq i \leq M-1$, denote $\bar{i} = M-1-i$. For each set $S \subseteq \mathbb{R}$, denote $\mu - S = \{\mu - x | x \in S\}$. We say that the M-level quantizer Q is symmetric about μ if $b_{M/2} = \mu$, $C_i = \mu - C_{\bar{i}}$ and $y_i = \mu - y_{\bar{i}}$, for each $0 \leq i \leq M-1$. Finally, for each $0 \leq i \leq M-1$, denote $\mu_i = \frac{\int_{C_i} xf(x)\ dx}{P(C_i)}$. In the sequel, we assume that for each $0 \leq i \leq M-1$, $y_i \in [b_i, b_{i+1}]$.

Proposition 1. *Assume that the source X and the quantizer Q are symmetric about μ. Then, for any solution $\mathbf{z}^{(0)}$ to problem (2.11), $\mathbf{x}(\mathbf{z}^{(0)})$ is a solution to problem (2.10), for $l \in \{1,2,3\}$, when one of the following holds:*

i) $l \in \{1,3\}$ and $d(x,y) = \rho(|x-y|)$, where ρ is nondecreasing;

ii) $l = 2$, $d(x,y) = (x-y)^2$ and, for any $i \in \mathcal{I}$, one has $y_i - \mu \geq 2(\mu_i - \mu)$.

Proof. When $d(x,y) = \rho(|x-y|)$, shifting both the pdf and the quantizer by the same amount does not change the values $w_{i,j}^{(l)}$, $0 \leq i < j \leq M-1$. Thus, we may assume that $\mu = 0$.

In the following argument we will use Lemmas (1)-(5), stated and proved in the appendix of Chapter 2 A. Let $F_{1,opt}$ and $F_{2,opt}$ denote the optimal cost for problems (2.10) and (2.11), respectively. Since for any feasible solution $\mathbf{z} = (z_{i,j})_{i\in\mathcal{I},j\in\bar{\mathcal{I}}}$ to problem (2.11), $\mathbf{x}(\mathbf{z}) = (x(z)_{i,j})_{0\leq i<j\leq M-1}$ is also a feasible solution for problem (2.10) and $F_1(\mathbf{x}(\mathbf{z})) = F_2(\mathbf{z})$, it follows that

$$F_{1,opt} \geq F_{2,opt}. \tag{2.13}$$

Next we prove that, when one of conditions i) and ii) is satisfied, one has $F_{1,opt} \leq F_{2,opt}$. For this, let $\mathbf{x}^{(0)} = (x_{i,j}^{(0)})_{0\leq i<j\leq M-1}$ be a solution to problem (2.10) and define $\bar{\mathbf{x}}^{(0)} = (\bar{x}_{i,j}^{(0)})_{0\leq i<j\leq M-1}$, where $\bar{x}_{i,j}^{(0)} = x_{\bar{j},\bar{i}}^{(0)}$, $0 \leq i < j \leq M-1$. Lemma 1 implies that $w_{i,j}^{(l)} = w_{\bar{j},\bar{i}}^{(l)}$, for all $0 \leq i < j \leq M-1$. Then $\mathbf{x}^{(0)}$ is also a solution to problem (2.10). Consider now $\mathbf{x}^{(1)} = (x_{i,j}^{(1)})_{0\leq i<j\leq M-1}$, where $x_{i,j}^{(1)} = \frac{1}{2}\left(x_{i,j}^{(0)} + \bar{x}_{i,j}^{(0)}\right)$, $0 \leq i < j \leq M-1$. (Notice that $x_{i,j}^{(1)}$ are not necessarily integer values anymore.) Since $\mathbf{x}^{(1)}$ is a convex combination of two solutions to problem (2.10), it follows that $F_1(\mathbf{x}^{(1)}) = F_{1,opt}$ and that $\mathbf{x}^{(1)}$ satisfies the equality constraints, i.e., for any $0 \leq i \leq M-1$,

$$\sum_{j=i+1}^{M-1} x_{i,j}^{(1)} + \sum_{k=0}^{i-1} x_{k,i}^{(1)} = 1. \tag{2.14}$$

Additionally, notice that $\mathbf{x}^{(1)}$ has the following symmetry property:

$$x_{i,j}^{(1)} = x_{\bar{j},\bar{i}}^{(1)}, \text{ for all } 0 \leq i < j \leq M-1. \tag{2.15}$$

Let us construct now the tuple $\mathbf{z}^{(1)} = (z_{i,j}^{(1)})_{i\in\mathcal{I},j\in\bar{\mathcal{I}}}$ as follows

$$z_{i,j}^{(1)} = \begin{cases} x_{i,\bar{i}}^{(1)} & \text{if} \quad j = \bar{i} \ (\iff i = \bar{j}) \\ x_{i,j}^{(1)} + x_{i,\bar{j}}^{(1)} & \text{if} \quad j < \bar{i} \ (\iff i < \bar{j}) \\ x_{i,j}^{(1)} + x_{\bar{j},i}^{(1)} & \text{if} \quad j > \bar{i} \ (\iff i > \bar{j}) \end{cases} . \tag{2.16}$$

According to Lemma 2, $\mathbf{z}^{(1)}$ is a feasible solution to problem (2.12). Since problem (2.12) has a solution that satisfies the integer constraints of problem (2.11) [50], it follows that the two problems achieve the same cost at optimality, i.e., $F_{2,opt}$, and further that $F_2(\mathbf{z}^{(1)}) \leq F_{2,opt}$.

According to Lemmas 3 and 4, one has $w_{i,j}^{(l)} \leq w_{i,\bar{j}}^{(l)}$ and $w_{i,j}^{(l)} \leq w_{j,\bar{i}}^{(l)}$, for all $0 \leq i < j \leq M/2 - 1$.

In addition, Lemma 5 can be further applied leading to the conclusion that $F_2(\mathbf{z}^{(1)}) \geq F_1(\mathbf{x}^{(1)})$. Since $F_{1,opt} = F_1(\mathbf{x}^{(1)})$, it follows that $F_{1,opt} \leq F_{2,opt}$. Combining with (2.13), one obtains that $F_{1,opt} = F_{2,opt}$. Now let $\mathbf{z}^{(0)}$ be a solution to problem (2.11). Then one has $F_1(\mathbf{x}(\mathbf{z}^{(0)})) = F_2(\mathbf{z}^{(0)}) = F_{2,opt} = F_{1,opt}$, which completes the proof. □

Remark 1. *Note that the condition $y_i - \mu \geq 2(\mu_i - \mu)$ is satisfied for all $i \in \mathcal{I}$ when the pdf and the quantizer are symmetric about μ and the codebook of the quantizer Q is optimized for the squared error distortion, since in this case $y_i = \mu_i < \mu$. Moreover, if Q is uniform with step size $\Delta > 0$, then $b_{i+1} = \mu + (-M/2 + i + 1)\Delta$, and $y_i = b_{i+1} - \Delta/2$, for each $i \in \mathcal{I}$. Clearly, for $0 \leq i \leq M/2 - 2$, one has*

$(y_i - \mu)/2 > b_{i+1} - \mu > \mu_i - \mu$. *Thus, the only situation when the condition is not automatically guaranteed is for* $i = M/2 - 1$.

2.5 Experimental Results

In this section, we assess the practical performance of the optimized binary code for several i.i.d. sources. Specifically, we consider the Gaussian and Laplacian sources with $\mu = 0$ and $\sigma^2 = 1$ since they are commonly used to model the distribution of transform coefficients and of prediction residuals [31]. In addition, we consider a mixed Gaussian source with pdf

$$f(x) = 0.7\left(\frac{1}{\sqrt{2\pi}}e^{-(x-\nu_1)^2/2}\right) + 0.3\left(\frac{1}{\sqrt{2\pi}}e^{-(x-\nu_2)^2/2}\right),$$

where $\nu_1 = -2$, and $\nu_2 = 2$ (the mean is -0.80 and the variance is 4.36).

The distortion measure is the squared error. In each case, we consider uniform quantizers with M bins for each $M \in \{4, 8, 16, 32, 64, 128\}$. In a uniform quantizer, all bounded cells are intervals of the same size and the reconstruction values are the midpoints of the bounded intervals. Each such quantizer is characterized by the step size Δ and the shift δ. Then $b_{i+1} = \delta + (-M/2 + i + 1)\Delta$, and $y_i = b_{i+1} - \Delta/2$, for each $0 \leq i \leq M - 2$, while $y_{M-1} = b_{M-1} + \Delta/2$. The values of Δ and δ are chosen such that the distortion is minimized for the corresponding source and number of levels. Specifically, the values of Δ (in increasing order of M) are $0.9957, 0.586, 0.3352, 0.1881, 0.1041, 0.0569$, for the Gaussian source, $1.0873, 0.7309, 0.4609, 0.2799, 0.1657, 0.0961$ for the Laplacian source and $1.81, 1, 0.54, 0.29, 0.16, 0.08$ for the mixed Gaussian pdf. The shift δ is 0 for the Gaussian and Laplacian pdfs, while for the mixed Gaussian it takes the values

$-0.21, -0.17, -0.15, -0.14, -0.12, -0.12$. Note that in the Gaussian and Laplacian cases, both the pdf and the quantizer are symmetric about 0, which is not true for the mixed Gaussian distribution.

In the sequel, the distortion will be represented in dB, i.e., as $10\log_{10} D$. We will use the acronym OPT for the optimized binary code. When we need to specify the strategy l and the value of λ we add (l) as a superscript and λ as a subscript. Thus, $\text{OPT}_{\lambda}^{(l)}$ refers to the binary code obtained by solving the problem (2.7) for $l = 1, 3$. After converting the problem to the form (2.10) we use the MATLAB linear programming solver, unless otherwise specified.

For comparison with the state of the art, we choose NBC and FBC. Before proceeding to the presentation of the experimental results, we first provide the definition of the binary codes NBC and FBC and justify why we use them for the performance comparison.

2.5.1 NBC and FBC

NBC is the binary code π defined as follows: for $0 \leq k \leq M-1$, $\pi(k) = (s_0, \cdots, s_{b-1})$, where $k = \sum_{i=1}^{b} s_{i-1} 2^{b-i}$. The reversed NBC (RNBC) is the binary code obtained from NBC by flipping the MSB s_0. FBC is the binary code π, where $\pi(k)$ is the same as the RNBC codeword for $k \geq M/2$, while for $k < M/2$, $\pi(k)$ equals the NBC codeword corresponding to $M-1-k$. Table 2.1 illustrates the three binary codes when $b = 3$.

Notice that NBC and RNBC have the same set of connected pairs of integers, namely $\{(k, k + M/2) : 0 \leq k \leq M/2 - 1\}$. Thus, they have the same values for

TABLE 2.1: Binary codes NBC, RNBC, and FBC, for $M = 8$.

k	0	1	2	3	4	5	6	7
NBC	000	001	010	011	100	101	110	111
RNBC	100	101	110	111	000	001	010	011
FBC	111	110	101	100	000	001	010	011

$D_E^{(l)}$ and $H(S_0)$. For FBC, the set of connected pairs of integers is $\{(k, M-1-k) : 0 \leq k \leq M/2-1\}$.

In the JPEG standard, the amplitudes of the non-zero AC coefficients are encoded by dividing them into categories and using a fixed-length binary code for each category. Specifically, for $t \geq 1$, category t is the set $\mathcal{I}_t = \{k \in \mathbb{N} : 2^{t-1} \leq |k| \leq 2^t - 1\}$. The elements in $\mathcal{I}_t$ are encoded with a t-bit binary code. Consider relabeling these integers in their increasing order with labels from 0 to $2^b - 1$ and let π denote the binary code applied to these labels. Then the code π specified by the standard is RNBC. Since NBC and RNBC have the same values of $D_E^{(l)}$ and $H(S_0)$, we use NBC in our comparison. Note that SE methods that encrypt the sign bit of the non-zero AC coefficients can be regarded as the application of scenario A in each category.

In the baseline profile of the H.264/AVC standard, the MVDs are encoded using the 0th order Exp-Golomb code [57]. As a result, all integers in the set $\mathcal{I}_t$ are encoded with a $(2t + 1)$-bit binary code. For fixed t, only the last t bits of the codeword are different for different integers in $\mathcal{I}_t$. Therefore, we can consider the portion formed of the last t bits as the effective binary code π applied to $\mathcal{I}_t$. The last bit of the codeword is the sign bit. Thus, if we interpret the last bit as

the MSB, the SE methods that encrypt the sign bits of the MVDs correspond to scenario A applied to each category. If we consider that π is applied to the labels 0 to $2^b - 1$ after relabeling the integers in $\mathcal{I}_t$ in their increasing order, then the binary code π specified by the H.264/AVC standard corresponds to FBC (with the provision that the MSB is moved to the last position, while all the other bits remain in the same order).

In the main profile of the H.264/AVC standard, the MVDs are first binarized, i.e., converted to intermediate binary codewords, which are subsequently encoded with a binary arithmetic coder. The binarization process uses a third order Exp-Golomb code, which divides all MVDs with the absolute value larger than 9 into categories, and assigns binary codewords of the same length within each category. Like in the previous case, the effective code within each category corresponds to FBC. SE methods that encrypt the sign bit of the MVDs before applying the arithmetic coder can also be regarded as an instance of scenario A applied to each category, with the code π being FBC.

2.5.2 Gaussian and Laplacian Sources

Table 2.2 presents the distortion at Eve's obtained by the optimized binary code in scenario A in comparison with NBC and FBC for the Gaussian and Laplacian sources. The results show that for both sources FBC is optimal under strategy 1, but has very poor performance under strategy 3. The performance gap from the optimum when $l = 3$ increases as M increases and is more pronounced for the Laplacian source, reaching more than 9 dB at $M = 128$. Interestingly, NBC is optimal or very close to optimal under strategy 3 but has weak performance

under strategy 1 for the higher values of M. The gap in performance relative to the optimum is also higher for the Laplacian source.

TABLE 2.2: $D_E^{(l)}$ for $\text{OPT}_0^{(l)}$, NBC, and FBC when $l = 1, 3$, for the Gaussian and Laplacian sources.

		M	4	8	16	32	64	128
$l=1$	Gaussian	$\text{OPT}_0^{(1)}$, FBC	0	0	0	0	0	0
		NBC	−0.13	−0.03	−0.16	−0.50	−1.00	−1.66
$l=1$	Laplacian	$\text{OPT}_0^{(1)}$, FBC	0	0	0	0	0	0
		NBC	−0.01	−0.15	−0.76	−1.65	−2.78	−4.13
$l=3$	Gaussian	$\text{OPT}_0^{(3)}$	3.15	4.41	5.560	6.562	7.444	8.217
		NBC	3.15	4.41	5.559	6.559	7.441	8.215
		FBC	2.74	2.93	2.99	3.003	3.008	3.010
$l=3$	Laplacian	$\text{OPT}_0^{(3)}$, NBC	3.82	6.25	8.29	10.00	11.48	12.77
		FBC	2.56	2.85	2.95	2.99	3.00	3.01

Let us discuss now the performance of $\text{OPT}_\lambda^{(1)}$ in scenario B. Figure 2.3 demonstrates how the variation of λ changes the optimal BC π when $M = 16$. The BC of Figure 2.3(a) is obtained when $\lambda = 0$, meaning that the optimization criterion is just maximizing D_E without considering the minimization of $H(S_0)$. Therefore, the OPT achieves the maximum D_E, which is exactly equal to the variance of the input distribution. To see this, note that each C_i is connected to its symmetric cell about 0, i.e., C_{M-1-i}, therefore the centroid of their union is 0, which is the mean of the pdf f_X. For this BC, P_0 is equal to 0.5 leading to $H(S_0) = 1$ bit, which is the maximum possible entropy for a binary RV.

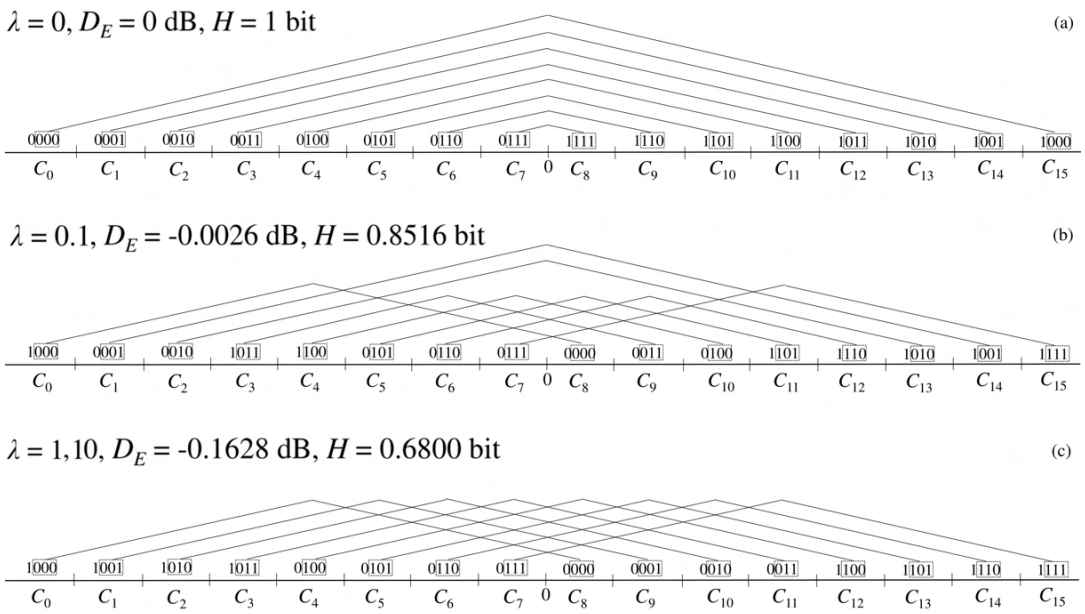


FIGURE 2.3: Optimized index assignments for a 16-level quantizer.

In Figure 2.3(b), the criterion of increasing P_0 is also accounted for in the optimization objective by setting $\lambda = 0.1$. We can see that even if λ is very small, the resulting BC has a different set of connected pairs of cells in comparison to case (a), in order to allow for an increase of P_0, and hence a decrease of the entropy of the MSB. As a result of the trade-off between D_E and P_0, D_E decreases, but only very slightly. On the other hand, the decrease in the entropy of the MSB is quite significant - about 15%.

Figure 2.3(c) shows an BC obtained when the optimization places more emphasis on the maximization of P_0 by considering a large value of λ. Specifically, $\lambda = 10$. The obtained BC has the same set of connected pairs of cells as NBC, but it is entropy-optimized for the given distortion. Thus, each cell C_i is connected to $C_{i+M/2}$, for $0 \leq i < M/2$. Actually, this BC achieves the smallest entropy of the MSB, namely 0.68, since the cells assigned the MSB 0 are the eight cells with

the highest probability, namely $C_4, C_5, \cdots, C_{11}$. As expected, the reduction of the entropy leads to the reduction of D_E as well, which is now -0.1628 dB.

Figures 2.4 and 2.5 plot the value of $D_E^{(1)}$ versus the entropy $H(S_0)$ obtained in scenario B when $M = 4, 8, 16, 32, 64, 128$, in the case of the Gaussian source, respectively the Laplacian source. For each M, various points are obtained by gradually increasing λ from 0 up to some very large value.

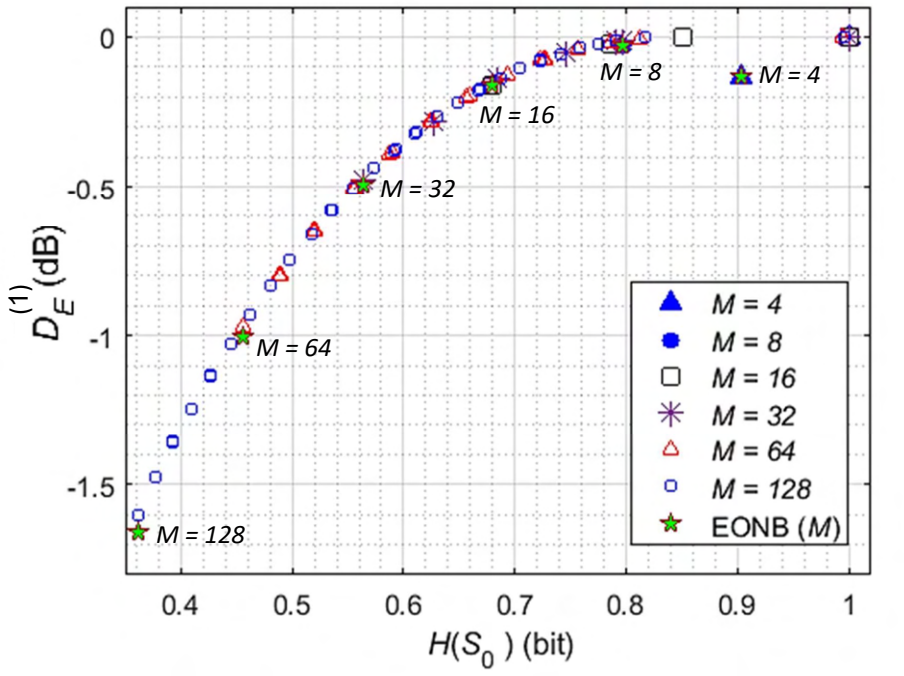


FIGURE 2.4: Plot of $D_E^{(1)}$ versus $H(S_0)$ for $M = 4, 8, 16, 32, 64, 128$, for the Gaussian source.

For both sources, $\text{OPT}_0^{(1)}$ achieves the highest secrecy since $D_E^{(1)}$ is the largest, but also the highest encryption complexity as $H(S_0) = 1$ is also the largest. As λ gradually increases above 0, the complexity of the encryption decreases at the expense of decreasing the secrecy. However, it is noteworthy that at the beginning,

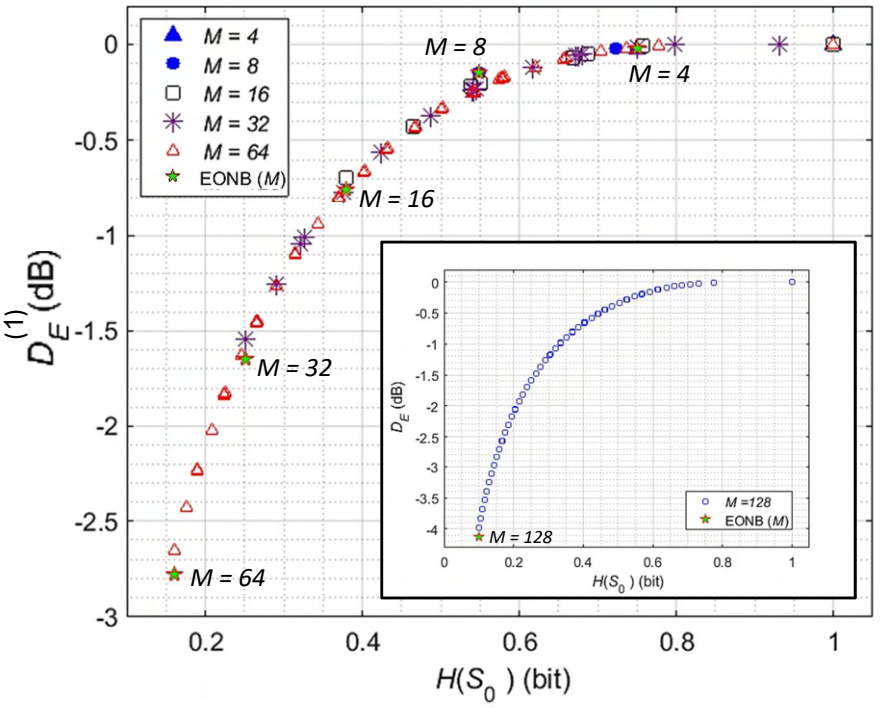


FIGURE 2.5: Plot of $D_E^{(1)}$ versus $H(S_0)$ for $M = 4, 8, 16, 32, 64, 128$, for the Laplacian source.

$H(S_0)$ decreases at a high rate, while the reduction in $D_E^{(1)}$ is very slow. In particular, when $M \geq 8$, $H(S_0)$ can be reduced from 1 to 0.8 or less, while decreasing $D_E^{(1)}$ by only ≈ 0.025 dB. Based on this observation, it follows that NBC and FBC have very poor performance in scenario B since they have $H(S_0) = 1$, i.e., the largest encryption complexity.

Another observation is that the entropy-optimized NBC (EONB) has the smallest value of $H(S_0)$. This is because the $M/2$ cells with the largest probabilities are $C_{M/4}, \cdots, C_{3M/4-1}$, and each of them is connected by EONB to a cell which is not in this set, thus each of them is assigned the MSB 0 by EONB. Table 2.3 shows the values of $H(S_0)$ for $\text{OPT}_0^{(l)}$, $l = 1, 3$, NBC, FBC and EONB for all M. The pairs

TABLE 2.3: $H(S_0)$ for $\text{OPT}_0^{(l)}$, NBC, EONB, and FBC when $l = 1, 3$ for the Gaussian and Laplacian sources.

	M	4	8	16	32	64	128
Gaussian	$\text{OPT}_0^{(1)}$, NBC, FBC	1	1	1	1	1	1
	$\text{OPT}_0^{(3)}$, EONB	0.90	0.80	0.68	0.56	0.46	0.36
Laplacian	$\text{OPT}_0^{(1)}$, NBC, FBC	1	1	1	1	1	1
	$\text{OPT}_0^{(3)}$, EONB	0.75	0.55	0.38	0.25	0.16	0.10

$(H(S_0, D_E^{(1)})$ corresponding to EONB are marked by red stars in Figures 2.4 and 2.5. We see that they are very close to the pairs obtained by $\text{OPT}_\lambda^{(1)}$ corresponding to similar values of $H(S_0)$.

Finally, another key observation for strategy 1 is that there is a large number of trade-off pairs $(H(S_0), D_E^{(1)})$ and they cover a wide range of values of $H(S_0)$ and $D_E^{(1)}$, especially for higher M. Under strategy 3, on the other hand, $\text{OPT}_0^{(3)}$ already achieves the highest distortion $D_E^{(3)}$ and the lowest entropy $H(S_0)$, for both the Gaussian and Laplacian sources. In other words, maximizing the secrecy and minimizing the complexity of encryption can be done simultaneously and there is no further impact by increasing λ above 0. Also note that EONB has the same performance as $\text{OPT}_0^{(3)}$ in scenario B, while NBC and FBC are considerably inferior since their entropy $H(S_0)$ is much lower than that of $\text{OPT}_0^{(3)}$.

2.5.3 Mixed Gaussian Source

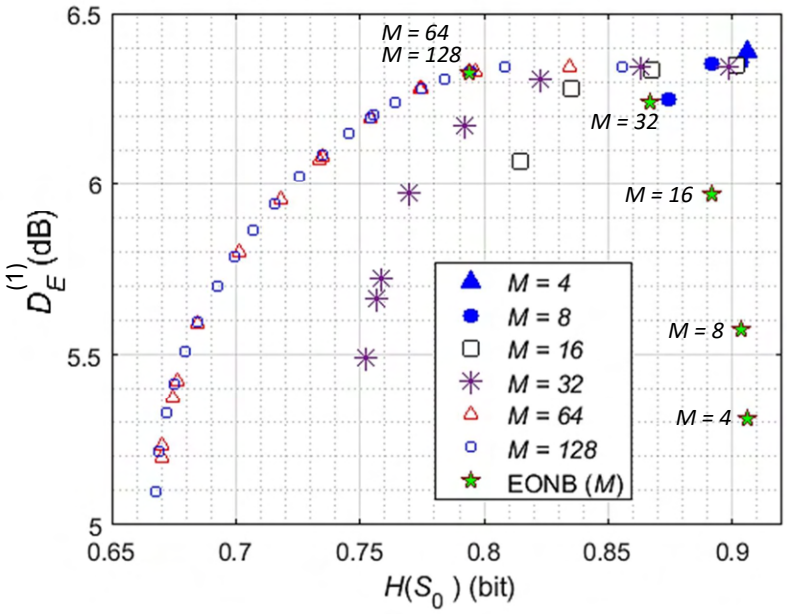
Table 2.4 illustrates the comparison in performance between the optimized binary code, NBC, and FBC for the mixed Gaussian source in scenario A. It can be seen that under strategy 1, FBC is optimal or very close to optimal. NBC is

suboptimal, but the gap to the optimal performance narrows as M increases, unlike for the Gaussian and Laplacian sources. Under strategy 3 both NBC and FBC have weaker performance than $\text{OPT}_0^{(3)}$, except for FBC when $M = 4$. The gap between NBC and $\text{OPT}_0^{(3)}$ is between 0.8 and 1.1 dB, while the gap between FBC and $\text{OPT}_0^{(3)}$, for $M \geq 16$, is between 1 and 2.16 dB.

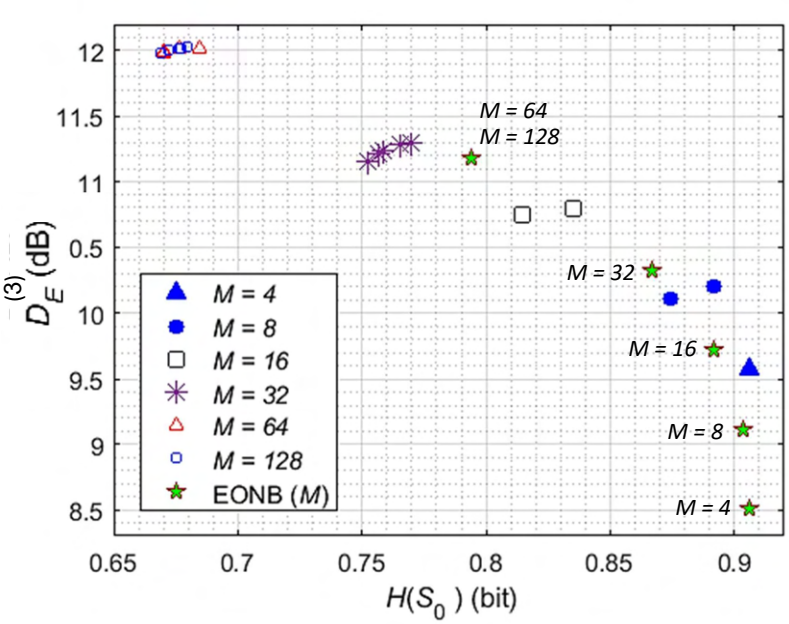
TABLE 2.4: $D_E^{(l)}$ for $\text{OPT}_0^{(l)}$, NBC, and FBC when $l = 1, 3$, for the mixed Gaussian source.

	M	4	8	16	32	64	128
$l = 1$	$\text{OPT}_0^{(1)}$	6.39	6.36	6.35	6.34	6.344	6.345
	NBC	5.31	5.57	5.97	6.24	6.32	6.32
	FBC	6.39	6.36	6.35	6.34	6.341	6.340
$l = 3$	$\text{OPT}_0^{(3)}$	9.58	10.21	10.79	11.30	12.02	12.02
	NBC	8.51	9.11	9.72	10.33	11.18	11.18
	FBC	9.58	9.74	9.79	9.81	9.84	9.84

Let us consider now scenario B. Figures 2.6a and 2.6b plot the value of $D_E^{(l)}$, for $l = 1$, respectively $l = 3$, versus $H(S_0)$, for $\text{OPT}_\lambda^{(l)}$ with various values of λ, when $M = 4, 8, 16, 32, 64, 128$. We can observe that by varying λ, various trade-off pairs $(H(S_0), D_E^{(3)})$ can be obtained. The number of these pairs and the range of values covered are larger for strategy 1. As seen from Table 2.5, $\text{OPT}_0^{(1)}$ achieves a high entropy $H(S_0)$. As Figure 2.6a shows, for $M \geq 16$, a large decrease of $H(S_0)$ can be obtained with only a small decrease of $D_E^{(1)}$. Since according to Table 2.5, NBC and FBC have the value of $H(S_0)$ larger than or equal to that of $\text{OPT}_0^{(1)}$, it follows that both are inferior to $\text{OPT}^{(1)}$ in scenario B for $M \geq 16$. The same conclusion holds for NBC when $M = 4, 8$ since in this case, NBC has the value of $D_E^{(1)}$ much



(A) $l = 1$



(B) $l = 3$

FIGURE 2.6: Plot of $D_E^{(l)}$ versus $H(S_0)$ when $M = 4, 8, 16, 32, 64, 128$, for the mixed Gaussian source.

smaller than that of $\text{OPT}_0^{(1)}$, while $H(S_0)$ is the same. Under strategy 3, when $M \geq 8$, we observe that NBC and FBC have a larger value of $H(S_0)$ than $\text{OPT}_0^{(3)}$, while the distortions $D_E^{(3)}$ are much lower, thus they have poorer performance than $\text{OPT}_0^{(3)}$ in scenario B. This conclusion extends to the case when $M = 4$ for NBC.

TABLE 2.5: $H(S_0)$ for $\text{OPT}_0^{(1)}$, $\text{OPT}_0^{(3)}$, NBC, EONB, and FBC for the mixed Gaussian source.

M	4	8	16	32	64	128
$\text{OPT}_0^{(1)}$	0.91	0.90	0.90	0.898	0.83	0.86
$\text{OPT}_0^{(3)}$	0.91	0.89	0.84	0.77	0.66	0.68
NBC, FBC	0.91	0.90	0.90	0.901	0.90	0.90
EONB	0.91	0.90	0.89	0.86	0.79	0.79

Figures 2.6a and 2.6b also contain the pairs $(H(S_0, D_E^{(1)})$ corresponding to EONB. Notice that they are far away from the pairs corresponding to $\text{OPT}_\lambda^{(l)}$, for all M under strategy 3 and for all $M \leq 32$ under strategy 1. We conclude that EONB is inferior to $\text{OPT}_\lambda^{(l)}$ in these cases. The exception is when $M \geq 64$ in strategy 1. In this case EONB is very close to the optimum.

2.6 Conclusion

In this chapter, we considered the scenario where SE is applied to a sequence of compressed quantization indexes. Specifically, only the plane of MSBs (possibly after being entropy-coded) is encrypted. This corresponds to existing SE schemes for uncompressed images and compressed images or videos, where only the sign bits of some syntax elements are encrypted. We observe that the binary code, i.e,

the mapping of binary sequences to quantization cells, can control the level of security by influencing the distortion at the eavesdropper's side, while also impacting the amount of computation needed for encryption by affecting the length of the compressed MSB stream. Therefore, we formulate the problem of optimal binary code design for SE as the maximization of a weighted sum of the eavesdropper's distortion and of the probability of the MSB being 0. We show that the problem is equivalent to a maximum weight matching problem, which can be solved in polynomial time. Moreover, we prove that, when the source and the quantizer are symmetric, the problem can be cast as a linear program. Experimental comparison with commonly used binary codes shows that in certain situations the proposed design could bring considerable improvement.

Chapter 3

BC Designed for the SE of Uncompressed Images

SE methods for uncompressed images often encrypt MSBs of the binary representations of pixel intensity values. Traditionally, NBC has been used to represent these values. However, the SE method based on MSB encryption cannot provide high security when NBC is used, since it cannot achieve sufficient degradation of Eve's reconstructions. This chapter proposes two BC design methods that will provide higher degradation on Eve's reconstructions by MSB encryption. First, we apply the concept of BC design of Chapter 2 to the SE of uncompressed images. Here the binary code is applied to obtain a binary representation of each pixel value, and the performance of the proposed optimized BC will be investigated for the SE of uncompressed or "lightly" compressed images. The SE is obtained by encrypting one or two MSB planes within two scenarios. In scenario A, the images are not compressed. In scenario B, only the MSB plane is compressed before being encrypted. As the second method, in this chapter, we develop structured BCs

to increase the security of SE without the computational overhead of the optimization by exploiting the fact that in natural images, the pixels that are in close proximity tend to have close intensity values as well. For both optimization and structured design approaches, we show that higher visual secrecy can be obtained in comparison with the conventional method of coding.

This chapter is organized as follows. Section 3.1 explains how the optimized BC design will be applied to code the pixel values, and the next section demonstrates the performance of the optimized BC in comparison with NBC. In Section 3.3, the design method of structured BCs will be introduced. Sections 3.4 and 3.5 describe how to develop structured BCs that enhance the security of MSB-SE of uncompressed images compared with NBC.

3.1 Optimized BCs

In our desing, we consider gray-level images with 8-bit pixel values. Thus, the intensity values of the pixels are integers ranging from 0 to 255. The binary code π maps each integer in this range to an 8-bit sequence. We will consider two cases, i.e., encryption of one MSB plane, 1 bit plane encryption (1BPE) and encryption of two MSB planes, 2 bit planes encryption (2BPE). The binary code will be optimized for 1BPE, but its practical performance will be assessed for both 1BPE and 2BPE.

We assume that Eve uses a replacement attack. Thus, in the case of 1BPE, she uses two reconstructions, obtained by replacing all MSBs by 0, respectively

by 1. Under 2BPE, Eve uses four reconstructions, obtained by replacing the two unknown MSB bits by 00, 01, 10 and 11, respectively.

To optimize the binary code, we solve the optimization problem (2.7) for $l = 3$, $\lambda = 0$ and $\lambda = 10^6$. For this, we convert the problem to the form (2.10) and solve it using an integer programming package in Python 3 (the MIP package). Note that the solution to the problem (2.10) only specifies the connected pairs i, j, i.e., the pairs having the same 7-bit LSB (least significant bit) representation. It does not specify how to assign the 7-bit LSB sequences to these pairs. This assignment does not impact the image degradation at Eve's side under 1BPE, but it does under 2BPE. Our intuition is that a random assignment will decrease the correlation between adjacent pixels in the reconstruction at Eve's side, which is desired for higher secrecy. Therefore, we will use a random assignment of 7-bit LSB sequences to the connected pairs for both scenarios A and B. Likewise, the assignment of the MSB value to each element in a connected pair does not affect the value of $D_E^{(3)}$. However, it influences the visual distortion since adjacent pixels in an image are correlated. Therefore, in scenario A, we will randomly assign the MSB to each integer in a connected pair.

Recall that in scenario B, our optimization procedure uses $H(S_0)$ as a measure of the encryption complexity. Since the adjacent pixel values are correlated, the entropy of the MSB plane is no longer guaranteed to be equal to $H(S_0)$. However, we expect that by increasing $H(S_0)$, the entropy of the MSB plane to increase too. The assignment of the MSB value to the elements in a connected pair influences $H(S_0)$. Therefore, in scenario B we use the entropy-optimized binary code, i.e., the MSB 0 is assigned to the integer in the pair having the largest probability.

We use the acronyms OPTA and OPTB for the binary codes obtained as specified above in scenario A, respectively B. When we are interested in specifying the value of λ, we add it as a subscript to OPTB. The performance of OPTA and OPTB will be compared against NBC, which is the binary code used in the prior work on SE schemes that encrypt the high order MSBs of the pixel values [79, 28, 59, 41, 56].

3.2 Experimental Results for Optimized BCs

We consider five gray-level images of size 512×512, with 8-bit pixel values. The images are shown in Figure 3.1. To measure the degradation in the reconstruction at Eve's side, we will use both the visual assessment and the objective assessment based on evaluating the PSNR and SSIM [73]. We will also consider the correlation between adjacent pixel values as a measure of security of each technique. In Table 3.1 the PSNR, SSIM, and the diagonal correlation (DCorr) for the reconstructions at Eve's side are presented for the binary codes under investigation. The PSNR is computed based on the average MSE over all reconstructions used at Eve's side. The SSIM and DCorr values are also averaged over all reconstructions.

We observe from Table 3.1 that the PSNR values for OPTA and OPTB are always smaller than for NBC in case of 1BPE. This is expected since the maximization of the distortion, hence the minimization of the PSNR, under 1BPE is accounted for in the optimization objective. For all images except Zelda, this observation holds for 2BPE, too. While the difference in terms of PSNR between the optimized binary codes and NBC is not that dramatic, we see a significant difference in SSIM. Note that for NBC, the SSIM values are at least 0.2 in most

FIGURE 3.1: Test images, from top left to the bottom right: Lena, Cameraman, Livingroom, Goldhill, and Zelda.

of the cases, while for OPTA the SSIM is extremely low (≤ 0.01). The SSIM for OPTB, although higher than for OPTA, is still lower than half of the SSIM value of the NBC in all cases except for Cameraman under 1BPE. As for the diagonal correlation, it can be noticed that NBC has high values (over 0.5 for 1BPE and over 0.3 for 2BPE), while OPTA significantly decreases the correlation. More specifically, for four out of the five images, DCorr for OPTA is very low, i.e., at most 0.12. The values of DCorr for OPTB, although higher than for OPTA, are still lower than for NBC with only one exception (Cameraman at 1BPE).

Figures 3.2 and 3.3 show the reconstructions at Eve's side for Lena, for the three binary codes under comparison in the case of 1BPE, respectively 2BPE. The

TABLE 3.1: Performance of 1BPE and 2BPE in terms of secrecy using objective measures.

Images	**Measure**	NBC		OPTA		$OPTB_0$		$OPTB_{10^6}$	
		1BPE	2BPE	1BPE	2BPE	1BPE	2BPE	1BPE	2BPE
Cameraman	PSNR(dB)	9.00	8.96	7.38	7.83	7.38	7.83	7.64	8.01
	SSIM	0.23	0.44	0.01	0.01	0.41	0.06	0.44	0.05
	DCorr	0.65	0.64	0.40	0.18	0.81	0.32	0.84	0.30
Goldhill	PSNR(dB)	9.00	8.96	8.76	8.88	8.76	8.88	8.83	8.88
	SSIM	0.20	0.25	0.01	0.01	0.09	0.09	0.08	0.08
	DCorr	0.55	0.37	0.07	0.07	0.18	0.18	0.23	0.23
Lena	PSNR(dB)	9.00	8.97	8.69	8.95	8.69	8.95	8.71	8.96
	SSIM	0.25	0.35	0.01	0.01	0.05	0.05	0.04	0.04
	DCorr	0.69	0.51	0.11	0.11	0.23	0.23	0.19	0.19
Livingroom	PSNR(dB)	9.00	9.37	8.69	9.00	8.69	9.00	8.72	9.02
	SSIM	0.15	0.26	0.01	0.01	0.07	0.07	0.07	0.07
	DCorr	0.60	0.41	0.10	0.10	0.23	0.23	0.21	0.21
Zelda	PSNR(dB)	9.00	8.86	8.84	8.89	8.84	8.89	8.84	8.90
	SSIM	0.27	0.38	0.00	0.00	0.08	0.08	0.08	0.08
	DCorr	0.78	0.59	0.12	0.12	0.29	0.29	0.30	0.30

reconstructions corresponding to the remaining images are provided in Appendix B. It can be seen that OPTA leads to significant image degradation in both 1BPE and 2BPE. NBC, on the other hand, fails at obscuring all the details, even in the case of 2BPE. The same conclusions can be drawn for the other images, based on the reconstructions shown in Appendix B. We conclude that OPTA ensures a high level of content secrecy even when only one MSB plane is encrypted.

The quality degradation achieved with OPTB in case of 1BPE is similar to that corresponding to NBC. On the other hand, in the case of 2BPE, OPTB ensures

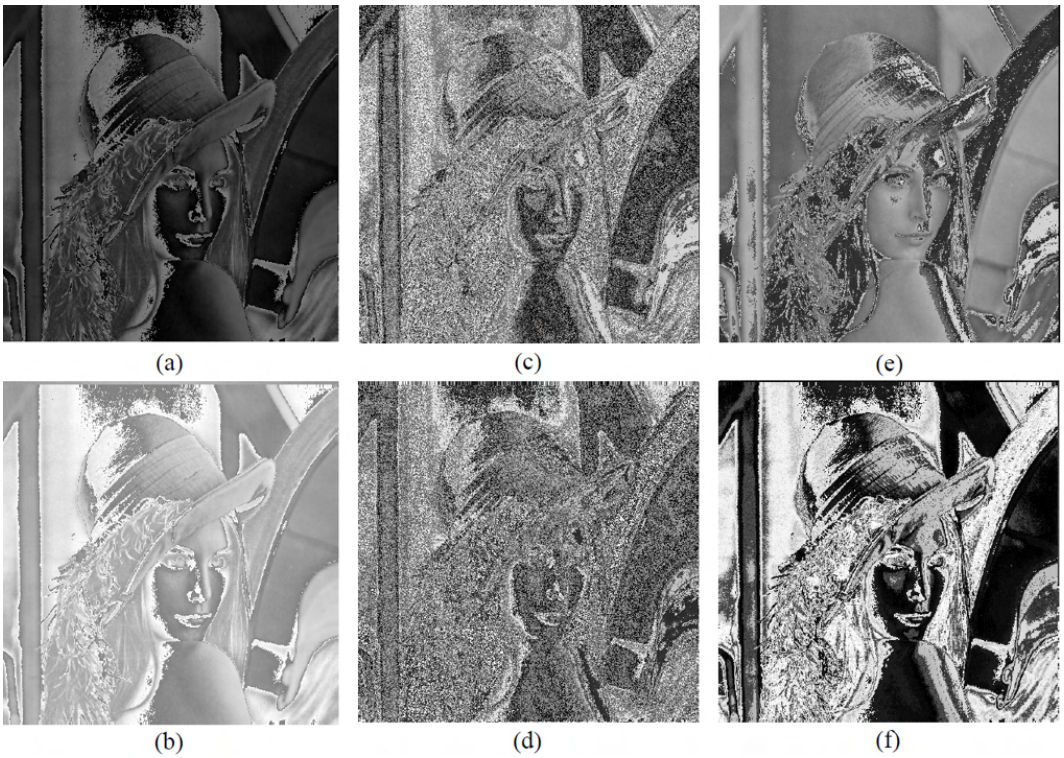


FIGURE 3.2: Eve's reconstruction of Lena in 1-bitplane encryption (a) NBC MSB0, (b) NBC MSB1, (c) OPTA MSB0, (d) OPTA MSB1, (e) $OPTB_{10^6}$ MSB0, (f) $OPTB_{10^6}$ MSB1.

TABLE 3.2: The size of the MSB plane (KB) as the measure of the encryption complexity.

Images	NBC	OPTA	$OPTB_0$	$OPTB_{10^6}$
Cameraman	6.6	32.2	7.2	7.1
Goldhill	9.0	32.8	7.3	6.8
Lena	8.2	32.4	15.4	14.9
Livingroom	11.1	32.6	9.8	12.1
Zelda	6.8	32.7	6.9	6.1

a much higher secrecy level than NBC by hiding much more detail. For instance, as seen in Figure 3.3, in the reconstruction with NBC, the smooth surface of the mirror or Lena's skin is visible, while with OPTB it is blurred. The Lena's hair

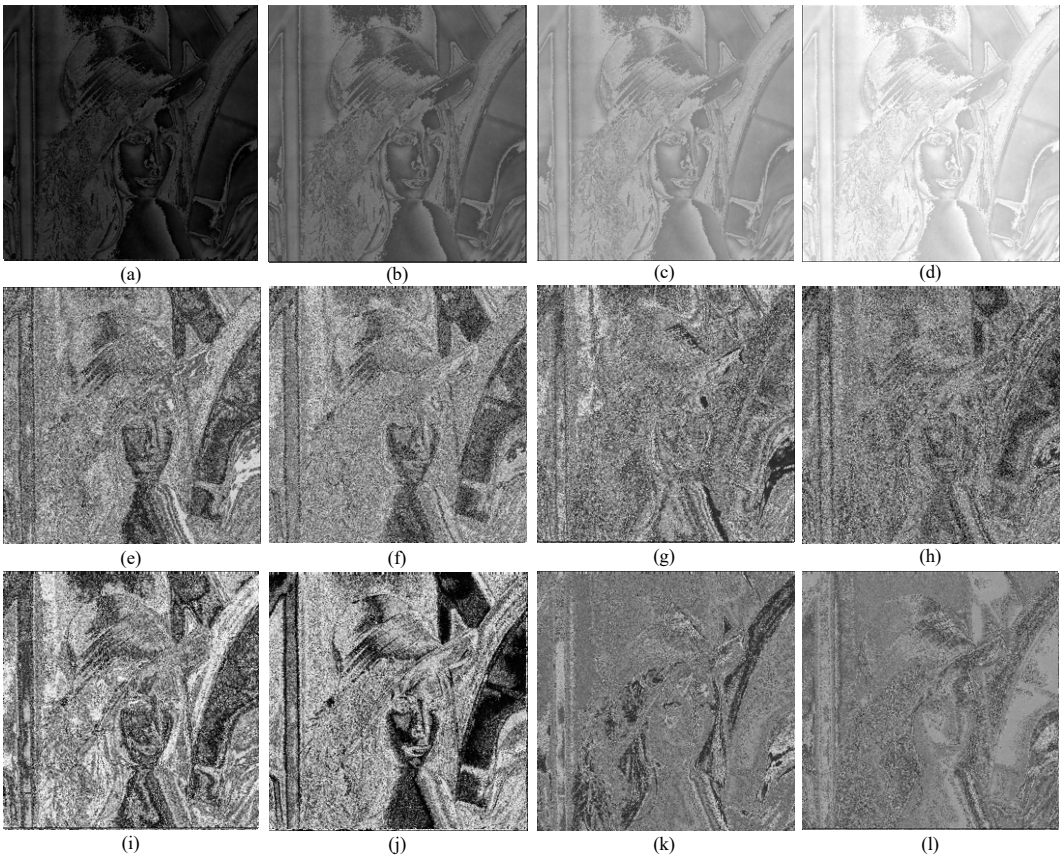


FIGURE 3.3: Eve's reconstruction of Lena under 2BPE (a) NBC 00, (b) NBC 01, (c) NBC 10, (d) NBC 11, (e) OPTA 00, (f) OPTA 01, (g) OPTA 10, (h) OPTA 11, (i) OPTB_{10^6} 00, (j) OPTB_{10^6} 01, (k) OPTB_{10^6} 10, (l) OPTB_{10^6} 11.

strands are also clearly distinguishable in the former case, but very blurry in the latter case.

Table 3.2 illustrates the size of the compressed MSB plane, as a measure of encryption complexity. The compression is performed by using the WebP standard as one of the most efficient lossless compression techniques [74]. Note that the size of the uncompressed MSB plane is 32 KB. We see from the table that the MSB plane of OPTA is not compressible, which is expected since OPTA randomizes the MSBs. We conclude that OPTA is not suitable for use in scenario B. NBC

achieves a high compression ratio of the MSB plane (between 3 : 1 and 4.8 : 1). Recall that in OPTB, the MSB 0 is assigned to intensity values that have high probability. This technique is not guaranteed to remove the correlation between adjacent bits in the MSB plane, therefore, $H(S_0)$ is no longer an accurate measure of the compressibility of the MSB plane. Thus, OPTB is not guaranteed to decrease the size of the compressed MSB plane in comparison with NBC. However, as we see from the table, in four out of the ten cases (five images with two values of λ for each image) the size of the compressed MSB plane for OPTB is smaller than for NBC, while in the remaining cases the two sizes are close, except for Lena. Since in most cases the sizes of the compressed MSB planes for NBC and OPTB are comparable, while OPTB provides higher secrecy than NBC in case of 2BPE, we conclude that OPTB is a good candidate in scenario B.

3.3 Structured BCs

The optimization of Section 3.1 must be carried out for each image individually, and thus it adds a computational overhead to the SE process. In addition, the BC has to be communicated to the decoder as well, which increases the length of the bitstream. In this section, we develop structured BCs that increase the security of SE under the replacement attack, without the computational overhead.

Our code construction exploits the fact that in natural images, the pixels that are in close proximity tend to have close intensity values as well. Therefore, in order to increase the degradation at Eve's, we aim to design the BC in such a way that after a replacement attack, pixels that are spatially close have values that are far away. We achieve this goal in two steps. As noticed in 3.2, when NBC is used

the SE method is not sufficiently secure, we first propose a small modification to NBC which has the effect of increasing the distance (i.e., the absolute difference) between the reconstructions of intensity values that differ by 1. This BC is called Alternate NBC (ANBC) and is explained in the following section. In the second step, we further modify ANBC so that to increase not only the distance between the reconstructions of consecutive integers, but also between the reconstructions of integers that differ by 2. These codes are called Spaced BCs (SBCs) and they are presented in Section 3.5. For SBCs, the aforementioned distance can be controlled, fact which provides flexibility in the choice of the degree of secrecy that we would like the SE method to achieve.

3.4 Structure of NBC and Alternate NBC

NBC is conventionally used to code the pixel intensity values of uncompressed images. NBC assigns to a value $y, 0 \leq y \leq 2^b - 1$, the codeword $(s_0, \cdots, s_{b-1})$ satisfying

$$y = \sum_{i=0}^{b-1} s_i 2^{b-1-i}.$$

Let us denote by U_0, respectively U_1, the set of integers between 0 and $2^{b-1}-1$ and the set of integers between 2^{b-1} and $2^b - 1$ (inclusive). Then, when using NBC, any integer in U_0 has the MSB 0, while any integer in U_1 has the MSB 1. Further, for any integer $y \in U_0 \cup U_1$ let us denote by $f(y)$ the fellow of y according to the

BC used. Then for NBC we have

$$f(y) = \begin{cases} y + 2^{b-1} & \text{if } y \in U_0 \\ y - 2^{b-1} & \text{if } y \in U_1 \end{cases}.$$

This implies that consecutive integers have as fellows consecutive values as well, i.e., $f(y+1) = f(y) + 1$ for all y, except for $y = 2^{b-1} - 1$. Moreover, in NBC, for most pairs of integers x, y with small $|x - y|$, both x and y have the same MSB. Thus, in the replacement attack, they are either both correctly recovered or both replaced by their fellows. But in the latter case, the absolute difference between their reconstructions still remains small since $|f(x) - f(y)| = |x - y|$. In natural images, the pixels that are spatially close mostly have close intensities. Thus, under the replacement attack, their reconstructions remain close when NBC is used, even when they are replaced incorrectly, thus rending the structure of the image understandable. This observation is corroborated by our simulation results demonstrated in Figure 3.2. For our tests,

Next we perform the correlation analysis on our test images, which measures the correlation between two adjacent horizontal, vertical, and diagonal pixels. The correlation coefficients of adjacent pixels is calculated by:

$$C = \frac{\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y))}{N\sqrt{D(x)D(y)}} \tag{3.1}$$
$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2, \;\; E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i,$$

where N is the number of pixels of the image, x_i is the intensity value of the pixel at position i, and y_i is the intensity value of the x_i's adjacent pixel. Considering

TABLE 3.3: Correlations of original images and Eve's reconstructions

Corr.	Attack	Lena	Camera	Living	Goldhill	Zelda
HCorr	Org.	0.97	0.98	0.95	0.97	0.98
	NBC	0.72	0.74	0.69	0.68	0.79
	ANBC	0.18	0.17	0.21	0.16	0.22
VCorr	Org.	0.99	0.99	0.95	0.97	0.99
	NBC	0.77	0.73	0.71	0.62	0.85
	ANBC	0.20	0.17	0.22	0.15	0.23
DCorr	Org.	0.96	0.97	0.91	0.95	0.98
	NBC	0.69	0.65	0.60	0.55	0.78
	ANBC	0.17	0.15	0.18	0.13	0.21

y_i, in turn, as horizontally, vertically, or diagonally adjacent to x_i, we obtain results in the horizontal correlation, HCorr, vertical correlation, VCorr, or diagonal correlation, DCorr, respectively. A plain image often exhibits a strong correlation between adjacent pixels, whereas these parameters should be reduced significantly for the encrypted images. Table 3.3 shows the three correlation values of original images and their reconstruction at Eve's. In the latter case, for each of HCorr, VCorr, and DCorr, we show the average of the values obtained with the MSB0 and MSB1 attacks. As shown in Table 3.3, the correlation values of original images are close to 1, as expected. The correlation values of Eve's reconstructions of images coded by NBC are reduced, but still remain relatively high.

Let us introduce a small modification of NBC to further degrade the reconstructions at Eve's. For this, we keep the same codewords as in NBC for the even integers, while for the odd integers we use the NBC codeword with the MSB flipped. We use the term Alternate NBC (ANBC) for this new binary code. Table 3.4 shows the codewords using ANBC for several pairs of fellows. With ANBC, if two pixels have the values $(y,\ y+t)$, where t is a positive odd number and y

TABLE 3.4: ANBC and its pairs of fellows

$f$ellow 1		$f$ellow 2	
value	code	value	code
0	00000000	128	10000000
1	10000001	129	00000001
2	00000010	130	10000010
3	10000011	131	00000011
...	...	...	...
127	11111111	255	01111111

and $y + t$ are both in U_0, then their reconstructions are either $(y,\ y + t + 2^{b-1})$ or $(y + 2^{b-1},\ y + t)$. If y and $y + t$ are both in U_1, then their reconstructions are either $(y,\ y + t - 2^{b-1})$ or $(y - 2^{b-1},\ y + t)$. Thus, the distance between their reconstructions under both MSB0 and MSB1 attacks is at least $2^{b-1} - t$, which is very high, especially when t is small. This simple modification has the effect of decreasing significantly the correlation between adjacent pixels, Table 3.3 shows. It also adds some noise in the image reconstructions. However, it still does not make the structure unintelligible. We can see this in Figure 3.4 and Figure 3.5, which show Eve's reconstructions under MSB0 and MSB1 attacks when the BC used is ANBC.

3.5 Spaced Binary Codes

The reason for ANBC not succeeding to provide sufficient degradation at Eve's could be the fact that it maintains a small distance between the reconstructions of intensities values y and $y + 2$ when $y, y + 2 \in U_0$ or $y, y + 2 \in U_1$. This is because y and $y + 2$ have the same MSB, thus their reconstructions are either both correct or both replaced by their fellows, in which case the distance between them does

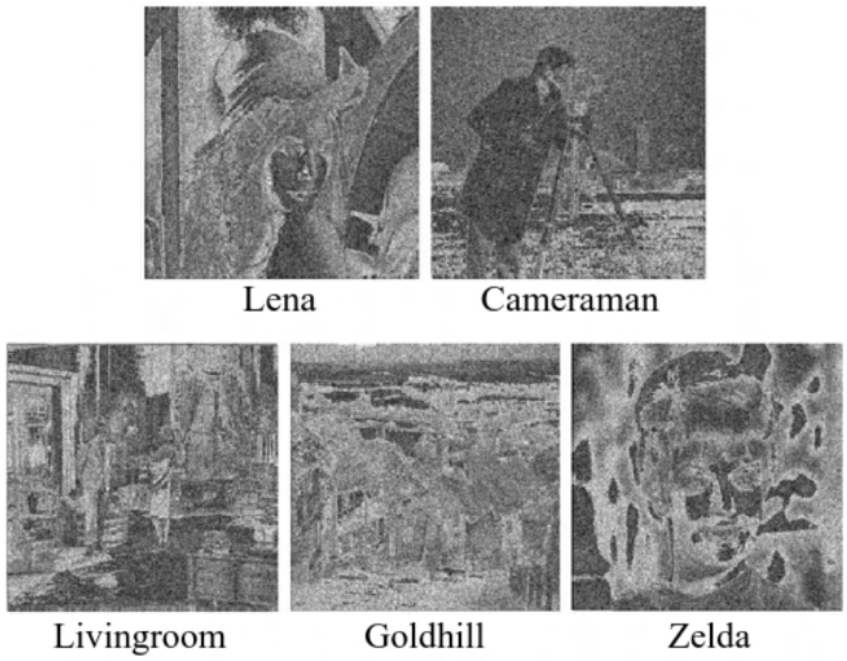


FIGURE 3.4: Eve's reconstruction of MSB encrypted images coded by ANBC with MSB0 attack.

not change since $|f(y+2) - f(y)| = 2$.

To alleviate this issue, we propose a family of BCs termed Spaced Binary Codes (SBC). In SBC, we keep the same binary representations as in ANBC for the integers in U_0, while the codewords for the integers in U_1 are changed. Each pair of fellows still contains one integer from U_0 and one integer from U_1. In order to complete the description of the code, it is sufficient to specify the fellows assigned to the integers in U_0. This assignment depends on what we call the *order* of the code, denoted by d. The even integers are assigned fellows from a list $\mathcal{L}$, whose construction depends on the order of the code and will be described shortly. More specifically, the fellow assigned to integer $2(k-1)$ is the k-th element in the list $\mathcal{L}$, denoted by ℓ_k, i.e., $f(2(k-1)) = \ell_k$, for $1 \leq k \leq 2^{b-2}$. Further, the fellow assigned to integer $2(k-1)+1$ is $\ell_k + 1$, for $1 \leq k \leq 2^{b-2}$. Note that

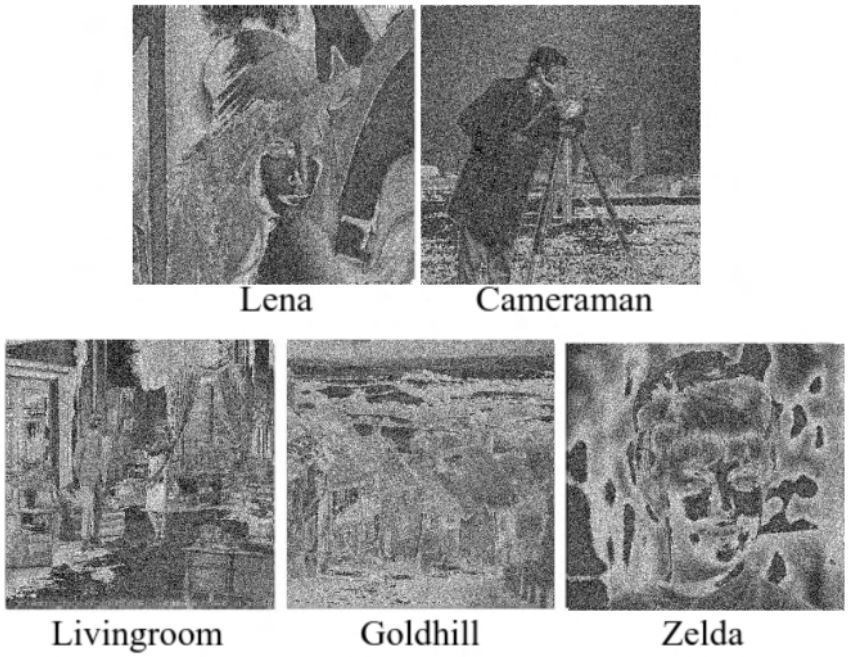


FIGURE 3.5: Eve's reconstruction of MSB encrypted images coded by ANBC with MSB1 attack.

$f(2(k-1)+1) = 1 + f(2(k-1))$.

Let us explain now the construction of the list $\mathcal{L}$. Let d be an even integer $2 \leq d \leq 2^{b-1} - 2$. For an SBC of order d, the list $\mathcal{L}$ is obtained by concatenating $d/2$ smaller lists $\mathcal{L}_0, \mathcal{L}_1, \cdots \mathcal{L}_{d/2-1}$. For each $i, 0 \leq i \leq d/2 - 1$, the list $\mathcal{L}_i$ consists of the integers $2^{b-1} + 2i, 2^{b-1} + 2i + d, \cdots, 2^{b-1} + 2i + k \times d, \cdots, 2^{b-1} + 2i + n_i \times d$, where $n_i = \left\lfloor \frac{2^{b-1}-2i-1}{d} \right\rfloor$, i.e., n_i satisfies

$$2^{b-1} + 2i + n_i \times d \leq 2^b - 1 < 2^{b-1} + 2i + (n_i + 1) \times d.$$

We impose the condition that all lists have at least two elements, which leads to $d \leq 2^{b-2}$. Also note that when $d = 2$, we have $\mathcal{L} = \mathcal{L}_0$ and the code is identical to ANBC.

The construction of SBCs is motivated by the desire to obtain a large value for $|f(y+2) - f(y)|$. Therefore, let us compute the quantity $\delta(d)$ defined as

$$\delta(d) = \min_{y+2 \in U_0 \text{ or } y \in U_1} |f(y+2) - f(y)|. \tag{3.2}$$

According to the code construction, it is sufficient to consider even integers y in the above minimization. It is easy to see that for $d = 2$ we have $\delta(2) = 2$. Assume now that $d > 2$. There are three cases to consider.

Case 1: $y, y+2$ are even integers in U_0 and their fellows are from a smaller list $\mathcal{L}_i$. Clearly, we have $|f(y+2) - f(y)| = d$.

Case 2: $y, y+2$ are even integers in U_0 and $f(y)$ is from the list $\mathcal{L}_i$, while $f(y+2)$ is from $\mathcal{L}_{i+1}$ for some i. Then

$$\begin{aligned} |f(y+2) - f(y)| &= |2^{b-1} + 2i + 2 - (2^{b-1} + 2i + n_i \times d)| \\ &= |n_i \times d - 2|. \end{aligned}$$

Case 3: y and $y+2$ are even integers in U_1. Since $d > 2$, y and $y+2$ are from consecutive small lists, i.e., $y \in \mathcal{L}_i$ and $y+2 \in \mathcal{L}_{i+1}$ for some i. Then the distance between their fellows equals $2(p+1)$, where p is the number of elements in the big list $\mathcal{L}$ situated between y and $y+2$. A moment of thought reveals that $p = n_i$. Thus, $2(p+1) = 2(n_i+1)$.

Summarizing, we obtain

$$\delta(d) = \min\{d, |n_{d/2-1} \times d - 2|, 2(n_{d/2-1} + 1)\}. \tag{3.3}$$

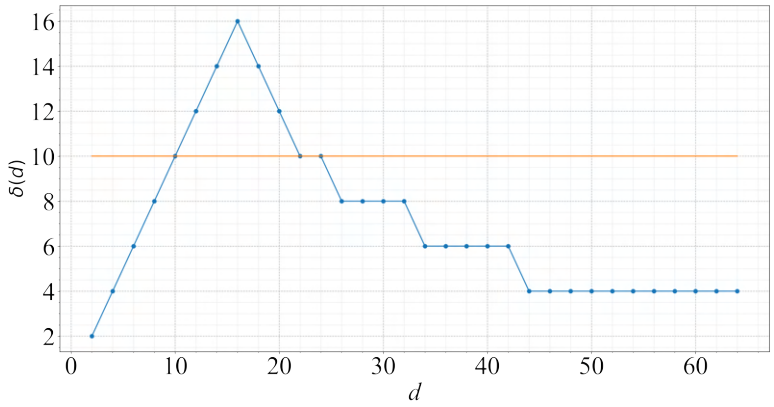


FIGURE 3.6: Variation of $\delta(d)$ versus d.

It is easy to verify that when $d > 2$, we have $|n_{d/2-1} \times d - 2| \geq 2(n_{d/2-1} + 1)$. We conclude that

$$\delta(d) = \min\{d, 2(n_{d/2-1} + 1)\}, \tag{3.4}$$

for $2 \leq d \leq 2^{b-2}$. Further, by computing the minimum in (3.4), we obtain

$$\delta(d) = \begin{cases} d & \text{if } d \leq 2^{b/2} \\ 2\left\lfloor \frac{2^{b-1}+1}{d} \right\rfloor & \text{if } d > 2^{b/2} \end{cases}, \tag{3.5}$$

where we have assumed that b is even and $b \geq 2$. According to the above relation, $\delta(d)$ increases linearly as d increases up to $2^{b/2}$, after which it decreases at a slower rate. The maximum is achieved for $d_0 = 2^{b/2}$. Figure 3.6 depicts the variation of $\delta(d)$ over d when $b = 8$.

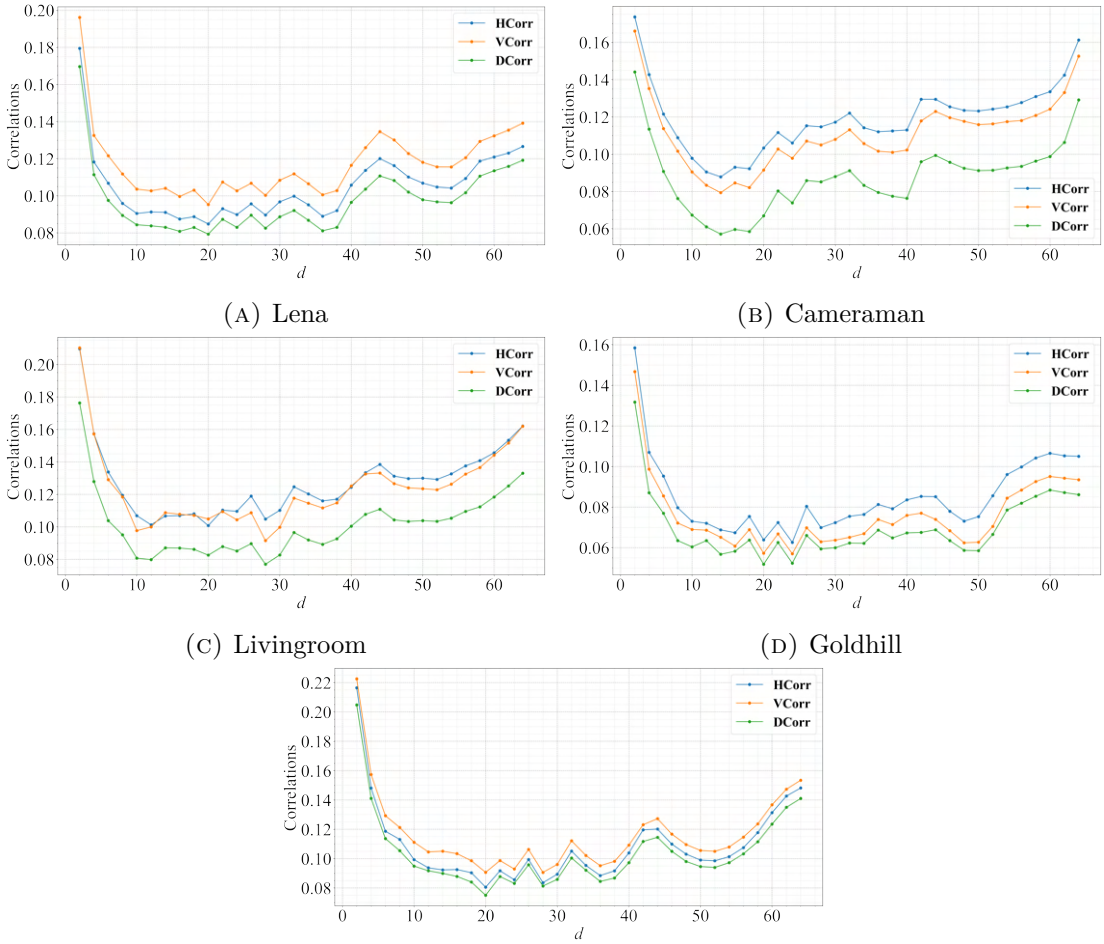
Although the highest value of $\delta(d)$ is achieved for $d = d_0$, in practice, we expect to observe similar performance for values of d that are sufficiently close to d_0. In order to test this claim, we plot in Figure 3.7, the average correlations of the

TABLE 3.5: Codewords for several pairs of fellows under SB.

$f$ellow 1		$f$ellow 2	
value	code	value	code
0	00000000	128	10000000
1	10000001	129	00000001
2	00000010	$128 + d$	10000010
3	10000011	$129 + d$	00000011
4	00000100	$128 + 2 \times d$	10000100
5	10000101	$129 + 2 \times d$	00000101
...	...	...	...

reconstructions at Eve's for the test images, for d ranging from 2 to $2^{b-2} = 64$. Recall that SBC of order $d = 2$ is actually ANBC. As we can observe in Figure 3.7, at $d = 2$ the correlations are high and match the results of Table 3.3. Further, we notice that the correlations decrease sharply as d varies from 2 up to 10. The lowest values of the correlations are around $d = d_0 = 16$ as we expected, namely for d from 10 up to 20 or 24. As we see in Figure 3.6, the aforementioned range equals the range of values of d for which $\delta(d) \geq 10$. As d increases further, the correlations tend to increase, but at a slower rate and exhibiting oscillations. This increase at a slower rate matches the behaviour of $\delta(d)$ seen in Figure 3.6. We do not have currently an explanation for the oscillations, but it can be subjected to investigate as a future work.

Figure 3.8 also demonstrates Eve's MSB0 reconstructions at $d_0 = 16$. The reconstructions under the MSB1 attack are very similar. It can be observed that SBC with a proper choice of d can induce higher noise in comparison to NBC and ANBC on Eve's reconstructions resulting in more security. Furthermore, we can observe in Figure 3.9 that various levels of degradation can be attained by varying the value of d. As expected, the degradation increases from $d = 2$ to $d = 16$, and



(A) Lena

(B) Cameraman

(C) Livingroom

(D) Goldhill

(E) Zelda

FIGURE 3.7: Correlations of Eve's reconstructions versus the order d of SBC.

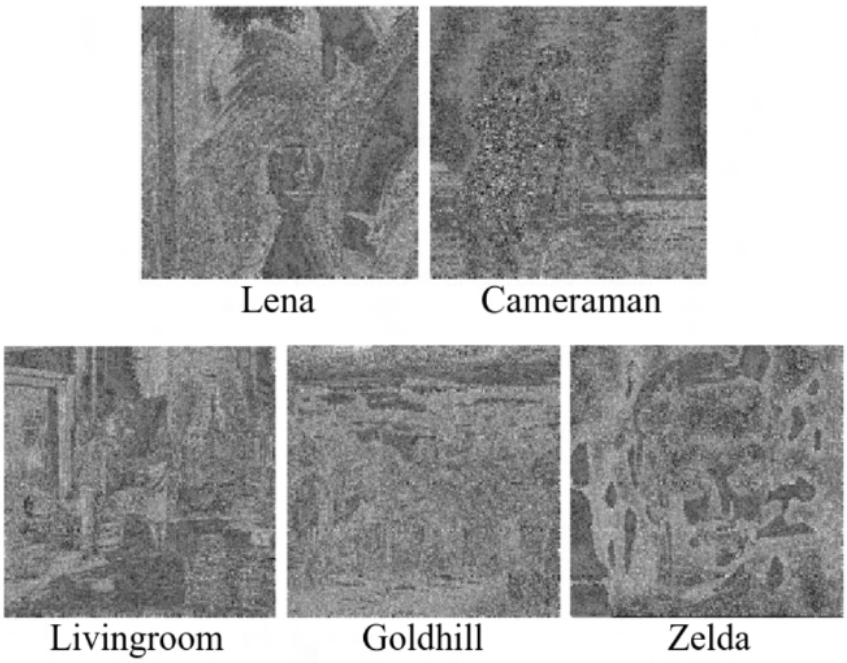


FIGURE 3.8: Eve's reconstruction of MSB encrypted images coded by SBC at $d = d_0 = 16$ with MSB0 attack.

then it reduces at higher ds such as $d = 64$. The order of SBC yields flexibility in choosing the level of secrecy. Thus, intermediate security levels can also be obtained to meet the requirements of particular applications.

3.6 Conclusion

The performance of the optimized BC was studied first for the SE of uncompressed or "lightly" compressed images. The SE was realized by encrypting one or two MSB planes within scenario A where the images are not compressed and scenario B, where only the MSB plane is compressed before being encrypted. Next, we utilize the innate property of natural images of having similar intensities for spatially close pixels. More specifically, we proposed a family of structured BCs,

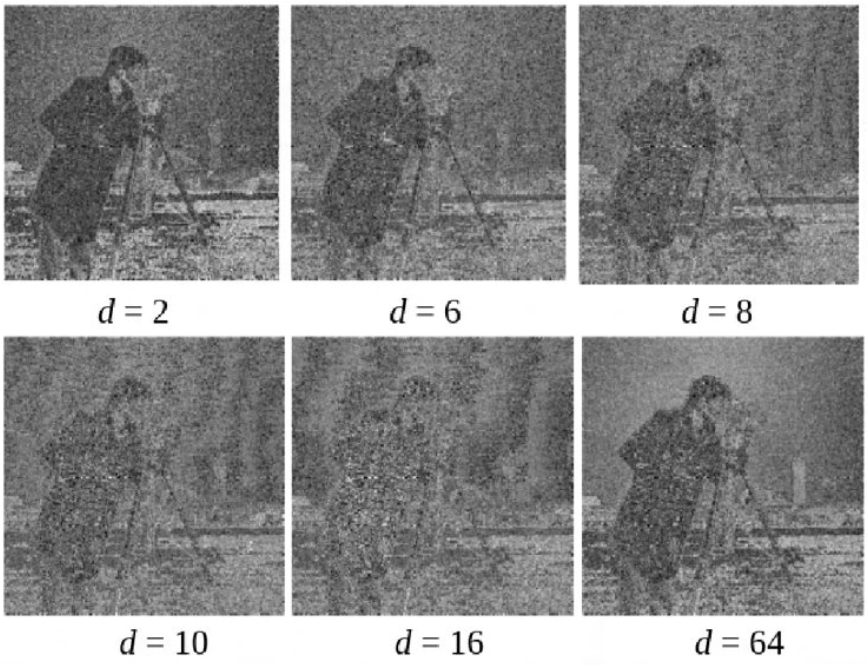


FIGURE 3.9: Eve's reconstruction of MSB encrypted Cameraman with MSB0 attack, coded by SBC at $d = 2, 6, 8, 10, 16, 64$.

called Spaced BCs, that achieves a high distance between the reconstructions of intensity values that differ by a small amount. Our theoretical and experimental analyses showed that by judiciously selecting the order d of the code, various levels of secrecy can be achieved. SBC with a proper order is able to significantly degrade Eve's reconstruction by destroying the smooth areas and blurring the edges. Consequently, without needing optimization overhead, SBCs can be developed that demonstrate the same security performance as optimized BCs.

Chapter 4

Variable-Length BC Tailored for the SE of JPEG

The JPEG compression algorithm is accredited as the most popular image lossy compression standard. The majority of the image formats in digital cameras and social network platforms as of 2017 are in JPEG due to its outstanding compression ratio, and perceptual quality [11]. A multitude of format-compliant encryption (FCE) methods have been designed for this standard to make the storage and transmission of JPEG images secure in telecommunication channels and in third-party delegates such as cloud environments [82, 10, 55, 32, 12, 49, 29, 30, 82, 40, 10, 9, 69, 35, 33, 80, 52, 47, 8, 82, 7, 18, 37, 25, 62, 36, 45, 46, 63, 48, 3, 15, 54, 65, 64, 39, 27, 68, 4, 77, 34, 53].

The most common requirements of an FCE scheme are

1. format compliance;

2. high security, i.e., resilience under various attacks such as brute force, replacement, sketch, jigsaw puzzle solver attacks, etc;

3. zero or small file size increase in comparison with the JPEG file without encryption;

4. low processing demands;

5. tunability of the perceptual quality of the encrypted file to meet the needs of different scenarios and applications.

The requirements mentioned above are generally conflicting with each other. Therefore the existing schemes have striven to achieve satisfactory trade-offs. In Section 4.1, we review these methods highlighting their strengths and weaknesses. However, obtaining optimal or very good trade-offs is difficult. Therefore, discovering new FCE techniques or enhancing old techniques is of interest.

In this chapter, we are concerned with FCE of JPEG compressed images. It has been acknowledged that for high security, the FCE method should encrypt both the DC coefficients (DCCs) and AC coefficients (ACCs) [55, 18, 45]. One way to achieve the ciphering of ACCs is by encrypting the MSBs of non-zero ACCs. However, this method is not secure enough even when combined with the full encryption of the DC coefficients. [77]. In this chapter, we address this issue too by leveraging the idea that the binary codewords used to represent the syntax elements can influence the quality of the reconstruction at Eve. We focus on the SE method that encrypts only the MSBs of non-zero quantized AC coefficients along with the full encryption of the DC differential bitstream and propose novel

binary code representations for the non-zero quantized AC coefficients that achieve high security.

We primarily utilize the level of degradation of Eve's reconstruction as a measure of security. The higher the degradation, the higher the security. We assume that Eve uses a simple replacement attack for image reconstruction, which consists of replacing all DCCs by a fixed value, while the encrypted MSBs are either all replaced by 0s or all replaced by 1s. To demonstrate the efficacy of our method, we assume that the encryption of DCCs is highly secure. Any highly secure method proposed for this purpose can be used, such as encryption of the whole bitstream obtained by differentially encoding the DCCs. We point out that the information about the modified binary code is not encrypted, and Eve has access to it, but even with this information, the replacement attack cannot be successful. Alternatively, the side information about the new codes can be transmitted to the legitimate receiver (Bob) separately. In this situation, Eve can still use the JPEG decoder and the reconstruction that she achieves will be even more degraded.

Note that in the JPEG standard, the non-zero ACCs are divided into categories, and the ACCs within each category are assigned binary codewords of the same size, i.e., the same number of bits [70]. Therefore, the binary code design proposed in Chapter 2 cannot be applied directly to all non-zero ACCs since that method addresses only the case when all binary codewords have the same length, which is not the case in JPEG. In Section 4.3, we demonstrate that by optimizing the BC within each category of JPEG codewords with the method proposed in Chapter 2, the security of the SE does not increase. As a consequence, in Section 4.4, we proceed to apply the method of Chapter 2 to optimize the BC over all categories.

Actually, the method of Chapter 2 only specifies how to pair the non-zero ACCs whose binary representations differ only in the MSB but does not assign them codewords of variable length. Therefore, we propose a method for further assigning the non-zero ACCs pairs to JPEG categories. The experimental results show that this approach can ensure confidentiality, but the cost in terms of bitrate increase is too high.

To alleviate this issue, we propose, in Section 4.5, a novel BC design technique. This is also based on solving an optimization problem to determine the pairing of non-zero ACCs, but the objective function, besides containing a term to maximize the distortion at Eve's, also includes a term that penalizes the entropy of the pairs of non-zero ACCs. By varying the parameter that controls the trade-off between the distortion at Eve's and the bitrate, different trade-offs between secrecy and compression efficiency can be achieved. In particular, confidential SE can be attained with a much smaller increase in bitrate.

In Section 4.6, we analyze the structure of the optimized BC, and the conclusions drawn lead us to propose a faster BC design technique where only a few codewords in the original JPEG BC are swapped. In Section 4.7, we perform a theoretical analysis of the distortion at Eve's, which aids us in identifying the codewords to be swapped. Section 4.8 assesses the empirical performance of the latter BC design approach showing that high levels of secrecy can be achieved with a small bitrate increase.

The security analysis continues in Section 4.9. Here we demonstrate the robustness against the EAC sketch attack. In addition, we show that our method

enables higher resilience under the statistical attack than state of the art in JPEG SE methods, while the visual distortion of the encrypted image measured with PSNR and SSIM is higher or comparable with the state of the art.

Next Section reviews the FCE methods for JPEG compressed images with their strengths and weaknesses and emphasizes the motivation and contribution of the our work.

4.1 Review of Encryption Methods for JPEG

Encryption methods for JPEG images can be divided into two approaches: a) Encryption then Compression (EtC) and b) Compression then Encryption (CtE). EtC schemes encrypt the image at the pixel level before going through the compression process, mostly by using permutations of pixels (or pixel blocks) and color transformations [12, 49, 29, 30, 82, 40]. Recent work on EtC has shown resilience to various security attacks, including the jigsaw puzzle solver attack [12]. Since EtC systems apply the compression after the encryption, the encrypted file is a JPEG-compliant file too. However, since spatial encryption destroys the local correlation and spatial orderliness of pixels, it is possible for the compression efficiency to decrease, leading to a file size increase of the compressed encrypted image [55, 35].

CtE methods, which are also named in the literature joint, or integrated compression and encryption methods, apply the encryption during one or more stages of the compression process [**97Tang** , 33, 32, 80, 55, 52, 47, 8, 7, 18, 37, 25, 62, 36, 45, 46, 63, 48, 3, 15, 54, 65, 8, 27, 68, 35, 4, 77, 53]. In Figure 4.1, we show the stages of the JPEG baseline encoder for the compression of grayscale images and

the main encryption methods that have been applied at each stage. For compressing color images, the RGB channels are initially transformed into luminance (Y) and chrominances (Cb, Cr) channels. Then each channel passes the JPEG stages shown in Figure 4.1, but with different quantization tables.

Before discussing the existing CtE methods, we briefly describe the JPEG compression algorithm. In JPEG, the source image is divided into non-overlapping 8-by-8 blocks, and the Discrete Cosine Transform (DCT) is applied to each block. The zero-frequency DCT coefficient, called the DC coefficient (DCC), is equal to the average intensity of the block. The remaining 63 DCT coefficients are the AC coefficients (ACCs). Next, the DCT coefficients are quantized using the quantization table according to the quality factor (QF), which takes values from 0 to 100. The quantized DCCs are encoded using DPCM prediction. The quantized ACCs in each block are scanned in zig-zag order, and the run of zeros (consecutive number of zero ACCs) before each non-zero ACC is computed. Each run of zeros and its next non-zero ACC are combined together, and this combination is represented by a pair of symbols: Symbol 1 (Run of Zeros, Size), called RS pair, and Symbol 2 (Amplitude). Amplitude is the value of the non-zero ACC and Size is the number of bits used to encode the amplitude, i.e., Size $= \lfloor \log_2(|Amplitude|) \rfloor + 1)$. The differences between DCCs are also represented by these two symbols, with the distinction that Symbol 1 consists only of the size information. Further, Symbol 1s are encoded with a variable-length code (VLC) specified by Huffman tables. Symbol 2s are coded by a variable-length integer (VLI) code. For this, the set of all possible values is divided into several categories, each category consisting of a specific range of amplitudes. The codewords assigned to the ACCs within

the same category have the same size, and different sizes correspond to different categories.

Next, we summarize the main encryption techniques used at different stages of the compression process.

1. **Alternate 8x8 block transforms:** Some works use alternate orthogonal transforms instead of DCT [33, 32, 80]. A transform is selected at random for each 8x8 block. These methods can preserve the file size but have weak security since the set of alternative transforms is limited [55].

2. **Encryption of the quantization table [52, 47, 8, 7]:** This method alone cannot provide high security, and it may cause format incompatibility of the encrypted file [18]. In addition, the level of distortion cannot be controlled [80].

3. **Permutation of DCT blocks [8, 36, 46, 3, 68]:** This method too produces large file size increases because it destroys the correlations of adjacent blocks' DCCs. Therefore, some works first calculate the DDC and replace it in the position of DCC of the block, and then these modified DCT blocks are permuted [63, 48, 27].

4. **Permutation of DCCs:** Inter-block global random permutations of DCCs [37, 25] can induce high visual distortion, but they lead to a significant increase in the file size. Intra-region permutations [47, 45] suppress the file size increase, but they also reduce the distortion of the reconstruction of the encrypted image.

5. **Category address mapping of DCCs [36]:** The DCCs are mapped to other values within the same size category. This approach also enlarges the

difference between consecutive DCCs, which results in a increase in the file size. In addition, the distortion of the encrypted image is not too high.

6. **Encryption of the differences of DCCs [46, 63, 48, 3, 52, 15, 7, 47, 8, 54]:** Despite file size preservation, this approach can cause an extensive overflow of the DCCs during the decoding process, which leads to format challenges. Therefore, some works developed computational-intensive processes of searching and grouping of differential DCCs to reduce the overflow of DCCs [65, 18]. Recently, Qin *et al.* [55] proposed an adaptive prediction method for the DCCs to alleviate this issue.

7. **Permutation of ACCs:**

   (a) *Intra-block permutation*: This method is one of the earliest attempts at encryption in the DCT domain [64]. This method suffers from a significant file increase because the permutation of all ACCs ruins the structure of the Run-Size (RS) pairs. In addition, it cannot withstand the non-zero coefficients count (NZC) attack proposed by *Li et al.* [37].

   (b) *Inter-block frequency-based permutation*[37]: This technique defeats the NZC sketch attack but still exhibits large file size increases.

   (c) *Intra-subsection permutation*: This method divides the 63 ACCs into four subsets of increasing frequencies within each block. Next, they are permuted within each subset to prevent significant size expansion [39]. However, this method cannot preserve the file size properly.

   (d) *Permutation of blocks of ACCs (i.e., DCT blocks without the DCCs)*: To

become resilient against NZC attacks, global permutation of DCT blocks was also proposed without permuting the DDCs to avoid file size increase [45, 55, 18].

8. **Category address mapping of ACCs**[36]: This method maps the ACCs to other values within the same size category [36]. Despite adequate file size preservation, this method cannot provide high distortion. In addition, it is vulnerable to the NZC sketch attack [37, 45].

9. **Permutation of RS pairs:** Intra-block permutation of RS pairs [45, 8, 63, 3, 27] leads to vulnerability to sketch attacks. Inter-block permutations of RS pairs[47, 48] resolves the aforementioned problem but can lead to format issues because it is possible that some blocks have more than 63 ACCs [55, 18]. To address the latter problem, the global permutation of RS pairs with proper EOB (end of block) identifiers was introduced. However, this technique leads to the increase in the file size [35].

10. **Alternate Scans of ACCs:** Ong *et al.* [47] proposed scanning the ACCs in eight different orders, and then the scan, which results in the smallest bitstream, was selected to reduce the file expansion. This method also has the format flaw.

11. **Alternate Huffman codes [4, 77]:** In these methods, multiple Huffman tables are considered, and random permutations among Huffman codeword lists are performed for coding the symbols. These methods have weak security since the number of possible permutations is limited. Moreover, the control of degradation of the encrypted image is extremely difficult [80].

12. **Encryption of Symbol 2 VLI codewords:** Encryption at this stage can provide tunability of the quality of the encrypted data, e.g., by bitplane encryption. VLI codewords encryption is format-compliant and preserves the file size properly [34]. Some works encrypt the codewords completely [52, 53, 63, 68, 3, 15, 7, 47], while others only encrypt the sign bits, i.e., the MSBs, [45, 39]. However, it is shown that encryption of only sign bits can lead to an intelligible reconstruction of the original image by using the replacement attack, i.e., by setting all encrypted MSBs to 0 or all to 1 [77]. Even encryption of the first four MSBs of the VLI codewords is still vulnerable to this attack. Additionally, if only the codewords in a subset are encrypted, even completely, the degree of security is still not sufficient [77].

It is worth pointing out that some of the encryption methods mentioned above perform modifications of the JPEG baseline coding, such as using new alternative block transforms [33, 32, 80], new adaptive prediction methods for DCCs [55], alternative scanning orders [47], multiple Huffman tables [4, 77], etc. The decoder must be informed about these modifications to be able to decode the encrypted file. These methods are referred to as "*JPEG Modified*" in Figure 4.1.

To achieve a high degree of security, more encryption methods from the above list are used. First of all, both the DCCs and ACCs must be encrypted[18, 45, 55]. If the DCCs are not cyphered, a blurred comprehensible version of the original image can still be obtained, [20, 61]. On the other hand, only encryption of the DCCs is not sufficient either since simply by replacing them with a constant value, the image reconstruction will still be intelligible [51]. In conclusion, if either the

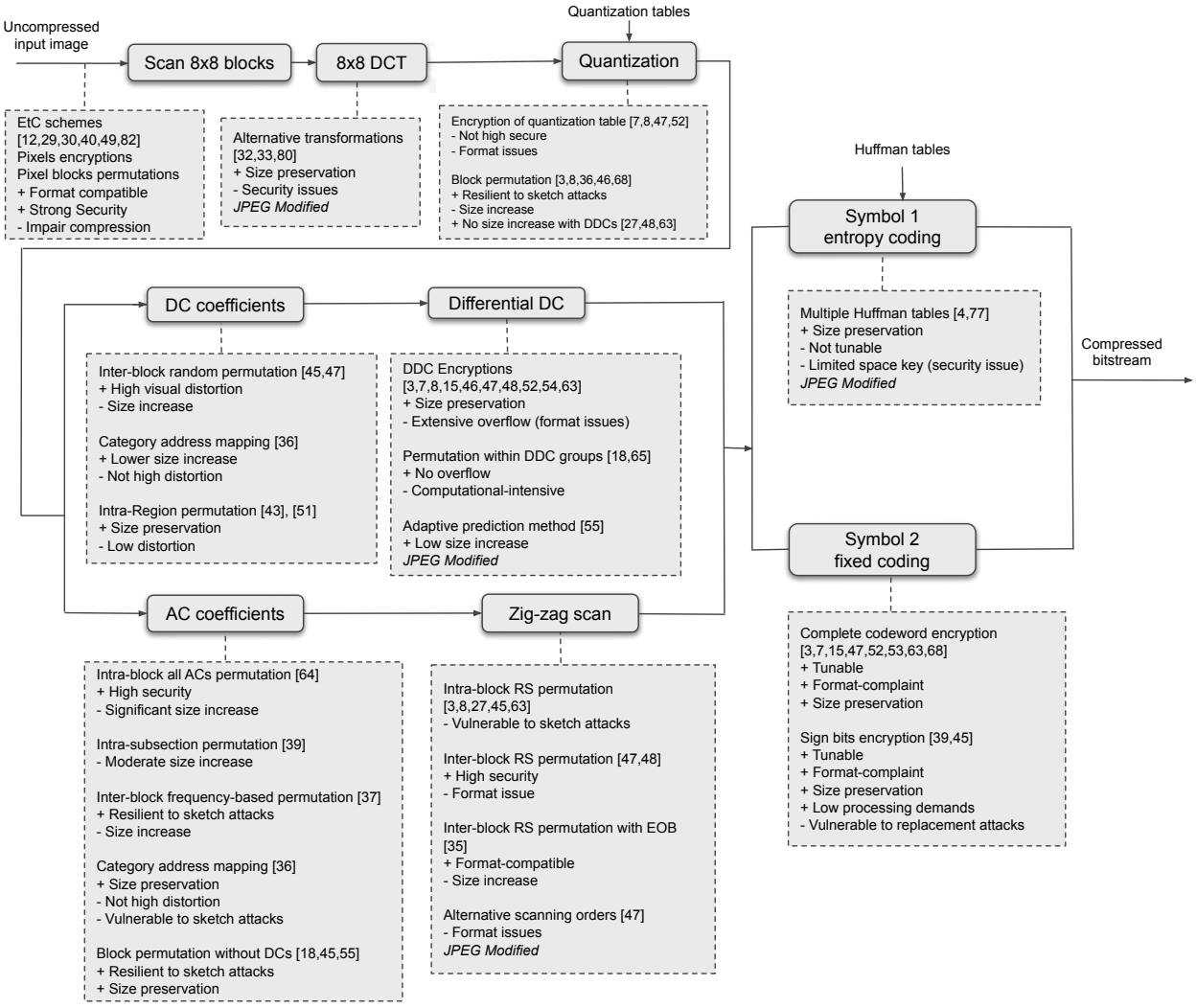


FIGURE 4.1: Encryption at different stages of JPEG compression with their weakness and competencies; JPEG modified methods are also depicted.

DCCs or ACCs are not encrypted, the method is vulnerable to a simple replacement attack. Sketch attacks have also been developed [37, 45], defeating some of the existing encryption methods. In particular, encryption methods that do not change the number of non-zero ACCs in a block are vulnerable to the NZC attack proposed by Li and Yuan [37]. Minemura *et al.* [45] have also developed three new sketch attacks, namely the DC Category Attack (DCCA), the Improved non-zero Coefficient Count (INCC) attack, and the Energy of AC coefficients (EAC) attack. In an effort to defeat the aforementioned sketch attacks while keeping the file size increase small, many recent works combine more encryption modules. In particular, Minemura *et al.* [45] use two techniques for the encryption of DCCs and three for the cyphering of ACCs. They are: 1) intra-region permutation of DCCs, 2) DCCs' prediction error scrambling, 3) ACCs sign randomization, which is equivalent to encryption of the MSB of non-zero ACCs, 4) permutation of the blocks of ACCs, and 5) intra-block permutation of RS pairs. He *et al.* [18], besides the encryption of the DCCs, use two other methods for the encryption of ACCs. The latter methods are: 1) permutation of the ACCs within each category, where the category is based on the run-zero length, and 2) permutation of the blocks of ACCs. Li and Lo [32] employ alternate block transforms, quantized blocks permutation, an additional change of the DCCs, and the encryption of the non-zero ACCs with run-length equal to 0. Finally, Qin *et al.* [55] also use multiple techniques for the encryption of both DCCs and ACCs. For the encryption of ACCs, they first perform the RS pairs permutation in the low frequency region of DCT blocks and then process the overflowed blocks to guarantee the correct number of ACCs. The resulting side information is then embedded into the ACCs through the RS pair rotation algorithm. Next, they apply intra-block RS pairs permutation

followed by permutation of all ACC blocks.

As we have seen from the above discussion, recent works combine more techniques to achieve high security of ACCs' cyphering. One of these techniques is the encryption of the MSBs. For instance, Minemura *et al.* [45] utilize it along other two ACC encryption methods. In this Chapter, we propose a modification of the VLI codes used for the non-zero ACCs to boost the efficacy of MSB encryption in defeating security attacks. By doing so, the number of methods used for the cyphering of ACCs could be decreased, leading to a decrease in the computational effort, which is one of the goals of FEC. We show that our method is able to ensure resistance against the replacement attack, EAC sketch attack, and the statistical attack when MSB encryption for ACCs' cyphering is used without being accompanied by other ACCs' cyphering techniques. To demonstrate the resilience under the statistical attack, we assume that the MSB encryption of non-zero ACCs is combined with a secure method for the DCCs' encryption, which guarantees random reconstruction of the DCC values. In addition, we empirically demonstrate that under the above assumption, the degree of security under the statistical attack is higher than that of the state of the art [45, 55] for CtE methods, while the visual degradation of the encrypted image measured using PSNR and SSIM is also higher or comparable.

Tunability of the level of distortion achieved at Eve's side is also a desirable feature of a CtE method. However, this requirement has received fewer attention [62]. Our method addresses this issue by enabling a mechanism to control the level of degradation at Eve's side when Eve uses the replacement attack to reconstruct the image.

4.2 Preliminaries

As mentioned in the previous Section, the SE method that we consider in this work encrypts the MSBs of the non-zero quantized ACCs and the bitstream obtained by differentially encoding the DCCs. For simplicity, we use in the rest of the work the acronym QAC for a non-zero quantized AC coefficient. We assume that Eve is aware of the binary code that is used. The strategy that Eve employs to reconstruct the encrypted image is the replacement attack, i.e., she either replaces all encrypted MSBs with 0 (the MSB0 attack) or replaces all encrypted MSBs with 1 (the MSB1 attack), while all DCs are replaced with a constant value.

We will say that two QACs are *fellows* if their binary representations differ only in the MSB. In this work, we will use the term *pair of QAC fellows* or simply *pair of QACs* only with this meaning. In general, we will denote a pair of QAC fellows by (qac_0, qac_1), where qac_0 is the QAC whose codeword has the MSB equal to 0, while qac_1 is the QAC whose codeword has the MSB 1. Notice that in the MSB0 attack, all qac_1's are replaced with their corresponding fellows, while the qac_0's are correctly reconstructed. In the MSB1 attack, the qac_1's are correctly recovered, while the qac_0's are replaced by their corresponding fellows. Figure 4.2 illustrates this process. Consequently, the assignment of binary codewords to QACs can affect the quality of Eve's reconstructions because it determines which QACs are paired together and, therefore, which QACs are replaced with each other in Eve's attacks.

The squared distance will be used as a distortion measure. We will denote by $D_{E,0}$ and $D_{E,1}$ the QACs' distortion at Eve's side under the MSB0 and the MSB1

attacks, respectively, and by D_E their average. Thus,

$$D_{E,k} = \sum_{(qac_0, qac_1)} (qac_1 - qac_0)^2 \Delta_Q^2 Pr(qac_{1-k}), \ k = 0, 1,$$
$$D_E = 0.5(D_{E,0} + D_{E,1}), \tag{4.1}$$

where the summations are over all pairs (qac_0, qac_1), Δ_Q denotes the step size of the quantizer and $Pr(qac)$ denotes the probability of qac.

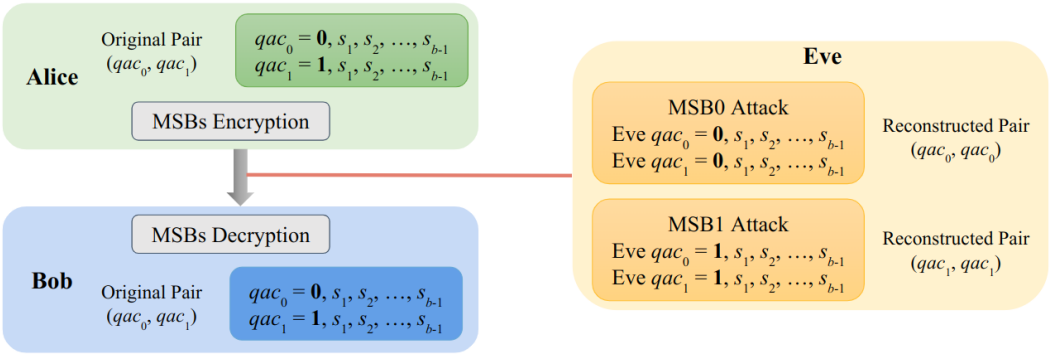


FIGURE 4.2: Original pairs of QAC fellows at Alice's side and the reconstructed pairs at Eve's side obtained with the MSB0 and MSB1 attacks.

To validate the efficacy of the proposed BCs we will perform experiments on five gray-level JPEG-compressed images of size 512×512, with 8-bit pixel values, which are shown in Figure 4.3 . Their JPEG quality factor is $QF = 100$, which means the highest quality in the compression, i.e., the quantization of ACCs is performed by rounding to the closest integer.

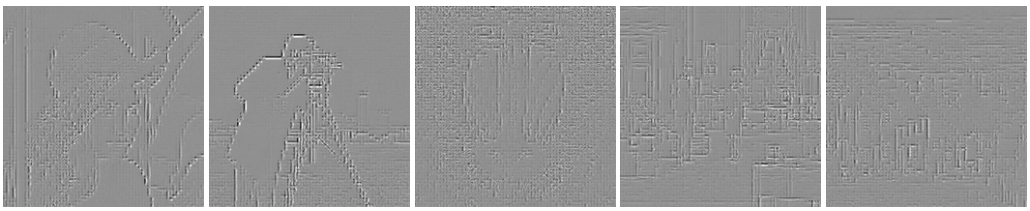
In the replacement attack, all DCCs are replaced by the value 128, while the MSBs of all QACs are replaced either by 0 (MSB0 attack) or by 1 (MSB1 attack). Next, the inverse discrete cosine transform (IDCT) is applied. Note that, since some of the reconstructed DCT coefficients are different from the original ones,

the pixel intensity values obtained after applying IDCT are not guaranteed to be within the valid range for 8-bit values, namely from 0 to 255. To resolve this issue we replace the negative intensities with 0 and the intensities larger than 255 with 255.

In the rest of the chapter, we will use the name OrigBC for the Original BC used for the QACs in the JPEG standard. In Figure 4.4 and Figure 4.5, we show Eve's reconstructions for test images when the SE method is applied with OrigBC and the MSB0 attack and MSB1 attacks are utilized. As we can see, in all cases, the contours of objects are visible; therefore the SE method is not sufficiently secure.

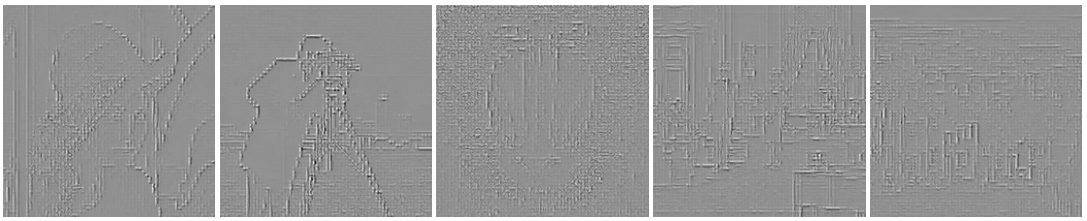
(A) (B) (C) (D) (E)

FIGURE 4.3: Test images: a) Lena, b) Cameraman, c) Mandrill, d) Livingroom, e) Goldhill.



(A) (B) (C) (D) (E)

FIGURE 4.4: Eve's reconstructions using the MSB0 replacement attack after SE with OrigBC, a) Lena, b) Cameraman, c) Mandrill, d) Livingroom, e) Goldhill.



(A) Lena (B) Cameraman (C) Mandrill (D) Livingroom (E) Goldhill

FIGURE 4.5: Eve's reconstructions using the MSB1 replacement attack, after SE with OrigBC.

4.3 SE-Optimized Coding within Each Category

Since the amplitudes of the QACs within each category are encoded with a fixed-length BC, the BC within each category can be optimized using the method developed in Chapter 2 for scenario A. More specifically, in the aforementioned scenario, the binary codewords assigned to syntax elements have the same length. The MSBs of the syntax elements are encrypted and the optimization problem is to find the binary codewords assignment that maximizes the average distortion of the reconstruction of the syntax elements using the replacement attack. We can apply the method proposed in Chapter 2 for the aforementioned problem to design the code within each category such that to maximize the distortion D_E at Eve's side. But the optimization within each category cannot increase the secrecy noticeably. As QACs with small amplitudes have high probabilities, they are assigned small-size codewords in OrigBC. High QACs are less frequent, therefore they are assigned large-size codewords. More specifically, for each $s \geq 1$, the 2^s binary codewords of size s are allocated to the integers in the union of intervals $[\alpha(s), \beta(s)] \cup [-\beta(s), -\alpha(s)]$, where $\alpha(s) = 2^{s-1}$ and $\beta(s) = 2^s - 1$. We will denote by $\mathcal{C}(s)$ the set of these integers and we will refer to it as the category of size s.

Let $\mathcal{C}_+(s)$ (respectively, $\mathcal{C}_-(s)$) denote the set of positive (respectively, negative) values in $\mathcal{C}(s)$. Additionally, denote $\delta(s) = 2^s - 1 + 2^{s-1}$. Then for any pair (qac_0, qac_1) of values in $\mathcal{C}(s)$, the following relations hold

$$qac_0 < 0, \;\; qac_1 = qac_0 + \delta(s) > 0. \tag{4.2}$$

By plugging the equation (4.2) in (4.1), we obtain the distortion at Eve's side for OrigBC,

$$D_{E,orig} = 0.5 \sum_{s=1}^{S} \delta(s)^2 \Delta_Q^2 Pr_s,$$

where S is the number of categories and Pr_s denotes the sum of the probabilities of QACs in category $\mathcal{C}(s)$. We expect the probability Pr_s for small s to be high and to gradually decrease as s becomes sufficiently high. This is confirmed in Table 4.1 for the images in our study. We can observe that for $s \geq 2$, Pr_s decreases as s increases, exhibiting an exponentially decreasing trend. This implies that the terms corresponding to lower sizes s have a determining role in the distortion. On the other hand, when s is small, the value $\delta(s)$ is small as well, therefore the distortion at Eve's side does not increase sufficiently.

TABLE 4.1: The sum of QACs' probabilities in each category $\mathcal{C}(s)$.

	Pr_1	Pr_2	Pr_3	Pr_4	Pr_5	Pr_6	Pr_7	Pr_8	Pr_9	Pr_{10}
Cameraman	0.2177	0.1488	0.0792	0.0461	0.0289	0.0159	0.0076	0.0031	0.0012	0.0001
Lena	0.2347	0.3045	0.1895	0.0749	0.0351	0.0179	0.0079	0.0030	0.00054	$3.9e^{-6}$
Mandrill	0.1946	0.1612	0.1344	0.1141	0.0855	0.0557	0.0244	0.0044	0.0002	0
Livingroom	0.1064	0.2841	0.2448	0.1227	0.0643	0.0315	0.0133	0.0042	0.0005	$7.8e^{-6}$
Goldhill	0.1740	0.2551	0.2362	0.1402	0.0636	0.0255	0.0089	0.0023	0.0002	0

Let us derive now an upper bound for the distortion D_E when the code within

each category is allowed to vary. In this case, relation (4.2) is not necessarily satisfied, however, we can derive a simple upper bound for $qac_1 - qac_0$ in $\mathcal{C}(s)$, namely

$$|qac_1 - qac_0| \leq \max_{y \in \mathcal{C}(s)} y - \min_{y \in \mathcal{C}(s)} y = 2^{s+1} - 2 < \frac{4}{3}\delta(s).$$

Plugging this in (4.1) leads to

$$D_E < 0.5 \sum_{s=1}^{S} \frac{16}{9} \delta(s)^2 \Delta_Q^2 P_s = \frac{16}{9} D_{E,orig},$$

which implies that optimizing the BC within each category cannot increase the distortion at Eve's side by more than 3 dB in comparison to OrigBC, which is not sufficient to guarantee security. Note that this upper limit of 3 dB is an overestimate. In practice the increase is much smaller, in particular, less than 0.1 dB for our test images as shown in Table 4.2.

TABLE 4.2: The distortion at Eve's achieved with OrigBC ($D_{E,orig}$), and with the BC optimized within each category ($D_{E,copt}$).

	Cameraman	Lena	Mandrill	Livingroom	Goldhill
$D_{E,orig}$ (dB)	32.85	31.37	33.84	32.97	31.07
$D_{E,copt}$ (dB)	32.93	31.38	33.85	32.98	31.08

4.4 SE-Optimized Coding over All Categories

The original JPEG code, OrigBC, does not yield strong security because the pairs of QAC fellows are in the same category, and either their absolute difference is low, or, if their difference is large, their probability is low. As we have seen in the

previous section, if we optimize the BC within each category the results are similar. Thus, if we desire a more degraded reconstruction at Eve's side, we must design a binary code that allows the pairing of QACs from different categories. To this end, we could apply the optimization method proposed in Chapter 2 to the whole set of QACs, instead of applying it to each category separately. The optimization has as input a set of reconstruction values $\mathcal{C}$ together with the probability of each value, and it outputs a pairing of these values such that the distortion D_E is maximized. Once the pairing is specified, the assignment of binary codewords to reconstruction values is straightforward, since all codewords have the same number of bits $b = \lceil \log_2 |\mathcal{C}| \rceil$.

In order to apply the method of Chapter 2 to this case, we construct the input set $\mathcal{C}$ as the set of all QACs with non-zero probabilities. Let $y_1 < y_2 < \cdots < y_M$ be the integers in $\mathcal{C}$, where M denotes its size. After obtaining the pairing output by the optimization algorithm, we have to assign binary codewords to QACs such that the pairing to be preserved. At this point the similarity with the problem in Chapter 2 ends. If it were to use a fixed-length code, i.e. to assign a $\lceil \log_2 M \rceil$-bit codeword to each QAC, the compression performance would decline drastically since the rate needed to encode all QACs would become at least three times larger than with OrigBC. This can be seen by inspecting the value ΔR_{fix} in Table 4.4, where ΔR_{fix} denotes the rate increase using the above procedure as a percentage of the bitrate needed to encode the QACs with OrigBC.

To avoid this escalation in bitrate we have to use a variable-length code. In order to minimize the modification brought to the JPEG algorithm, we will use the same codewords that are used in JPEG. This means that we still divide the QACs

into categories and assign a fixed-length code within each category, but the new BC implies a new assignment of QACs to categories with the constraint that any two fellows are in the same category. Let $\mathcal{C}'(s)$ denote the new category of size s. Thus, the maximum number of elements is 2^s and each QAC in $\mathcal{C}'(s)$ will be assigned an s-bit codeword. Clearly, in order to decrease the bitrate, the pairs (qac_0, qac_1) that have a high probability (i.e., for which $Pr(qac_0) + Pr(qac_1)$ is high) have to be assigned to a smaller size category, while pairs with a small probability have to be assigned to a larger size category. This observation leads to the following method. First, order the pairs in decreasing order of their probabilities. Next, assign the pairs in this order to categories by filling first $\mathcal{C}'(1)$, then $\mathcal{C}'(2)$, next $\mathcal{C}'(3)$, and so on. In other words, fill the lower size category before moving to the next bigger size category. It can be easily seen that this technique guarantees the smallest bitrate for the given pairing. Therefore, we will refer to it as "Efficient Codeword Assignment" (EffCA). Further, the codeword assignment within each pair is performed as follows. The MSB 0 is assigned to the smaller fellow in the odd-numbered pairs and to the larger fellow in the even-numbered pairs, where we consider the pair numbering in the sorted list. This is in order to guarantee that overall the probability of the MSB 0 is approximately the same as the probability of the MSB 1. As an example of EffCA, Table 4.3 shows the seven highest probability pairs obtained after solving (4.4) for Lena, their assignment to categories, and their codewords.

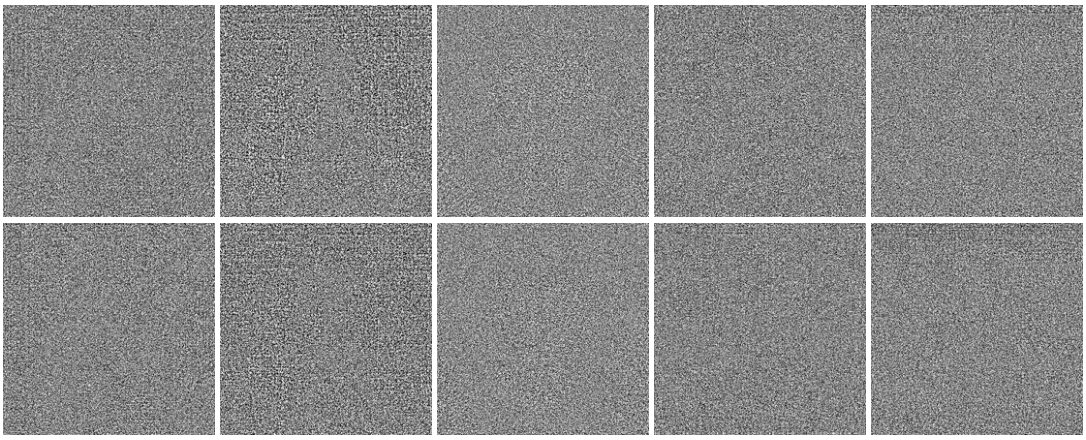
Finally, when assessing the rate increase, we need to account for the side information (SI) that is needed in order to communicate to the decoder the new codeword assignment. For this, we fix an ordering of the binary codewords and

TABLE 4.3: The seven highest probability optimized QAC pairs and their binary codewords with EffCA for Lena.

Pair	Pair Probability	Codewords	$\mathcal{C}'(s)$
$(-1,+634)$	0.1181	$(0,1)$	$\mathcal{C}'(1)$
$(+1,+508)$	0.1667	$(10,00)$	$\mathcal{C}'(2)$
$(-2,+506)$	0.0907	$(01,11)$	
$(+2,+497)$	0.0903	$(100,000)$	$\mathcal{C}'(3)$
$(-3,+472)$	0.0257	$(001,101)$	
$(+3,-465)$	0.0253	$(110,010)$	
$(+4,+459)$	0.0177	$(011,111)$	
$\cdots$	$\cdots$	$\cdots$	$\cdots$

form the SI stream by concatenating the QACs assigned to the codewords in the aforementioned ordering, where for each QAC we use an $(S+1)$-binary string. Note that it is sufficient to use $S+1$ bits for each QAC since the total set of non-zero QACs contains at most 2^{S+1} values. We append the $(S+1)$-bit binary representation of M at the beginning of the stream. We conclude that the length of the SI stream is $(M+1)(S+1)$.

We will use the acronym OptBC to refer to the BC obtained with the method described in this section. Table 4.4 shows the rate increase using OptBC, denoted by ΔR_{eff}, as a percentage of the bitrate needed to encode the QACs with OrigBC for all test images. The table also includes the value of M, the value of ΔR_{fix}, the number of bits used to encode all QACs in OrigBC, denoted by L_{orig}, and the fraction of the SI out of L_{orig} needed to communicate the new BC. In addition, the table includes the distortion at Eve's, D_E. We observe that the values of D_E in Table 4.4 are extremely high, which leads to the conclusion that the BC optimized over all categories can ensure a high degree of security. This claim is also supported by the fact that the reconstructions at Eve's, which are depicted in Figure 4.6 are incomprehensible.



(A) Lena (B) Cameraman (C) Mandrill (D) Livingroom (E) Goldhill

FIGURE 4.6: Eve's reconstructions of the test images using the replacement attack, after SE with OptBC. Top: MSB0 attack; bottom: MSB1 attack.

TABLE 4.4: Bitrate and distortion results for the BC optimized over all categories with the method of Chapter 2.

	M	L_{orig}	$\Delta R_{fix}(\%)$	$\Delta R_{eff}(\%)$	SI(%)	$D_E(dB)$
Cameraman	731	327077	332.92	28.30	2.46	52.22
Lena	592	536526	317.55	28.88	1.21	50.82
Mandrill	516	605284	230.17	24.29	0.94	47.37
Livingroom	600	656142	242.93	19.60	1.01	50.70
Goldhill	511	646888	225.22	27.02	0.87	48.94

However, the rate increase, even with EffCA, is quite high ranging between 19.6% and 28.88%. It is expected for the rate to increase when performing changes in the composition of the categories, however, we would like to have a way to control this increase. In the next section, we propose a method for optimal BC design that incorporates such a control mechanism.

4.5 Entropy-constrained Optimized BC

One way to regulate the rate increase is by changing the formulation of the optimization problem such that to take into account the effect that the pairing has

on the rate. Ideally, the new problem should be formulated as the problem of maximizing D_E with a constraint on the rate. However, since constrained discrete optimization problems are generally difficult to solve, we simplify the problem. The first step toward simplification is to eliminate the constraint and focus on maximizing the Lagrangian, instead. The second step is to replace the true rate with the entropy of the probability distribution of the pairs of QAC fellows, motivated by the intuition that a pairing with small entropy should lead to a small rate when the EffCA is used. In conclusion, we formulate the optimization problem as the problem of finding the pairing that maximizes the following cost

$$D_E - \lambda H_P, \tag{4.3}$$

where H_P denotes the entropy of the probability distribution of the pairs of QAC fellows and λ is a positive constant. Note that this problem formulation is different from the formulation introduced in Chapter 2 for scenario B. More specifically, in Chapter 2, the objective to maximize is $D_E + \lambda P_0$, where P_0 is the probability of the MSB 0, and it is motivated by the desire to increase the distortion at Eve's while limiting the entropy of the MSB plane.

For every integers i and j, $0 \leq i < j \leq M-1$, we use the binary variable $x_{i,j}$ to indicate the pairing status of y_i and y_j. In other words, $x_{i,j} = 1$ if y_i and y_j form a pair of fellows and $x_{i,j} = 0$ otherwise. Further, we denote $H_{i,j} = -(Pr(y_i) + Pr(y_j)) \log_2(Pr(y_i) + Pr(y_j))$. Then, the total entropy H_P can be

obtained as follows

$$H_P = \sum_{i=0}^{M-2} \sum_{j=i+1}^{M-1} H_{i,j} x_{i,j}.$$

As observed in Chapter 2, D_E can be written as

$$D_E = \sum_{i=0}^{M-2} \sum_{j=i+1}^{M-1} D_{i,j} x_{i,j},$$

where $D_{i,j} = 1/2(y_i - y_j)^2(Pr(y_i) + Pr(y_j))$. By denoting $\omega_{i,j} = D_{i,j} - \lambda H_{i,j}$, we further obtain that

$$D_E - \lambda H_P = \sum_{i=0}^{M-2} \sum_{j=i+1}^{M-1} \omega_{i,j} x_{i,j}.$$

We conclude that the optimization problem (4.3) can be formulated as

$$\max_{(x_{i,j})_{i,j}} \sum_{i=0}^{M-2} \sum_{j=i+1}^{M-1} \omega_{i,j} x_{i,j} \tag{4.4}$$

$$\text{subject to } x_{i,j} \in \{0,1\},\ 0 \leq i < j \leq M-1$$

$$\sum_{j=i+1}^{M-1} x_{i,j} + \sum_{k=0}^{i-1} x_{k,i} = 1,\ 0 \leq i \leq M-1,$$

where the equality constraints have the role of ensuring that each QAC has one and only one fellow. Problem (4.4) is a weighted non-bipartite graph matching problem and can be solved in $O(M^3)$ time [50]. After solving the problem (4.4), which optimizes the pairing of QACs, EffCA is further used to assign the binary codewords. We will use the acronym ECOptBC for the BC designed using this technique.

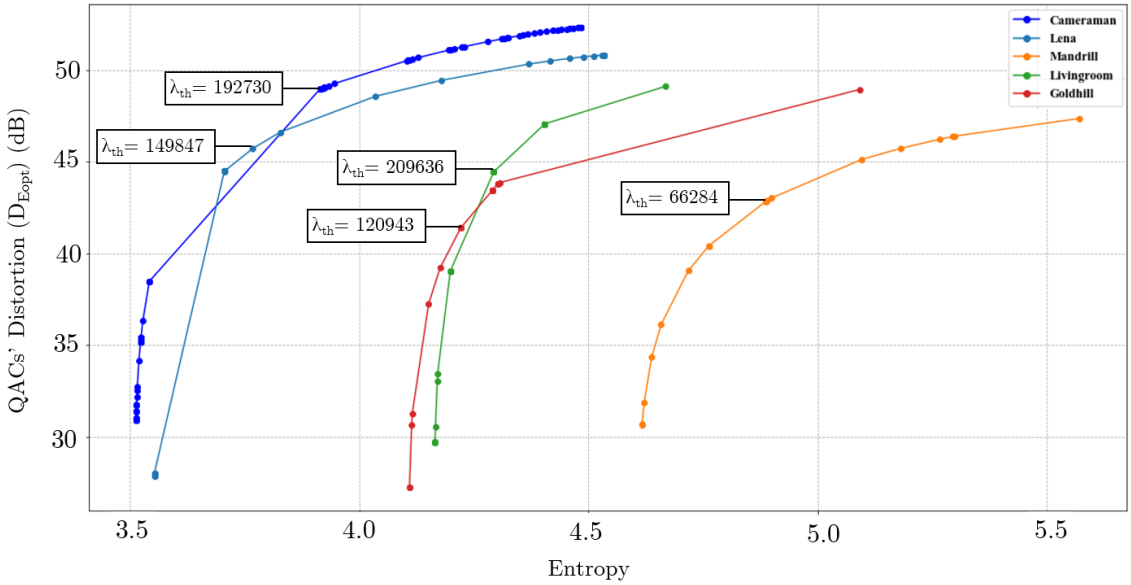


FIGURE 4.7: QACs' distortion at Eve's (D_{Eopt}) versus the entropy H_P of the QAC pairs optimized by solving (4.3) for various values of λ.

In order to test the practical performance of ECOptBC, we have solved (4.4) using the PuLP library in Python. To reduce the running time of the solution algorithm when M is high, we perform the optimization on a smaller subset of $\mathcal{C}$, denoted by $\mathcal{C}_{sub}$. We use $M_{sub} = |\mathcal{C}_{sub}| = 256$ and we include in $\mathcal{C}_{sub}$ the smallest $M_{sub}/4$ values and the largest $M_{sub}/4$ values in $\mathcal{C}$, along with the $M_{sub}/2$ values around 0 (half smaller than 0, the other half larger than 0). In this way, $\mathcal{C}_{sub}$ retains the QACs with the highest probabilities (i.e., those around 0), and the QACs that are far away from those (in order to allow pairings with high difference between the fellows). For the remaining QACs (which are not in $\mathcal{C}_{sub}$ and hence are not paired by (4.4)), we use the following pairing method. Any QAC whose fellow in OrigBC is not in $\mathcal{C}_{sub}$ either, remains paired with its original fellow. The rest of QACs are sorted in ascending order. Next, the smallest negative QAC is paired with the smallest positive QAC, the next negative QAC is paired with the

next positive QAC, and so on. Since the number of negative and positive QACs is not necessarily the same, a few unpaired QACs still remain, either all positive or all negative. The first half of this set is further paired with the second half.

We have performed experiments for various values of λ, starting with $\lambda = 0$, which corresponds to OptBC. The results are depicted in Figure 4.7 for the test images which plots D_E versus H_P for all images in our study. For each image, the point (H_P, D_E) obtained for $\lambda = 0$ is the rightmost point in the plot, i.e., achieving the highest distortion D_E, but also the highest entropy H_P. As λ progressively increases away from 0, the corresponding point (H_P, D_E) moves left and down, i.e., both D_E and H_P decrease. This is expected since when $\lambda > 0$, a penalty is imposed on the entropy H_P, thus H_P decreases. This penalty acts as a constraint on the optimization space, therefore the distortion D_E also decreases. The higher λ, the higher the penalty on H_P, thus the higher the decrease of both H_P and D_E is. As we have already discussed in Section 4.4, when $\lambda = 0$ the secrecy is very high, but there is a high loss in compression efficiency as well. However, we observe that as λ increases up to some image-specific threshold value, which we denote by λ_{th}, the entropy H_P decreases quite sharply, while the loss in secrecy is very modest since D_E decreases only slightly.

Since instead of the entropy, H_P, we are actually interested in the coding bitrate, while the degradation in the quality of the image reconstruction is more easily interpreted when measured by the PSNR, we plot in Figure 4.13 the average of the PSNRs obtained with the MSB0 and MSB1 attacks versus the percentage of the rate increase, for all test images. First, we notice that the rate indeed increases as the entropy H_P increases. Second, we notice that the threshold effect of λ_{th} is

more visible in these plots. Indeed, as λ increases from 0 to λ_{th}, the rate decreases drastically, while the increase in PSNR is indistinguishable in most of the cases. On the other hand, as λ increases from λ_{th} to $\lambda_{th}+1$, the increase in PSNR becomes noticeable. In certain cases, the jump in PSNR from λ_{th} to $\lambda_{th}+1$ is quite dramatic, as is the case of Cameraman and Lena.

The values of the PSNR and SSIM for both MSB0 and MSB1 attacks, achieved at λ_{th} are shown in Table 4.5 for all test images. The value of the rate increase (as a percentage of the rate of QACs in OrigBC) is also included in the table. The objective metrics indicate that a high level of secrecy is achieved when employing λ_{th}, while the increase in rate is significantly smaller than the optimization without constraint on entropy. Interestingly, in the case of Livingroom, the rate actually decreases. The reason for this phenomenon is the fact that Livingroom is the image from our test set for which the probability of a QAC is not necessarily larger than the probability of any QAC in the next size category (as shown in Figure 4.8), which makes OrigBC inefficient. This inefficiency is overcome in ECOptBC since it uses EffCA, which assigns pairs to categories in decreasing order of their probabilities.

For the subjective assessment of secrecy, we show in Figure 4.9 the reconstructions at Eve's under MSB0 and MSB1 attacks for λ_{th} and $\lambda_{th}+1$ for Lena. The corresponding reconstructions for the other test images are presented in Figure A3.0 in Appendix C. We observe that in all cases, the reconstructions corresponding to λ_{th} are incomprehensible, while as λ moves above λ_{th}, the objects' contours are becoming visible. We conclude that the point corresponding to λ_{th} could be considered the optimal trade-off point between secrecy and rate, since it yields the

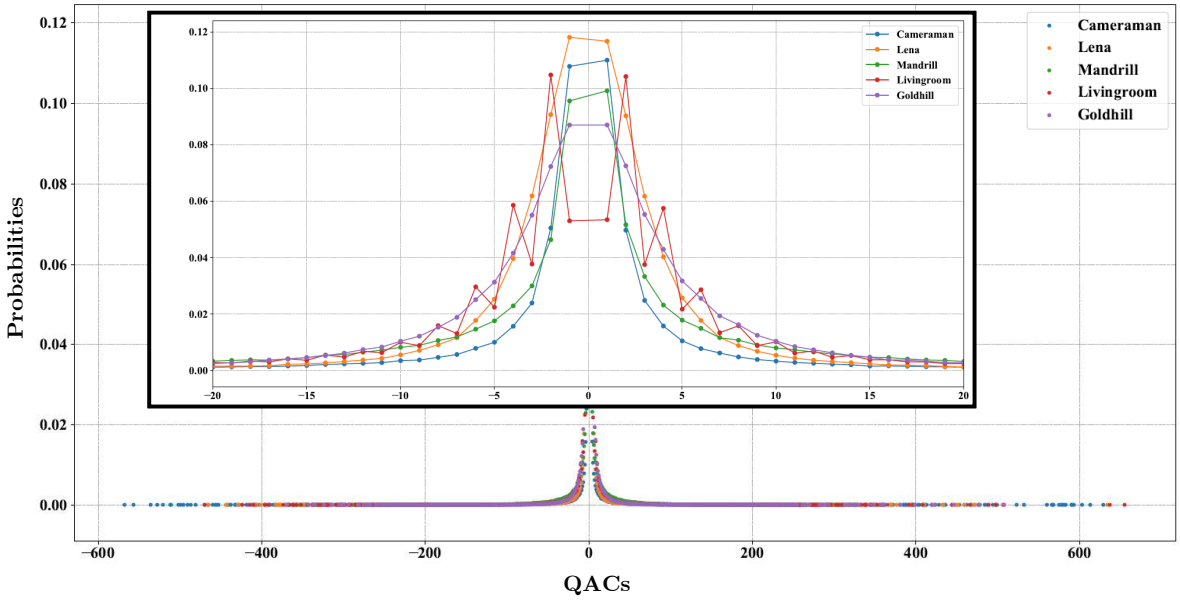


FIGURE 4.8: Histogram of the QACs for the test images.

smallest possible bitrate, while keeping the secrecy level sufficiently high. Since λ_{th} is specific to each image, in order to find the optimal trade-off point the algorithm to solve the problem (4.3) has to be augmented with a bisection search over λ. The search could stop when the highest PSNR value is still smaller than some upper bound has been achieved.

We conclude that for certain images ECOptBC can provide high security with only a modest increase in bitrate. However, performing the optimization at the encoder increases the encoding time. Another potential issue is that in some cases (such as the case of Cameraman and Lena), there is a big gap between the solutions corresponding to λ_{th} and $\lambda_{th}+1$. The reason is that the formulation of the problem (4.4) can only find the extreme points on the upper boundary of the convex hull of the set of possible (H_P, D_E) pairs. The pairs corresponding to λ_{th} and $\lambda_{th}+1$ are the extremities of a long convex hull edge. If intermediate trade-off points are

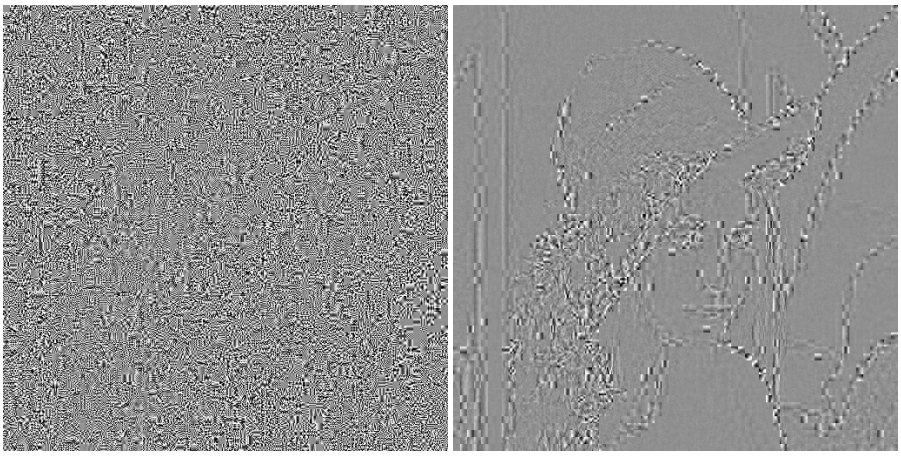
TABLE 4.5: Objective security metrics and the rate increase (ΔR) in percentage for ECOptBC with λ_{th}.

	Cameraman		Lena		Mandrill		Livingroom		Goldhill	
$\Delta R(\%)$	13.62		5.61		8.37		−2.76		2.42	
	MSB0	MSB1	MSB0	MSB1	MSB0	MSB1	MSB0	MSB1	MSB0	MSB1
PSNR	6.44	6.41	7.95	6.86	8.48	8.32	7.86	7.47	8.45	7.84
SSIM	0.0010	0.0087	−0.0031	0.0003	0.0100	−0.0010	0.0074	−0.0086	0.0074	−0.0007

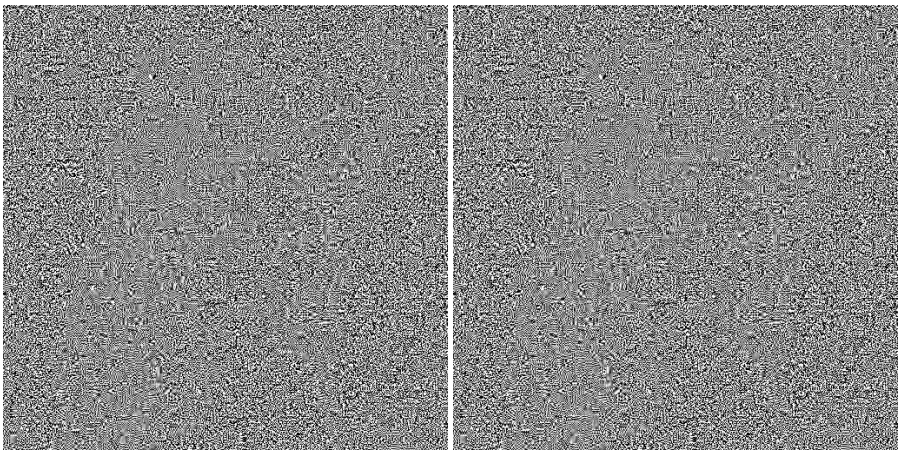
desired (i.e., points that create a smaller increase in rate than the optimal trade-off point, while increasing the PSNR only slightly), they cannot be found by solving the problem (4.4) since they are not on the upper boundary of the convex hull. The above observations motivate us to look for other approaches to construct BCs that achieve a good trade-off between security and bitrate.

4.6 Swap-Based Binary Code

As mentioned in the previous section, we would like to explore other methods for constructing good BCs (i.e., which offer sufficient secrecy with a small increase in rate). For this, we first analyze the structure of ECOptBC in order to acquire some insights that could help in our pursuit. Table 4.6 shows the seven highest probable QAC pairs and their allocation to categories for Lena, for four values of λ, namely 0, λ_{th}, $\lambda_{th}+1$, and a very high value $(500,000)$. We see that for $\lambda = 0$, the two fellows in each pair are at a high distance from each other. More specifically, one fellow is very small, while the other is very large. We call such pairs, *effective pairs* since they are largely responsible for the increase in distortion. For λ_{th} there are only two effective pairs, while for larger λ there is either one or no effective pair. We conclude that only two effective pairs could be sufficient to provide high secrecy. Therefore, we proceed to investigate binary codes with only



(A) Lena, MSB0, λ_{th} (B) Lena, MSB0, $\lambda_{th}+1$



(C) Lena, MSB1, λ_{th} (D) Lena, MSB1, $\lambda_{th}+1$

FIGURE 4.9: Eve's reconstructions using the replacement attack, after SE with ECOptBC for λ_{th} and $\lambda_{th}+1$ for Lena.

two effective pairs. A simple way to construct such a BC is to start with OrigBC, choose a pair (qac_0, qac_1) in a low size category $\mathcal{C}(s_L)$ and a pair (qac'_0, qac'_1) in a high size category $\mathcal{C}(s_H)$ and swap the second components of the two pairs. In other words, replace these two pairs by (qac_0, qac'_1) and (qac'_0, qac_1). We call such a BC, swap-based BC (SwapBC).

TABLE 4.6: The seven highest probability QAC pairs for Lena obtained by solving the problem (4.4) with $\lambda = 0$, λ_{th}, $\lambda_{th} + 1$ and $\lambda = 500000$. The effective pairs are in bold.

$\mathcal{C}'(s)$	$\lambda = 0$	$\lambda = \lambda_{th}$	$\lambda = \lambda_{th} + 1$	$\lambda = 500000$
$\mathcal{C}'(1)$	**(-1, +634)**	$(+1, -2)$	$(+1, -2)$	$(-1, +1)$
$\mathcal{C}'(2)$	**(+1, +508)**	$(+2, +3)$	$(+2, -3)$	$(-2, +2)$
	(-2, +506)	**(-1, +634)**	**(-1, +634)**	$(-3, +3)$
$\mathcal{C}'(3)$	**(+2, +497)**	$(-4, +4)$	$(+3, +4)$	$(-4, +4)$
	(-3, +472)	**(-3, +508)**	$(-4, +5)$	$(-5, +5)$
	(+3, -465)	$(-5, +5)$	$(-5, +6)$	$(-6, +6)$
	(+4, +459)	$(-6, +6)$	$(-6, +7)$	$(-7, +7)$

After finding the pairs subjected to the swap, we can apply EffCA to determine the codeword assignment. Recall that EffCA requires sorting all the pairs in decreasing order of their probabilities. To avoid the time overhead due to sorting, we propose a Faster Codeword Assignment (FastCA), which exploits the following properties of OrigBC: 1) all pairs in the same category are assigned codewords of the same length, therefore they do not need to be sorted in any particular order; 2) most of the pairs in category $\mathcal{C}(s+1)$ have smaller probabilities than the pairs in category $\mathcal{C}(s)$. As we will see shortly, FastCA is not only faster than EffCA, but it involves only a small number of changes in comparison with OrigBC, therefore the amount of SI that needs to be transmitted to the decoder will be smaller.

FastCA proceeds as follows. First, we determine for each $s, s_L < s < s_H$, the pair with the highest probability in $\mathcal{C}(s)$, denoted by $P(s, max)$, and the pair with the lowest probability in $\mathcal{C}(s)$, denoted by $P(s, min)$. Next, we compute the probabilities of the new pairs (qac_0, qac'_1), and (qac'_0, qac_1), and determine the pair with higher probability, denoted by P_1, while the other pair is denoted by P_2. We initially place P_1 in category $\mathcal{C}(s_L)$ in the place of the old pair (qac_0, qac_1) (i.e., we assign the same codewords). If $Pr(P_1) \geq Pr(P(s_L + 1, max))$, then this is

the final position of P_1. Otherwise, P_1 exchanges positions with $P(s_L+1, max)$. When the latter happens, $Pr(P_1)$ is further compared with $Pr(P(s_L+2, max)$. If $Pr(P_1) \geq Pr(P(s_L+2, max)$, then P_1 remains in $\mathcal{C}(s_L+1)$. Otherwise, P_1 exchanges positions with $P(s_L+2, max)$. The process continues in this manner until P_1 is placed in category $\mathcal{C}(s_1)$, where s_1 is the smallest integer larger than s_L such that $Pr(P(s_1, max)) > Pr(P_1) \geq Pr(P(s_1+1, max))$.

Pair P_2 is initially placed in the category $\mathcal{C}(s_H)$ in the place of the old pair (qac'_0, qac'_1). This will be its final position if $Pr(P_2) \leq Pr(P(s_H-1, min))$. Otherwise, the positions of P_2 and $P(s_H-1, min)$ are exchanged. When the latter happens, $Pr(P_2)$ is further compared with $Pr(P(s_H-2, min)$. If $Pr(P_2) \leq Pr(P(s_H-2, min)$, then P_2 remains in $\mathcal{C}(s_H-1)$. Otherwise, P_2 exchanges positions with $P(s_H-2, min)$. The process continues in this manner until P_2 is placed in category $\mathcal{C}(s_2)$, where s_2 is the largest integer smaller than s_H such that $Pr(P(s_2, min)) < Pr(P_1) \leq Pr(P(s_2-1, min))$.

The side information that needs to be communicated to the decoder consists of: s_L (using $\lceil \log_2(S) \rceil$ bits); qac_1 (s_L bits), qac'_1 ($S+1$ bits); s_1 and s_2 ($\lceil \log_2(S) \rceil$ bits for each). Then for each $s, s_L < s \leq s_1$ and for each $s, s_2 \leq s < s_H$ we need to specify which pair from $\mathcal{C}(s)$ was moved. For this, s bits are sufficient for each s.

Now the question remains what pairs from $\mathcal{C}(s_L)$ and $\mathcal{C}(s_H)$ to swap to achieve a high enough security level with only a small increase in bitrate. In order to guide us in making this decision, in the following section we perform the theoretical analysis of the distortion at Eve's side for SwapBC.

4.7 Theoretical Analysis of Swap-based BC

In the sequel, we will use the following shorter notations. We use x instead of qac_1 and z instead of qac'_1. Then $qac_0 = -\bar{x}$ and $qac'_0 = \bar{z}$, where $\bar{x} = \delta(s_L) - x$ and $\bar{z} = \delta(s_H) - z$. Thus, the old pairs are $(-\bar{x}, x)$, $(-\bar{z}, z)$ and the new pairs are $(-\bar{x}, z)$, $(-\bar{z}, x)$.

In order to analyze the security of SwapBC, we will evaluate the difference between the distortion at Eve's achieved with SwapBC and the distortion achieved with OrigBC, for both the MSB0 and MSB1 attacks, denoted by $\Delta D_{E,0}(x, z)$, respectively $\Delta D_{E,1}(x, z)$. In the analysis, we will use the assumption that the pmf of QACs is almost symmetric w.r.t. to the origin and that it obeys a Laplacian distribution, a claim that is supported by Figure 4.8, which depicts the histogram of QACs for test images. Note that the Laplacian distribution is commonly used to model the distribution of the transform coefficients [31]. Further, note that

$$\begin{aligned}
\Delta D_{E,0}(x,z) &= D_{E,0}(x,z) - D_{E,0,orig} \\
&= Pr(z)\Delta_Q^2(z+\bar{x})^2 + Pr(x)\Delta_Q^2(x+\bar{z})^2 \\
&\quad - Pr(z)\Delta_Q^2\delta(s_H)^2 - Pr(x)\Delta_Q^2\delta(s_L)^2 \\
&= \Delta_Q^2\{Pr(x)[(x+\bar{z})^2 - \delta(s_L)^2] \\
&\quad - Pr(z)[\delta(s_H)^2 - (z+\bar{x})^2]\},
\end{aligned}$$

$$\begin{aligned}\Delta D_{E,1}(x,z) &= D_{E,1}(x,z) - D_{E,1,orig} \\ &= Pr(-\bar{x})\Delta_Q^2(z+\bar{x})^2 + Pr(-\bar{z})\Delta_Q^2(x+\bar{z})^2 \\ &\quad - Pr(-\bar{z})\Delta_Q^2\delta(s_H)^2 - Pr(-\bar{x})\Delta_Q^2\delta(s_L)^2 \\ &= \Delta_Q^2(\{r(-\bar{x})[(z+\bar{x})^2 - \delta(s_L)^2] \\ &\quad - Pr(-\bar{z})[\delta(s_H)^2 - (x+\bar{z})^2]\}.\end{aligned}$$

Observation O1

Using the above relations and the fact that $Pr(-\bar{x}) \approx Pr(\bar{x})$ and $Pr(-\bar{z}) \approx Pr(\bar{z})$, it follows that

$$\Delta D_{E,0}(x,z) = \Delta D_{E,1}(\bar{x},\bar{z}),\ \ \Delta D_{E,1}(x,z) = \Delta D_{E,0}(\bar{x},\bar{z}).$$

Observation **O1** implies that for each s_L, it is sufficient to investigate all possible swaps obtained with $x \leq \delta(s_L)/2 + 1$ (i.e., when $x \leq \bar{x}$).

Observation O2

Assuming that the pmf of QACs obeys a Laplacian distribution, it follows that when $s_H - s_L$ is very large, the second term in $\Delta D_{E,0}(x,z)$, respectively $\Delta D_{E,1}(x,z)$, is much smaller than the first term. In addition, z and $\bar{z}$ are much larger than x and $\bar{x}$. Therefore, for sufficiently large $s_H - s_L$, we obtain

$$\begin{aligned}\Delta D_{E,0}(x,z) &\approx \Delta_Q^2 Pr(x)[(x+\bar{z})^2 - \delta(s_L)^2] \approx \Delta_Q^2 Pr(x)\bar{z}^2, \\ \Delta D_{E,1}(x,z) &\approx \Delta_Q^2 Pr(\bar{x})[(z+\bar{x})^2 - \delta(s_L)^2] \approx \Delta_Q^2 Pr(\bar{x})z^2.\end{aligned}$$

Observation O3

Based on Observation **O2**, we obtain that for sufficiently large $s_H - s_L$,

- $\Delta D_{E,0}(x,z)$ decreases from $\Delta_Q^2 Pr(x) 2^{2s_H}$ to $\frac{1}{4}\Delta_Q^2 Pr(x) 2^{2s_H}$;
- $\Delta D_{E,1}(x,z)$ increases from $\frac{1}{4}\Delta_Q^2 Pr(\bar{x}) 2^{2s_H}$ to $\Delta_Q^2 Pr(\bar{x}) 2^{2s_H}$.

The next step in our analysis is to evaluate the difference between the distortions under the MSB0 and the MSB1 attacks. For this, notice that

$$D_{E,k}(x,z) = D_{E,k,orig} + \Delta D_{E,k}(x,z),\ k = 0,1.$$

We first make the observation that $D_{E,0,orig} \approx D_{E,1,orig}$. This is because

$$D_{E,0,orig} = \sum_{s=1}^{S} \delta(s)^2 \Delta_Q^2 Pr_{s,+},\ D_{E,1,orig} = \sum_{s=1}^{S} \delta(s)^2 \Delta_Q^2 Pr_{s,-},$$

and $Pr_{s,+} \approx Pr_{s,-}$ based on the symmetry of the pmf of the QACs, where $Pr_{s,+}$ (respectively, $Pr_{s,-}$) denotes the sum of probabilities of QACs in $\mathcal{C}_+(s)$ (respectively, $\mathcal{C}_-(s)$). The above conclusion is also supported by the results shown in Table 4.7 which lists these distortions for the test images.

TABLE 4.7: $D_{E,0,orig}$ and $D_{E,1,orig}$ for the test images.

	Cameraman	Lena	Mandrill	Livingroom	Goldhill
$D_{E,0,orig}$ (dB)	32.77	29.17	31.86	30.45	28.58
$D_{E,1,orig}$ (dB)	32.10	28.75	32.03	30.66	28.38

Since $D_{E,0,orig} \approx D_{E,1,orig}$, we will use the following approximation in order to evaluate the gap in dB between $D_{E,0}(x,z)$ and $D_{E,1}(x,z)$, denoted by $G_{0/1}(x,z)$,

$$\begin{aligned} G_{0/1}(x,z) &= |10\log_{10} D_{E,1}(x,z) - 10\log_{10} D_{E,0}(x,z)| \\ &\approx \left|10\log_{10}\frac{\Delta D_{E,0}(x,z)}{\Delta D_{E,1}(x,z)}\right| \approx \left|10\log_{10}\frac{Pr(x)\bar{z}^2}{Pr(\bar{x})z^2}\right|. \end{aligned}$$

Further, note that the function $\frac{Pr(x)\bar{z}^2}{Pr(\bar{x})z^2}$ is decreasing in z and $\frac{Pr(x)\bar{z}^2}{Pr(\bar{x})z^2} = 1$ when $z = z_{eq}$, where

$$z_{eq} = \frac{\sqrt{Pr(x)}\delta(s_H)}{\sqrt{Pr(x)} + \sqrt{Pr(\bar{x})}}. \tag{4.5}$$

Recall that $\delta(s_H) = 3 \times 2^{s-1} - 1 \approx 3 \times 2^{s-1}$. Replacing in the above, we obtain that $z_{eq} \in \mathcal{C}_+(s_H)$ if and only if

$$\frac{1}{4} \leq \frac{Pr(\bar{x})}{Pr(x)} \leq 4. \tag{4.6}$$

When $x = 1$, we have $\bar{x} = 1$, therefore the above relations are true. From Figure 4.8 we can also see that (4.6) is satisfied for the images in our study for small s_L. We conclude that when (4.6) holds, $G_{0/1}(x,z)$ decreases from $10\log_{10}\frac{4Pr(x)}{Pr(\bar{x})}$ from to 0 when z varies between $\alpha(s)$ and z_{eq}, and $G_{0/1}(x,z)$ increases from 0 to $10\log_{10}\frac{4Pr(\bar{x})}{Pr(x)}$ when z varies between z_{eq} and $\beta(s_H)$. It follows that

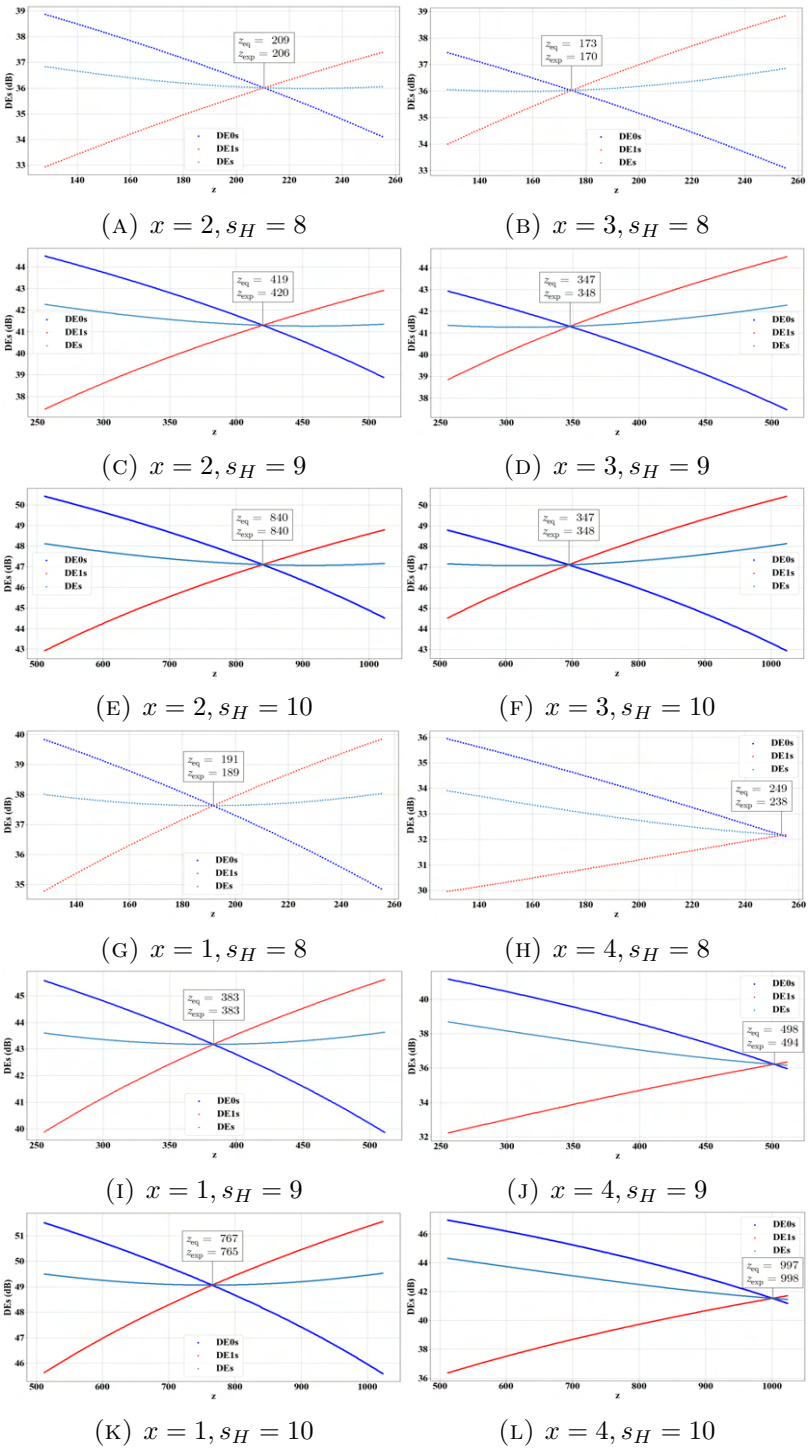
$$\begin{aligned} \max_{z\in\mathcal{C}_+(s_H)} G_{0/1}(x,z) &= 10\log_{10}\left(4\max\left(\frac{Pr(x)}{Pr(\bar{x})}, \frac{Pr(\bar{x})}{Pr(x)}\right)\right) \\ &\geq 10\log_{10} 4 \approx 6dB. \end{aligned}$$

We see that the gap between the quality of the reconstruction under the MSB0 attack versus the MSB1 attack can be as high as 6 dB, which is substantial. Having a large difference between the distortions under the MSB0 and MSB1 attacks is undesirable because this means that the encryption could be resilient under one of the attacks, but vulnerable under the other one. Under the above considerations, we conclude that it is desirable to choose a value of z close to z_{eq}.

In the following section, we validate empirically the conclusions drawn in this section.

4.8 Experimental Results with Swap-based BC

In this section, we empirically assess the effect of applying SwapBC to the test images, for $1 \leq s_L \leq 3$ and $8 \leq s_H \leq 10$. Figure 4.10 illustrates the variations of $D_{E,0}(x, z)$, $D_{E,1}(x, z)$, and $D_E(x, z)$ for fixed x, as z takes all values in $\mathcal{C}_+(s_H)$, for Lena. We consider $x = 1, 2, 3, 4$ and $s_H = 8, 9, 10$. The curves corresponding to the other images in our test set are shown in Figures A3.1, A3.2, A3.3, and A3.4 in Appendix C. It can be observed that as z increases from $\alpha(s_H)$ to $\beta(s_H)$, $D_{E,0}(x, z)$ decreases and $D_{E,1}(x, z)$ increases, behaviour which is in agreement with Observation **O3**. Additionally, we note that in most of the cases the two curves $D_{E,0}(x, z)$ and $D_{E,1}(x, z)$ meet at some point z_{exp}, which is very close to z_{eq} defined in equation (4.5). Note that when $x = 2$, we have $\bar{x} = 3$, and it can be observed that the relations in Observation **O1** are approximately satisfied. Finally, we can see that the gap between $D_E(x, z_{eq})$ and the largest value of $D_E(x, z)$ is small, as we have predicted.



(A) $x = 2, s_H = 8$ (B) $x = 3, s_H = 8$

(C) $x = 2, s_H = 9$ (D) $x = 3, s_H = 9$

(E) $x = 2, s_H = 10$ (F) $x = 3, s_H = 10$

(G) $x = 1, s_H = 8$ (H) $x = 4, s_H = 8$

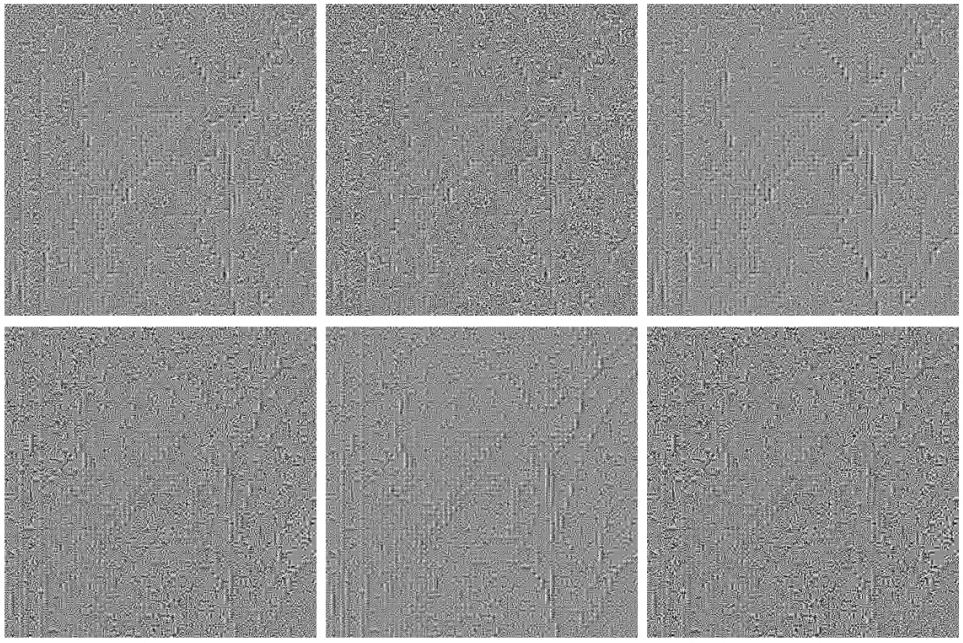
(I) $x = 1, s_H = 9$ (J) $x = 4, s_H = 9$

(K) $x = 1, s_H = 10$ (L) $x = 4, s_H = 10$

FIGURE 4.10: $D_{E,0}(x, z)$, $D_{E,1}(x, z)$, and $D_E(x, z)$ versus $z \in \mathcal{C}_+(s_H)$ for Lena.

We also see from Figure 4.10 and the other figures in Appendix C, that our prediction that the gap between $D_{E,0}(x,z)$ and $D_{E,1}(x,z)$ can reach approximately 6 dB or even higher values is confirmed for the case when s_H is 9 or 10. In order to illustrate the visual effect of this gap, we show in Figure 4.11 the reconstructions of Lena at Eve's side for $x=2$, $s_H=9$ and three values of z, namely z_{eq} (we actually use $\lceil z_{eq} \rceil$), $\alpha(s_H)$, and $\beta(s_H)$. The corresponding reconstructions for the other test images are shown in Figures A3.5, A3.6, A3.7, and A3.8 in Appendix C. We notice that there is some discrepancy between the levels of degradation under the MSB0 and MSB1 attacks for $z=\alpha(s_H)$ and $z=\beta(s_H)$. For Lena, one reconstruction is highly degraded, while the other reconstruction reveals a slight contour of the objects in the image. On the other hand, when $z=z_{eq}$, the reconstructions are almost equally degraded.

In Figure 4.12 we present the reconstructions at Eve's under the MSB0 and MSB1 attacks for Lena, when $x=1,2,4$, $s_H=8,9,10$ and $z=z_{eq}$. Table 4.8 also shows the PSNRs and SSIMs of the reconstructions for all aforementioned combinations. The figures and tables corresponding to the remaining test images are in Appendix C (Figures A3.9, A3.10, A3.11, and A3.12 and Tables A3.2, A3.1, A3.3, and A3.4). It can be observed that the security level increases as $s_H - s_L$ increases, as expected. For all test images, the degradation is extremely high, and hence appropriate for confidential encryption, when $(x,s_H)=(1,10),(2,10),(3,10)$. The degradation when $(x,s_H)=(1,9),(4,10)$ is slightly lower, but it could still ensure a high level of secrecy for all images, except the Cameraman. Finally, for most images, there are some other swaps that also guarantee high security, namely, $(x,s_H)=(2,9),(3,9)$ for Lena, Livingroom, and Goldhill.



(a) (b) (c)

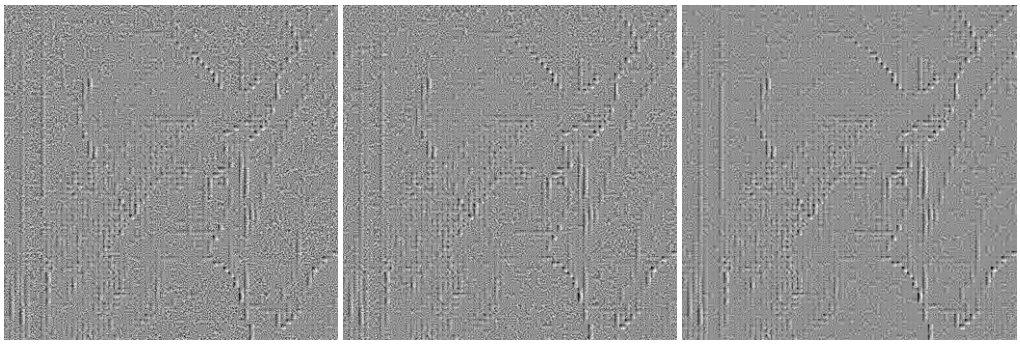
FIGURE 4.11: Comparison of the levels of degradation at Eve's under the MSB0 and MSB1 attacks for SwapBC for Lena, when $x = 2$, $s_H = 9$, and z is equal to a) z_{eq}, b) $\alpha(s_H)$, and c) $\beta(s_H)$. Top: MSB0 attack; bottom: MSB1 attack.

We also notice that the remaining swaps provide intermediate levels of degradation at Eve's and therefore they may be appropriate for scenarios where more flexibility is needed in choosing the degree of secrecy.

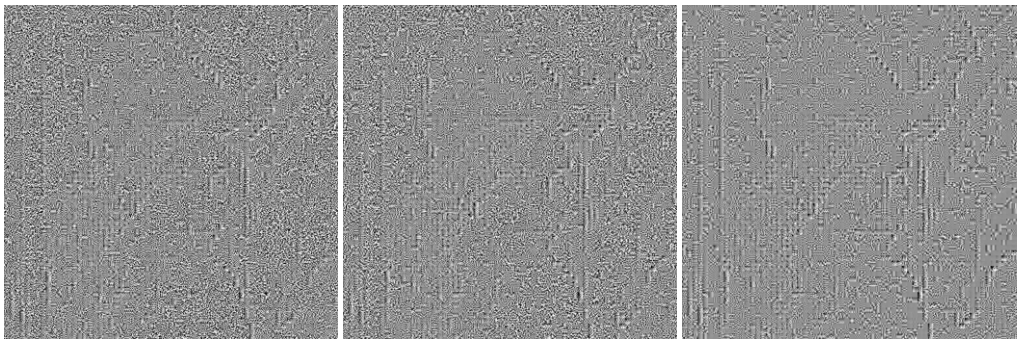
TABLE 4.8: PSNR and SSIM of Eve's reconstructions in SwapBC for Lena.

x		+1			+2			+3			+4		
s_H		8	9	10	8	9	10	8	9	10	8	9	10
PSNR	MSB0	9.70	8.48	8.41	11.33	9.33	8.51	12.12	10.10	9.33	14.68	12.78	10.66
	MSB1	9.52	8.46	8.40	11.30	10.31	9.51	11.13	9.26	8.49	15.90	14.19	13.10
SSIM	MSB0	0.019	0.009	0.008	0.039	0.016	0.011	0.076	0.051	0.047	0.202	0.135	0.111
	MSB1	0.016	0.008	0.009	0.088	0.060	0.057	0.038	0.015	0.010	0.408	0.376	0.371

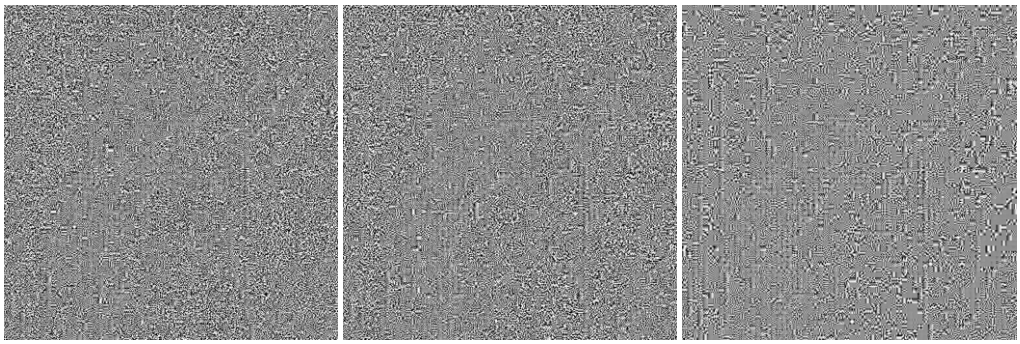
Next, we evaluate the rate increase obtained with SwapBC. In our experiments, we have observed that the rate increase with FastCA is very similar to that achieved



(A) $x = +1, s_H = 8$ (B) $x = +2, s_H = 8$ (C) $x = +4, s_H = 8$



(D) $x = +1, s_H = 9$ (E) $x = +2, s_H = 9$ (F) $x = +4, s_H = 9$



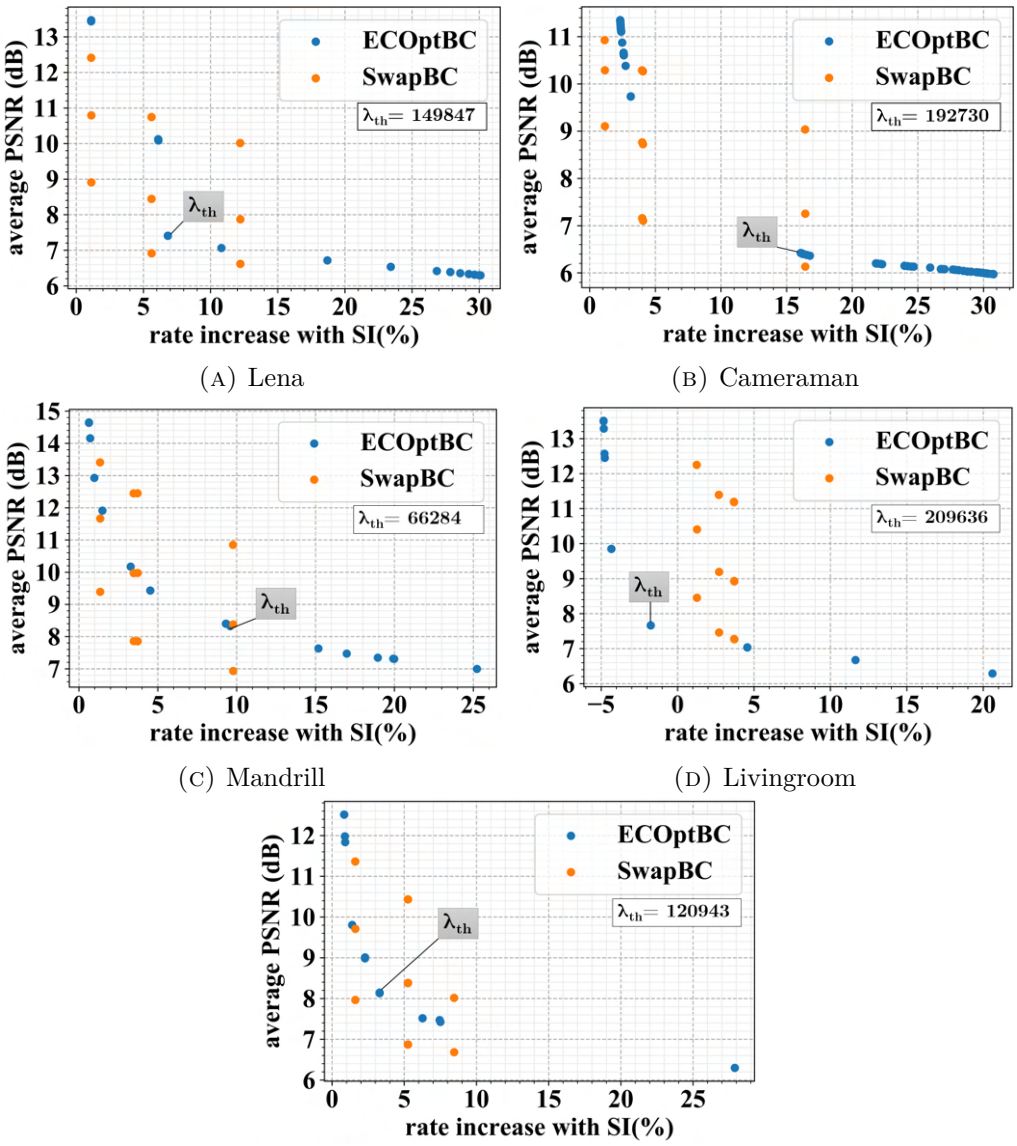
(G) $x = +1, s_H = 10$ (H) $x = +2, s_H = 10$ (I) $x = +4, s_H = 10$

FIGURE 4.12: Eve's reconstructions by MSB0 arrack for Lena after SE with SwapBC when $x = 1, 2, 4$, z_{eq} and $s_H = 8, 9, 10$.

with EffCA for all images, except for Livingroom, while the amount of SI is much smaller for FastCA. Therefore, FastCA provides a smaller rate overall (except for Livingroom). Therefore, we will only present the results achieved with FastCA. Another interesting observation is that for fixed x, the rate increase is the same when s_H is $8, 9$ or 10. In other words, only the choice of x determines the rate increase. Table 4.9 presents the percentage of rate increase achieved with SwapBC (ΔR_{swap}) when $x = 1, 2, 3$, along with the amount of SI, for all test images. We notice that for all images except Livingroom, there is a high discrepancy in the rate increase when $x \in \mathcal{C}(1)$ versus $x \in \mathcal{C}(2)$. In the former case, ΔR_{swap} is very small for Livingroom (2.63%), while for the other images, it ranges from 8.37% (for Goldhill) to 16.28% (for Cameraman). When $x \in \mathcal{C}(2)$, for all images ΔR_{swap} is lower than 5.54%, while for Cameraman, Mandrill, and Livingroom it is even smaller than 4%. While ΔR_{swap} might be too high when $x \in \mathcal{C}(1)$ for some images, the value of ΔR_{swap} for $x \in \mathcal{C}(2)$ is acceptable. Finally, it can be observed in Table 4.9 that the amount of SI is very small, namely at most 0.14%.

We conclude that by using SwapBC, confidential encryption can be achieved with only a modest sacrifice in terms of compression efficiency. This is guaranteed by choosing $x = 2$ or $x = 3$ and $z = z_{eq} \in \mathcal{C}(10)$, but other choices may work as well depending on the image.

Let us compare now the performances of SwapBC and ECOptBC. In Figure 4.13, we plot the average PSNR versus the increase in rate including the SI for both methods. We first point out that the amount of SI needed for ECOptBC (ranging from 0.87% to 2.46%) is much higher than for SwapBC (which is at most 0.14%). As a result, for all images except Livingroom, SwapBC can achieve some



(A) Lena (B) Cameraman

(C) Mandrill (D) Livingroom

(E) Goldhill

FIGURE 4.13: Average PSNR versus rate increase including the SI, for ECOptBC and SwapBC.

TABLE 4.9: Percentage of rate increase (ΔR_{swap}) and of SI for SwapBC with FastCA.

	Cameraman		Lena		Mandrill		Livingroom		Goldhill	
x	rate	SI	rate	SI	rate	SI	rate	SI	rate	SI
+1	16.28	0.14	12.13	0.08	9.70	0.07	2.63	0.08	8.37	0.08
+2	3.86	0.13	5.54	0.07	3.41	0.06	3.64	0.07	5.17	0.07
+3	3.93	0.13	5.53	0.07	3.66	0.07	3.63	0.07	5.19	0.07

PSNR-rate pairs there are better than the pairs achieved with ECOptBC. This implies that SwapBC can allow for confidential security to be achieved with a smaller rate increase than with ECOptBC.

4.9 Additional Security Analysis

In this section, we continue the investigation of the degree of security achieved by the SE method used in our work when the QACs are coded using the proposed BCs, namely: 1) ECOptBC at λ_{th} and 2) SwapBC with $x = 2$ and $z = z_{eq} \in \mathcal{C}(10)$. We demonstrate the perceptual degradation of the encrypted images (without using the replacement attack, but assuming that Eve knows the BC used) as well as the results of the sketch attacks and of the statistical analysis.

4.9.1 Perceptual Degradation

The PSNR and SSIM values of the ciphered images encrypted by our methods and the latest efforts CtE [55, 45, 18] are shown in Table 4.10. Note that Qin *et al.* [55] and He *et al.* [18] do not report the SSIM, therefore these values are not included in the table. The results are for the test images of the latter works for a better comparison. It can be observed that our encryption method with both ECOptBC and SwapBC code has lower PSNRs than the methods of [55]

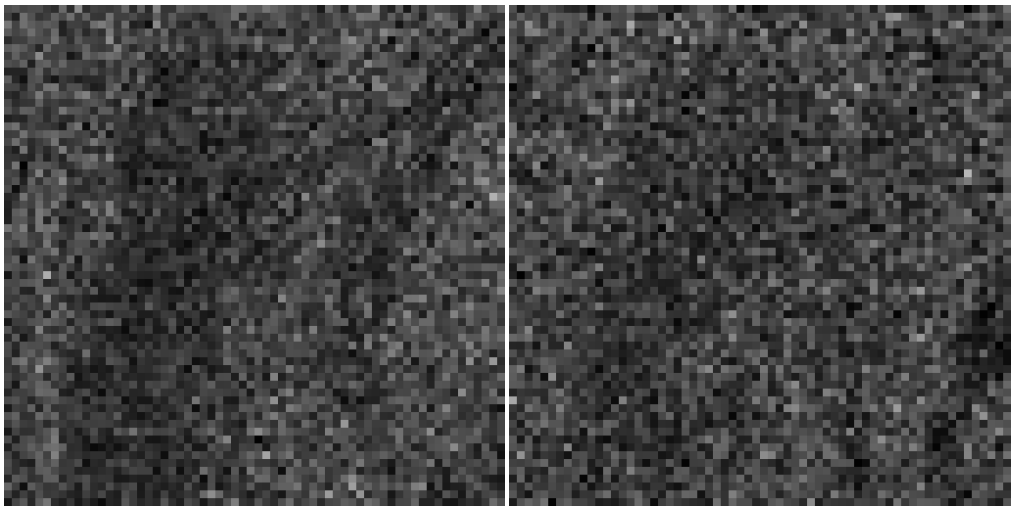
and [18]. Although the PSNRs of [45] are slightly lower than with our method, the SSIMs of our encrypted images are significantly smaller, which demonstrate higher degradation. It should be noted that by decreasing the value of λ_{th} or of x, higher levels of degradation can be achieved. Additionally, by increasing the aforementioned values or by decreasing s_H, lower levels of security will be provided. Therefore, using these parameters, a fine tunable encryption of JPEG images will be available for different applications.

TABLE 4.10: PSNRs (P) and SSIMs (S) of encrypted images of our proposed method and coding of ECOptBC and SwapBC and encryption methods of [55, 45, 18].

	ECOptBC		SwapBC		[45]		[55]	[18]
	P	S	P	S	P	S	P	P
Plane	6.33	−0.0003	6.26	−0.0027	5.44	0.1205	7.83	10.81
Peppers	7.54	−0.0001	6.61	−0.0045	6.30	0.0866	7.85	10.02
Lena	7.07	−0.0006	6.78	−0.0050	6.13	0.0903	9.02	10.60
Mandrill	7.89	0.0054	7.51	−0.0049	6.31	0.0227	–	–
Aerial	6.56	0.0044	6.47	−0.0058	–	–	7.92	9.28
Living.	7.33	−0.0014	7.07	−0.0059	–	–	9.42	11.76
Elaine	7.46	0.0012	6.97	−0.0039	6.18	0.1100	–	–
Boat	7.60	−0.0034	6.93	−0.0047	6.17	0.0624	–	–
House	6.58	0.0002	6.86	0.0077	5.85	0.0668	–	–
Sailboat	6.47	0.0004	6.38	−0.0058	6.00	0.0495	–	–
Splash	6.60	0.0007	6.54	−0.0041	5.63	0.1882	–	–

4.9.2 Sketch Attacks

Sketch attacks are used to obtain the outlines of the original image. There are several types of sketch attacks, such as DC category attack (DCCA), Non-zero coefficient count (NZC), and energy of ACCs (EAC) attack. Since our binary codes only affect the ACCs, we will discuss only theattacks based on ACCs, namely EAC and NZC. Figure 4.14 demonstrates the EAC attack for Lena coded by ECOptBC and SwapBC, as well as Figure A3.13 in Appendix C for all other test images.



(A) ECOptBC (B) SwapBC

FIGURE 4.14: Results of EAC attack on encrypted images coded by ECOptBC and SwapBC

It can be observed that our encryption method is safe against the EAC sketch attack. The number of non-zero ACCs of DCT blocks remains unchanged leading to vulnerability to the NZC attack. To solved this issue, the SE method could be combined with permutations of the DCT blocks without the DCCs, as one extra phase of encryption.

4.9.3 Statistical Analysis

Shannon Entropy: A high (Shannon) entropy of the pixel values of an encrypted image implies high randomness of the pixels' distribution. For an 8-bit grayscale image, the maximum entropy is 8. An encrypted image with the entropy close to 8 is considered a proper noise-like image that does not reveal any information about the original image. Table 4.11 shows the entropy of the original images and of their encrypted counterparts. Furthermore, the entropy of other JPEG encryption methods such as [18, 33, 32, 55] has been reported for test images, including Aerial,

Plane, Livingroom, Peppers, Lena, and Mandrill. The average entropy is 7.70 in [18], 7.20 in [32], 7.50 in [33], and 7.80 in [55], while it is 7.98 in our work with both BC schemes. We conclude that our method beets the other schemes with respect to this measure of security.

Correlations: Plain images generally have high correlations between their adjacent pixels, whereas encrypted images should have low correlations. Table 4.11 contains the correlation values of the adjacent horizontal pixels of the original images and their encrypted ones. The low value correlations of our encrypted images demonstarte the strength of SE method with both BC assignments. For comparison, we consider the work of Minemura *et al.* [45], where the horizontal correlation for encrypted Lena is reported. The latter value equals 0.38, while in our case the correlation for encrypted Lena is 0.06 indicating a higher level of security for our method.

TABLE 4.11: Shannon entropy (H) and pixels' horizontal correlations (Corr) of original images and the encrypted ones with ECOptBC and SwapBC schemes.

	Cameraman		Lena		Mandrill		Livingroom		Goldhill	
	H	Corr	H	Corr	H	Corr	H	Corr	H	Corr
Original	7.05	0.98	7.45	0.97	7.29	0.93	7.30	0.95	7.48	0.97
ECOptBC	7.98	0.12	7.98	0.06	7.98	0.15	7.98	0.13	7.98	0.24
SwapBC	7.98	0.32	7.98	0.06	7.98	0.16	7.98	0.01	7.98	0.03

4.10 Conclusion

Selective encryption (SE) of JPEG images was deemed not to be sufficiently secure. In this chapter, we demonstrated that confidential security can be achieved if

we judiciously design the binary code representing the non-zero quantized AC coefficients. More specifically, we considered the SE method that encrypts only the MSBs of the QACs together with the full encryption of the bitstream encoding the DC coefficients, and we proposed two methods for the design of the BC used to represent the QACs with the purpose of increasing the security. The first method solves an optimization problem that aims at maximizing the distortion at Eve, while keeping at bay the increase in rate. By changing the value of a certain parameter λ, various trade-offs between the level of secrecy and the increase in bitrate can be achieved, including the highest confidentiality, which can be attained at the cost of a small increase in bitrate. Further, we sought a faster and simpler approach for the BC design. To this end, we first analyzed the structure of the BC obtained with the optimization-based method and concluded that only swapping a few codewords in the original binary representation could be sufficient to guarantee high security. The theoretical analysis of the distortion at Eve's helped us identify the best candidate pairs for swapping. We further showed empirically that the swap-based approach can achieve high-security levels at bitrates comparable to or even smaller than the optimization-based approach.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

This thesis considered the fact that many SE methods proposed to date for images encrypt the MSB of some syntax elements in the case of compressed images or the MSB plane of the binary representations of the pixels' intensity values for uncompressed images. Our main idea was that the BC which encodes the syntax elements or pixels' intensity values can affect the quality of the reconstructed image at Eve's side. Therefore, we investigated the effect of BC on the degradation of Eve's images obtained by the replacement attack. Next, developed BC design methods to construct new BCs rather than conventional ones to maximize the Eve's degradation to increase the security of an MSB-SE method. Moreover, it is considered that if the stream of the MSBs is subjected to compression, the length of the compressed MSB stream can be affected by the assigned BC too. Since the shorter the compressed MSB stream is, the less overhead of the encryption is achieved, in Chapter 2, we addressed the problem of an optimal fixed-length

BC design to simultaneously maximize Eve's distortion and minimize the length of the compressed MSB stream. We assumed that SE is applied to a sequence of compressed quantization indexes and only the plane of MSBs (possibly after being entropy-coded) is encrypted. It was demonstrated how the mapping of binary sequences to quantization cells could control the level of security by influencing the distortion at Eve's side while also impacting the length of the compressed MSB stream. The optimal BC design problem was the maximization of a weighted sum of Eve's distortion and of the probability of the MSB being 0. It was shown that the problem is equivalent to a maximum weight matching problem, which can be solved in polynomial time. Additionally, it was proved that when the source and the quantizer are symmetric, the problem can be cast as a linear program. Experimental comparison with conventional BCs such as NBC and FBC demonstrated that the proposed BC design could significantly improve the security and computational overhead of MSB-SE methods.

In Chapter 3, optimized BC design method of Chapter 2 was utilized for coding the binary representation of pixel values to make more secure the SE of uncompressed or "lightly" compressed images by encrypting one or two MSB planes. Similar to Chapter 2, two scenarios are also considered here; Scenario A where the images are not compressed, and scenario B, where only the MSB plane is compressed before being encrypted. We presented that higher visual secrecy can be obtained in comparison with the conventional BCs. In this chapter, we also utilized the intrinsic property of natural images with similar intensities for spatially close pixels. We developed structured BCs for uncompressed images that increase the security of SE of MSB encryption but without the computational overhead of the

optimization. A family of structured BCs, termed SBCs, were introduced, which could achieve a high distance between the reconstructions of intensity values that differ by a small amount. Theoretical and experimental results showed that the order d of the code could provide various levels of secrecy in SBC. It was shown empirically that SBC, with proper order, can destroy the smooth areas, blur the edges in Eve's reconstruction, and increase security remarkably.

In Chapter 4, to enhance the security of MSB-SE of JPEG compressed images, we developed two methods for the design of variable-length BCs to represent the QACs. The first approach solved an optimization problem for maximizing Eve's distortion while keeping the rate increase small. Flexibility in the choice of the security level was obtained by changing the value of the parameter λ, which could provide trade-offs between the security and the increase in bitrate. Next, we analyzed the structure of the optimized BC and demonstrated that only swapping a few codewords in the baseline coding could be sufficient to achieve high security. Best candidate pairs for swapping were identified by theoretical analysis of the distortion at Eve's and the rate increase. Experimental results also showed that the swap-based method, which has a smaller computational cost, can achieve high-security levels at bitrates comparable to or even smaller than the optimization-based method.

5.2 Future Work

As a future work to extend Chapter 2, two MSBs can be considered for encryption. This entails solving an optimization problem to determine the quadruple of values that have two different MSBs and the same remaining bits that maximize the

distortion and minimize the length of the bitstream of the MSBs. One approach can be a greedy technique that first finds the optimized BC by considering only the first MSB encryption and determines the pairs. Next, the found pairs are subjected to another optimization stage to find the optimal pairs of pairs, i.e., quadruples.

To encrypt the uncompressed images, different stages of encryption are often used, including block scrambling, block rotations, negative-positive transformations, and color component shuffling. In Chapter 3, we considered only the encryption of MSBs, which is equivalent to negative-positive transformations. This is not sufficient to achieve complete confidentiality of the encrypted images. However, the proposed BCs can increase the degradation of the encrypted image significantly with only this stage of encryption, compared with NBC. As a future work, a combination of other steps of encryption with the MSB encryption can be investigated to determine which steps are necessary to attain complete confidential encryption and which steps can be removed due to the security increase of the new BC assignments.

In Chapter 3, we also observed that the order d_{min} of SBC where the minimum correlation (or maximum security) occurred is image specific. It could be considered as a future effort to find a method to determine d_{min} for each image before constructing an efficient SBC for that image. For instance, the value of d_{min} can be computed for each image in a database as the point where minimum correlation occurs for the encrypted image. Then a neural network is trained with the input of these images and the output of their corresponding d_{min} values. After proper training of the network, any new image can be inputted to the network, and its

d_{min} is achieved as the output.

In Chapter 4, new BCs were designed to make the proposed SE method (Encryption of DCCs and MSB of QACs) secure enough, eliminating the necessity of other phases of encryption. However, in our proposed SE, the number of zero-value DCT coefficients of each block remains intact. This fact makes the encrypted image vulnerable to NZC sketch attacks. Thus, as future work, another step of encryption, such as block permutation, must be investigated and included to make the SE method resilient against this kind of attack too.

Appendix A

Lemmas used in Chapter 2

Lemma 1. *If the pdf $f(x)$ and the quantizer Q are symmetric about 0, and the distortion measure is $d(x,y) = \rho(|x-y|)$, then $w_{i,j}^{(l)} = w_{\bar{j},\bar{i}}^{(l)}$, $0 \le i < j \le M-1$, $l \in \{1,2,3\}$.*

Proof. For each measurable set $S \subseteq \mathbb{R}$ and $y \in \mathbb{R}$ let $L(S,y) = \int_S \rho(|x-y|)f(x)\,dx$. Then the following property holds:

Property A: For any measurable set $S \subseteq \mathbb{R}$ and $y \in \mathbb{R}$, $L(S,y) = L(-S,-y)$.

The proof relies on the symmetry about 0 of the pdf and of the absolute value function. Specifically, one has $L(S,y) = \int_S \rho(|-x+y|)f(-x)\,dx = \int_{-S} \rho(|z+y|)f(z)\,dz = L(-S,-y)$, where the second equality is based on the change of variable $z = -x$.

Property A, together with the fact that $C_{\bar{j}} \cup C_{\bar{i}} = -(C_i \cup C_j)$, leads to $y_{\bar{j},\bar{i}}^{(1)} = \arg\min_{y\in\mathbb{R}} L(C_{\bar{j}} \cup C_{\bar{i}}, y) = \arg\min_{y\in\mathbb{R}} L(C_i \cup C_j, -y) = -y_{i,j}^{(1)}$. On the other hand, the symmetry of the quantizer readily implies that $y_{\bar{j},\bar{i}}^{(2)} = -y_{i,j}^{(2)}$. Then, for $l =$

$1, 2$, one has $D^{(l)}_{\bar{j},\bar{i}} = L(C_{\bar{j}} \cup C_{\bar{i}}, y^{(l)}_{\bar{j},\bar{i}}) = L(-(C_i \cup C_j), -y^{(l)}_{i,j}) = L(C_i \cup C_j, y^{(l)}_{i,j}) = D^{(l)}_{i,j}$, where the second last equality is based on Property A. Let us consider now $l = 3$. Using again Property A in conjunction with $y_{\bar{i}} = -y_i$, $y_{\bar{j}} = -y_j$ and $C_{\bar{j}} \cup C_{\bar{i}} = -(C_i \cup C_j)$, one obtains $D^{(3)}_{\bar{j},\bar{i}} = 0.5L(C_{\bar{j}} \cup C_{\bar{i}}, y_{\bar{i}}) + 0.5L(C_{\bar{j}} \cup C_{\bar{i}}, y_{\bar{j}}) = 0.5L(C_i \cup C_j, y_i) + 0.5L(C_i \cup C_j, y_j) = D^{(3)}_{i,j}$. Further, for each $k, 0 \leq k \leq M-1$,

$$P(C_{\bar{k}}) = \int_{C_{\bar{k}}} f(x)\, dx = \int_{-C_k} f(-x)\, dx = \int_{C_k} f(z)\, dz = P(C_k). \tag{A.1}$$

This implies that $P_{0,\bar{j},\bar{i}} = P_{0,i,j}$. Now the conclusion of the lemma follows immediately. □

Lemma 2. *The tuple* $\mathbf{z}^{(1)}$ *defined in (2.16) is a feasible solution to problem (2.12).*

Proof. Let us first fix some arbitrary $i \in \mathcal{I}$. Based on (2.16), one obtains

$$\sum_{j\in\bar{\mathcal{J}}} z^{(1)}_{i,j} = \sum_{j=M/2}^{\bar{i}-1} z^{(1)}_{i,j} + z^{(1)}_{i,\bar{i}} + \sum_{j=\bar{i}+1}^{M-1} z^{(1)}_{i,j} = \sum_{j=M/2}^{\bar{i}-1} \left(x^{(1)}_{i,j} + x^{(1)}_{i,\bar{j}}\right) + x^{(1)}_{i,\bar{i}} + \sum_{j=\bar{i}+1}^{M-1} \left(x^{(1)}_{i,j} + x^{(1)}_{\bar{j},i}\right)$$
$$= \sum_{j=M/2}^{\bar{i}-1} x^{(1)}_{i,\bar{j}} + \sum_{j=M/2}^{\bar{i}-1} x^{(1)}_{i,j} + x^{(1)}_{i,\bar{i}} + \sum_{j=\bar{i}+1}^{M-1} x^{(1)}_{i,j} + \sum_{j=\bar{i}+1}^{M-1} x^{(1)}_{\bar{j},i}.$$

By making the substitutions $m = \bar{j}$ in the first summation and $k = \bar{j}$ in the last one, leads to

$$\sum_{j\in\bar{\mathcal{J}}} z^{(1)}_{i,j} = \underbrace{\sum_{m=i+1}^{M/2-1} x^{(1)}_{i,m}}_{\text{rename } m \text{ by } j} + \sum_{j=M/2}^{M-1} x^{(1)}_{i,j} + \sum_{k=0}^{i-1} x^{(1)}_{k,i} = \sum_{j=i+1}^{M-1} x^{(1)}_{i,j} + \sum_{k=0}^{i-1} x^{(1)}_{i,j} = 1, \tag{A.2}$$

where the last equality is based on (2.14). Let us fix some arbitrary $j \in \bar{\mathcal{J}}$. Using (2.16) leads to

$$\sum_{i\in\mathcal{J}} z_{i,j}^{(1)} = \sum_{i=0}^{\bar{j}-1} z_{i,j}^{(1)} + z_{\bar{j},j}^{(1)} + \sum_{i=\bar{j}+1}^{M/2-1} z_{i,j}^{(1)} = \sum_{i=0}^{\bar{j}-1} \left(x_{i,j}^{(1)} + x_{i,\bar{j}}^{(1)}\right) + x_{\bar{j},j}^{(1)} + \sum_{i=\bar{j}+1}^{M/2-1} \left(x_{i,j}^{(1)} + x_{\bar{j},i}^{(1)}\right)$$
$$= \sum_{i=0}^{\bar{j}-1} x_{i,\bar{j}}^{(1)} + \sum_{i=0}^{\bar{j}-1} x_{i,j}^{(1)} + x_{\bar{j},j}^{(1)} + \sum_{i=\bar{j}+1}^{M/2-1} x_{i,j}^{(1)} + \sum_{i=\bar{j}+1}^{M/2-1} x_{\bar{j},i}^{(1)}. \tag{A.3}$$

According to (2.15), one has $x_{i,\bar{j}}^{(1)} = x_{j,\bar{i}}^{(1)}$. Applying the above, then performing the substitution $n = \bar{i}$, yields

$$\sum_{i=0}^{\bar{j}-1} x_{i,\bar{j}}^{(1)} = \sum_{i=0}^{\bar{j}-1} x_{j,\bar{i}}^{(1)} = \sum_{n=j+1}^{M-1} x_{j,n}^{(1)}. \tag{A.4}$$

Similarly, using $x_{\bar{j},i}^{(1)} = x_{\bar{i},j}^{(1)}$ followed by the substitution $m = \bar{i}$, one obtains

$$\sum_{i=\bar{j}+1}^{M/2-1} x_{\bar{j},i}^{(1)} = \sum_{i=\bar{j}+1}^{M/2-1} x_{\bar{i},j}^{(1)} = \sum_{m=M/2}^{j-1} x_{m,j}^{(1)}. \tag{A.5}$$

By plugging (A.4) and (A.5) in (A.3) leads to

$$\sum_{i\in\mathcal{J}} z_{i,j}^{(1)} = \sum_{n=j+1}^{M-1} x_{j,n}^{(1)} + \sum_{i=0}^{M/2-1} x_{i,j}^{(1)} + \sum_{m=M/2}^{j-1} x_{m,j}^{(1)} = \sum_{m=0}^{j-1} x_{m,j}^{(1)} + \sum_{n=j+1}^{M-1} x_{j,n}^{(1)} = 1,$$

where the last equality follows by replacing in (2.14) i with j, j with n and k with m. With this, the proof is complete. □

Lemma 3. *Assume that the pdf $f(x)$ and the quantizer Q are symmetric about 0. Let the distortion function satisfy $d(x,y) = \rho(|x-y|)$, where ρ is a nondecreasing function. Then, $w_{m,n}^{(l)} \le w_{m,\bar{n}}^{(l)}$ and $w_{m,n}^{(l)} \le w_{n,\bar{m}}^{(l)}$ for all $0 \le m < n \le M/2-1$ and*

$l \in \{1, 3\}$.

Proof. In virtue of Lemma 1, one has $w^{(l)}_{m,\bar{n}} = w^{(l)}_{n,\bar{m}}$. Therefore, it is sufficient to prove only that $w^{(l)}_{m,n} \leq w^{(l)}_{m,\bar{n}}$. Additionally, $P_{0,m,n} = P_{0,m,\bar{n}}$ since $P(C_n) = P(C_{\bar{n}})$ according to (A.1). We conclude that we only need to prove that

$$D^{(l)}_{m,n} \leq D^{(l)}_{m,\bar{n}}. \tag{A.6}$$

For this, we first prove the following property.

Property B: For any $y \leq 0$ and measurable set $S \subseteq (-\infty, 0]$, $L(S, y) \leq L(S, -y)$. To prove the above claim it is sufficient to show that $\rho(|x-y|) \leq \rho(|x+y|)$ for any $x, y \leq 0$. Indeed, when $x, y \leq 0$, one has $-|x+y| = x+y \leq x-y \leq -x-y = |x+y|$, which implies that $|x - y| \leq |x + y|$ and further leads to $\rho(|x - y|) \leq \rho(|x + y|)$ since ρ is nondecreasing.

Let us prove now (A.6) for the case $l = 1$. Consider first the situation when $y^{(1)}_{m,\bar{n}} \geq 0$. Since $C_m \subseteq (-\infty, 0]$, according to Property B, one has

$$L(C_m, -y^{(1)}_{m,\bar{n}}) \leq L(C_m, y^{(1)}_{m,\bar{n}}). \tag{A.7}$$

Moreover, the fact that $C_{\bar{n}} = -C_n$, combined with Property A, implies that

$$L(C_n, -y^{(1)}_{m,\bar{n}}) = L(C_{\bar{n}}, y^{(1)}_{m,\bar{n}}). \tag{A.8}$$

It follows that $D^{(1)}_{m,n} \overset{(a)}{\leq} L(C_m \cup C_n, -y^{(1)}_{m,\bar{n}}) = L(C_m, -y^{(1)}_{m,\bar{n}}) + L(C_n, -y^{(1)}_{m,\bar{n}}) \overset{(b)}{\leq} L(C_m, y^{(1)}_{m,\bar{n}}) + L(C_{\bar{n}}, y^{(1)}_{m,\bar{n}}) = L(C_m \cup C_{\bar{n}}, y^{(1)}_{m,\bar{n}}) = D^{(1)}_{m,\bar{n}}$, where (a) follows from the

definition of $D^{(1)}_{m,n}$, while (b) is based on (A.7) and (A.8).

Consider now the case $y^{(1)}_{m,\bar{n}} < 0$. Since $C_n \subseteq (-\infty, 0]$, by using Property B followed by Property A, one obtains $L(C_n, y^{(1)}_{m,\bar{n}}) \leq L(C_n, -y^{(1)}_{m,\bar{n}}) = L(C_{\bar{n}}, y^{(1)}_{m,\bar{n}})$. Then the following sequence of relations holds

$$\begin{aligned} D^{(1)}_{m,n} &\leq L(C_m \cup C_n, y^{(1)}_{m,\bar{n}}) \\ &= L(C_m, y^{(1)}_{m,\bar{n}}) + L(C_n, y^{(1)}_{m,\bar{n}}) \leq L(C_m, y^{(1)}_{m,\bar{n}}) + L(C_{\bar{n}}, y^{(1)}_{m,\bar{n}}) = D^{(1)}_{m,\bar{n}}. \end{aligned}$$

Consider now the case $l = 3$. Since $C_m \subseteq (-\infty, 0]$, $y_n \leq 0$ and $y_{\bar{n}} = -y_n$, according to Property B, one has

$$L(C_m, y_n) \leq L(C_m, y_{\bar{n}}), \tag{A.9}$$

while Property A implies that

$$L(C_n, y_n) = L(C_{\bar{n}}, y_{\bar{n}}). \tag{A.10}$$

Applying again Properties B and A leads to

$$L(C_n, y_m) \leq L(C_n, -y_m) = L(C_{\bar{n}}, y_m). \tag{A.11}$$

Further, relations (A.9), (A.10) and (A.11) imply that $D^{(3)}_{m,n} - D^{(3)}_{m,\bar{n}} = 0.5(L(C_m, y_m) + L(C_m, y_n) + L(C_n, y_m) + L(C_n, y_n)) - 0.5(L(C_m, y_m) + L(C_m, y_{\bar{n}}) + L(C_{\bar{n}}, y_m) + L(C_{\bar{n}}, y_{\bar{n}})) \leq 0$. This completes the proof of the lemma. □

Lemma 4. *Assume that $f(x)$ and the quantizer Q are symmetric about 0 and*

$y_i \geq 2\mu_i$, *for* $i \in \mathcal{I}$. *Let* $d(x,y) = (x-y)^2$. *Then,* $w^{(2)}_{m,n} \leq w^{(2)}_{m,\bar{n}}$ *and* $w^{(2)}_{m,n} \leq w^{(2)}_{n,\bar{m}}$ *for all* $0 \leq m < n \leq M/2-1$.

Proof. As in the proof of Lemma 3, we only need to show that $D^{(2)}_{m,n} \leq D^{(2)}_{m,\bar{n}}$. Note that, since $y_{\bar{n}} = -y_n$ and $C_{\bar{n}} = -C_n$, using the definition of $D^{(2)}_{m,\bar{n}}$ and Property A, one obtains

$$\begin{aligned} D^{(2)}_{m,\bar{n}} &= \int_{C_m} \left(x - \tfrac{y_m + y_{\bar{n}}}{2}\right)^2 f(x)\, dx + \int_{C_{\bar{n}}} \left(x - \tfrac{y_m + y_{\bar{n}}}{2}\right)^2 f(x)\, dx \\ &= \int_{C_m} \left(x - \tfrac{y_m - y_n}{2}\right)^2 f(x)\, dx + \int_{C_n} \left(x + \tfrac{y_m - y_n}{2}\right)^2 f(x)\, dx. \end{aligned}$$

Combining the above with the definition of $D^{(2)}_{m,n}$ and using straightforward algebra, leads to $D^{(2)}_{m,n} - D^{(2)}_{m,\bar{n}} = -y_n P(C_m)\,(2\mu_m - y_m) - y_m P(C_n)\,(2\mu_n - y_n) \leq 0$, where the last inequality follows from $y_n, y_m \leq 0$, $y_m \geq 2\mu_m$ and $y_n \geq 2\mu_n$. Now the proof is complete. □

Lemma 5. *Let* $l \in \{1,2,3\}$. *If* $w^{(l)}_{m,n} \leq w^{(l)}_{m,\bar{n}}$ *and* $w^{(l)}_{m,n} \leq w^{(l)}_{n,\bar{m}}$ *for all* $0 \leq m < n \leq M/2-1$, *then* $F_2(\mathbf{z}^{(1)}) \geq F_1(\mathbf{x}^{(1)})$.

Proof. Recall that F_1 and F_2 denote the cost functions of problems (2.10) and (2.11), respectively. Using the definition of $\mathbf{z}^{(1)}$, one obtains

$$F_2(\mathbf{z}^{(1)}) = \sum_{i=0}^{M/2-1} \left(\sum_{j=M/2}^{\bar{i}-1} w^{(l)}_{i,j} \left(x^{(1)}_{i,j} + x^{(1)}_{i,\bar{j}}\right) + w^{(l)}_{i,\bar{i}} x^{(1)}_{i,\bar{i}} \quad + \sum_{j=\bar{i}+1}^{M-1} w^{(l)}_{i,j} \left(x^{(1)}_{i,j} + x^{(1)}_{\bar{j},i}\right) \right). \tag{A.12}$$

According to the hypothesis, one has $w_{i,j}^{(l)} \geq w_{i,\bar{j}}^{(l)}$ when $M/2 \leq j < \bar{i}$, and $w_{i,j}^{(l)} \geq w_{\bar{j},i}^{(l)}$ when $M/2 \leq \bar{i} < j \leq M-1$. Plugging these in (A.12) leads to

$$\begin{aligned} F_2(\mathbf{z}^{(1)}) &\geq \sum_{i=0}^{M/2-1} \left(\sum_{j=M/2}^{\bar{i}-1} w_{i,\bar{j}}^{(l)} x_{i,\bar{j}}^{(1)} + \sum_{j=M/2}^{\bar{i}-1} w_{i,j}^{(l)} x_{i,j}^{(1)} + w_{i,\bar{i}}^{(l)} x_{i,\bar{i}}^{(1)} + \sum_{j=\bar{i}+1}^{M-1} w_{i,j}^{(l)} x_{i,j}^{(1)} + \sum_{j=\bar{i}+1}^{M-1} w_{\bar{j},i}^{(l)} x_{\bar{j},i}^{(1)} \right) \\ &= \sum_{i=0}^{M/2-1} \left(\sum_{j=i+1}^{M-1} w_{i,j}^{(l)} x_{i,j}^{(1)} + \sum_{k=0}^{i-1} w_{k,i}^{(l)} x_{k,i}^{(1)} \right), \end{aligned} \tag{A.13}$$

where the last equality is obtained in the same manner as (A.2). Further, relation (2.15) and Lemma 1 imply that $w_{k,i}^{(l)} x_{k,i}^{(1)} = w_{\bar{i},\bar{k}}^{(l)} x_{\bar{i},\bar{k}}^{(1)}$. By applying this in (A.13), it follows that

$$\begin{aligned} F_2(\mathbf{z}^{(1)}) &\geq \sum_{i=0}^{M/2-1} \sum_{j=i+1}^{M-1} w_{i,j}^{(l)} x_{i,j}^{(1)} + \sum_{i=0}^{M/2-1} \sum_{k=0}^{i-1} w_{\bar{i},\bar{k}}^{(l)} x_{\bar{i},\bar{k}}^{(1)} \\ &= \sum_{i=0}^{M/2-1} \sum_{j=i+1}^{M-1} w_{i,j}^{(l)} x_{i,j}^{(1)} + \sum_{m=M/2}^{M-1} \sum_{j=m+1}^{M-1} w_{m,j}^{(l)} x_{m,j}^{(1)} = F_1(\mathbf{x}^{(1)}), \end{aligned} \tag{A.14}$$

where the first equality is obtained by making the substitutions $m = \bar{i}$ and $j = \bar{k}$ in the second nested sum. This observation completes the proof. □

Appendix B

Supplementary Materials for Chapter 3

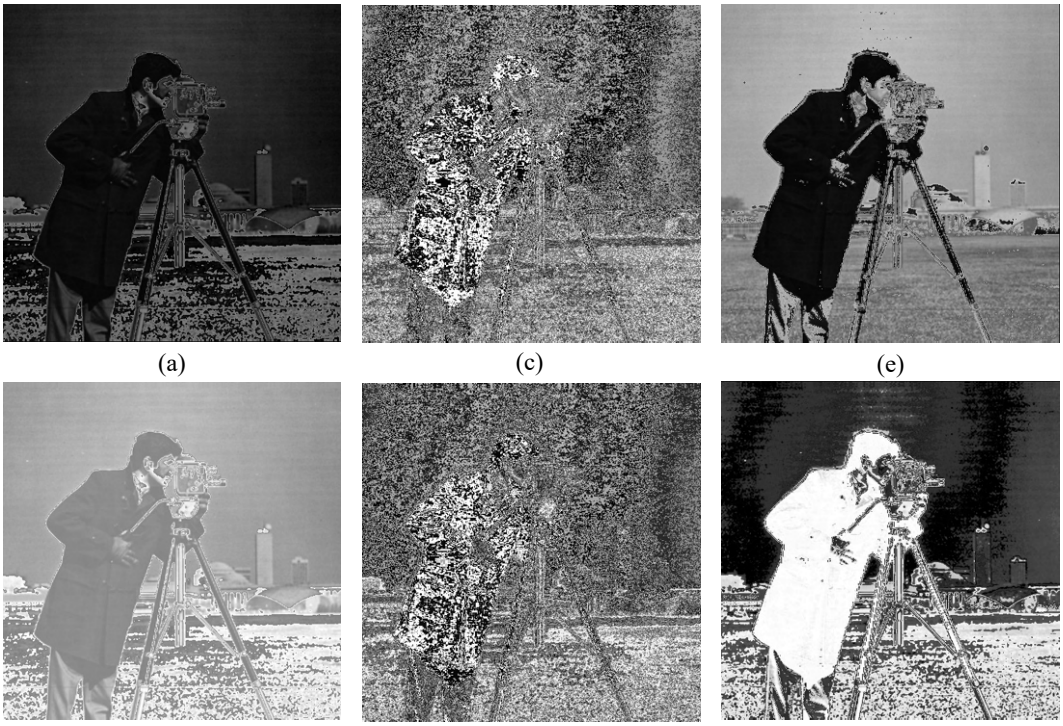


FIGURE A2.1: Eve's reconstruction of Camera Man in 1-bitplane encryption (a) NBC MSB0, (b) NBC MSB1, (c) OPTA MSB0, (d) OPTA MSB1, (e) OPTB_{10^6} MSB0, (f) OPTB_{10^6} MSB1.

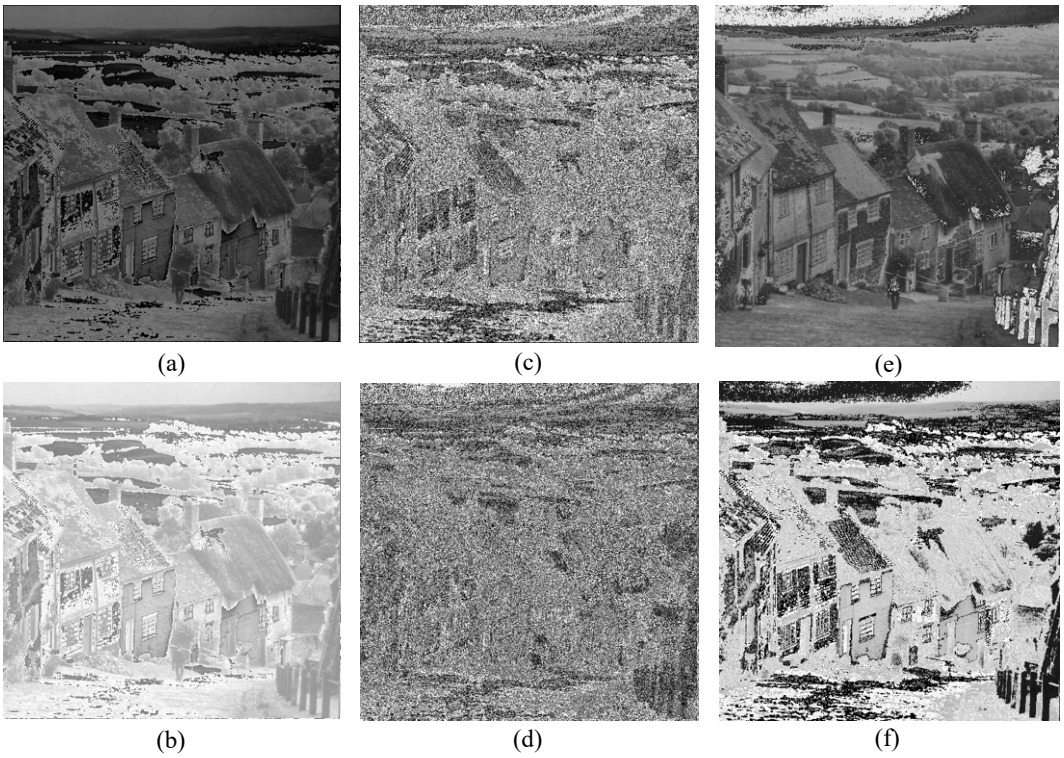


FIGURE A2.2: Eve's reconstruction of Gold Hill in 1-bitplane encryption (a) NBC MSB0, (b) NBC MSB1, (c) OPTA MSB0, (d) OPTA MSB1, (e) $OPTB_{10^6}$ MSB0, (f) $OPTB_{10^6}$ MSB1.

FIGURE A2.3: Eve's reconstruction of Living Room in 1-bitplane encryption (a) NBC MSB0, (b) NBC MSB1, (c) OPTA MSB0, (d) OPTA MSB1, (e) $OPTB_{10^6}$ MSB0, (f) $OPTB_{10^6}$ MSB1.

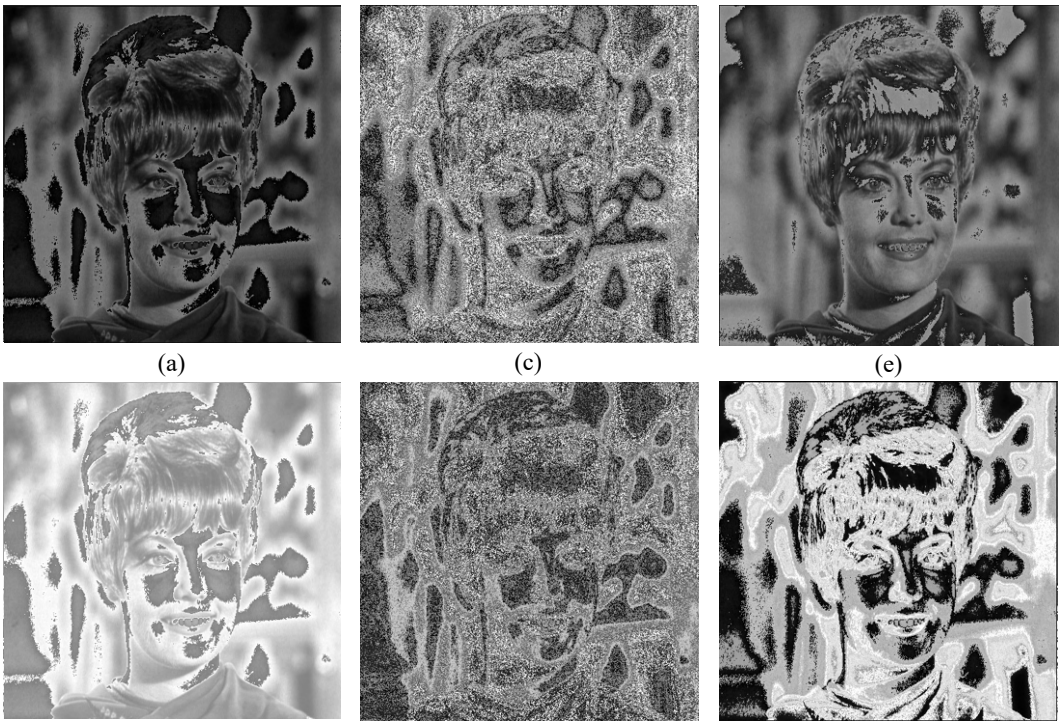


FIGURE A2.4: Eve's reconstruction of Zelda in 1-bitplane encryption (a) NBC MSB0, (b) NBC MSB1, (c) OPTA MSB0, (d) OPTA MSB1, (e) $OPTB_{10^6}$ MSB0, (f) $OPTB_{10^6}$ MSB1.

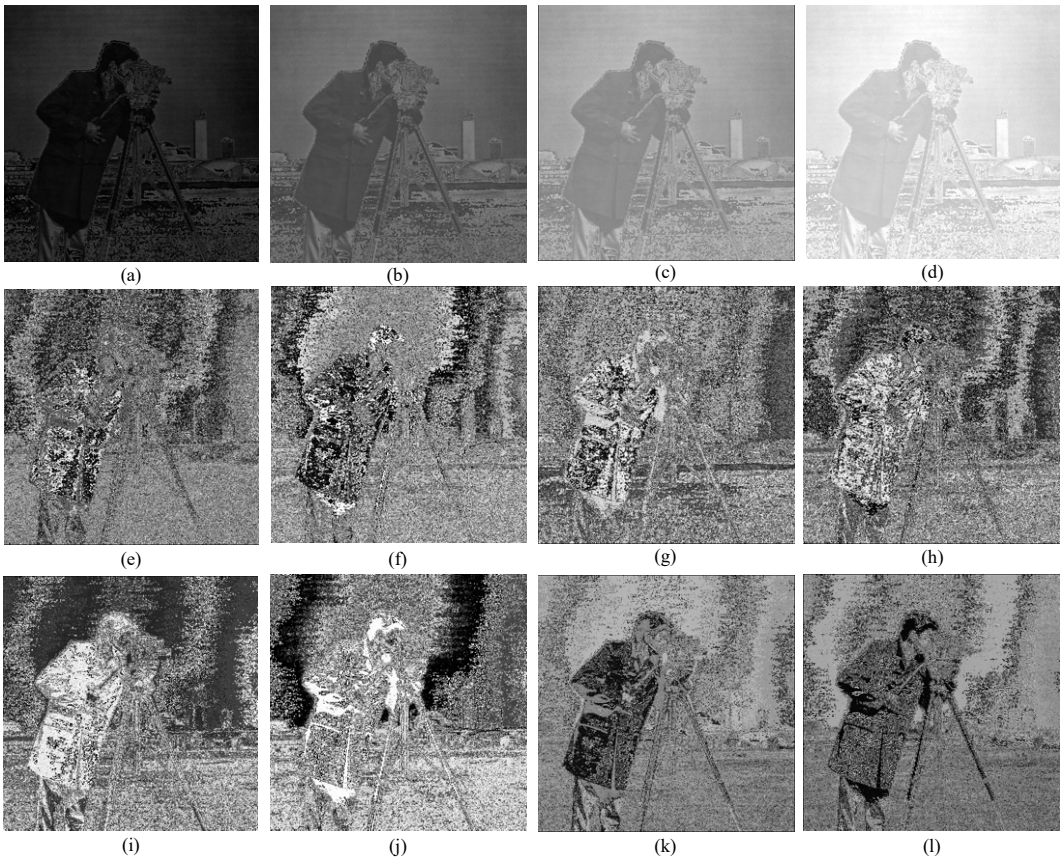


Figure A2.5: Eve's reconstruction of Camera Man under 2BPE (a) NBC 00, (b) NBC 01, (c) NBC 10, (d) NBC 11, (e) OPTA 00, (f) OPTA 01, (g) OPTA 10, (h) OPTA 11, (i) OPTB_{10^6} 00, (j) OPTB_{10^6} 01, (k) OPTB_{10^6} 10, (l) OPTB_{10^6} 11.

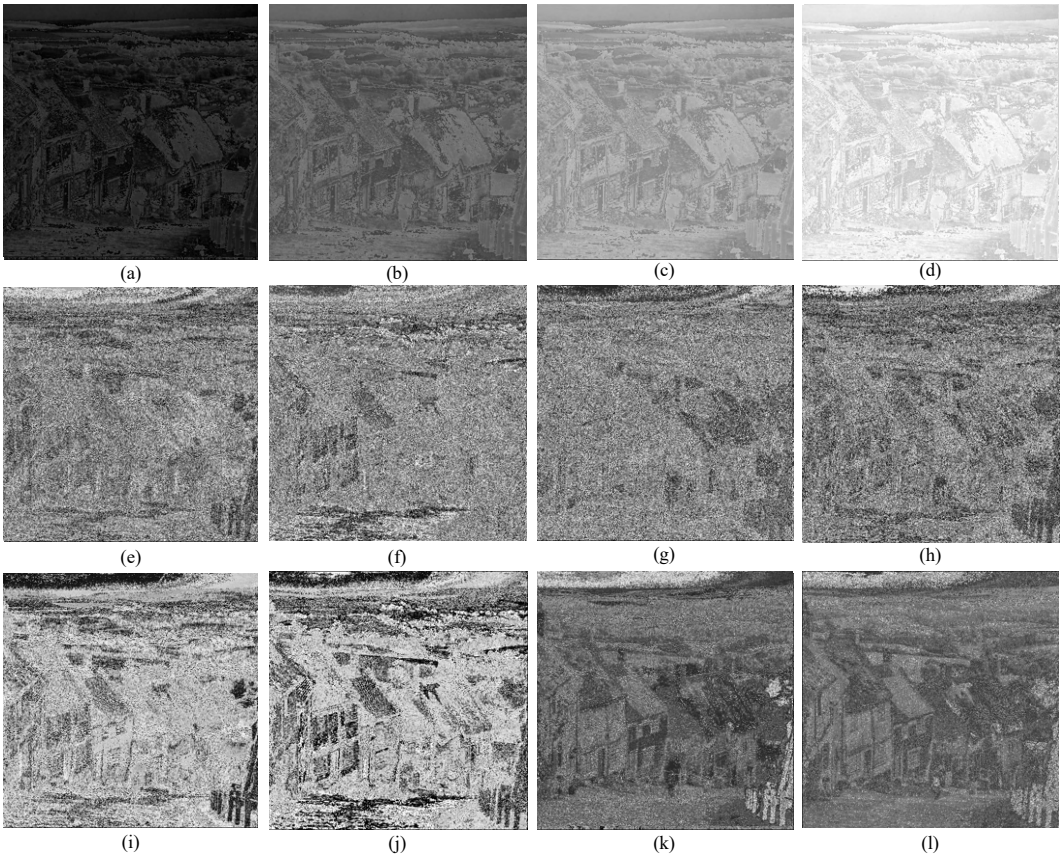


Figure A2.6: Eve's reconstruction of Gold Hill under 2BPE (a) NBC 00, (b) NBC 01, (c) NBC 10, (d) NBC 11, (e) OPTA 00, (f) OPTA 01, (g) OPTA 10, (h) OPTA 11, (i) OPTB_{10^6} 00, (j) OPTB_{10^6} 01, (k) OPTB_{10^6} 10, (l) OPTB_{10^6} 11.

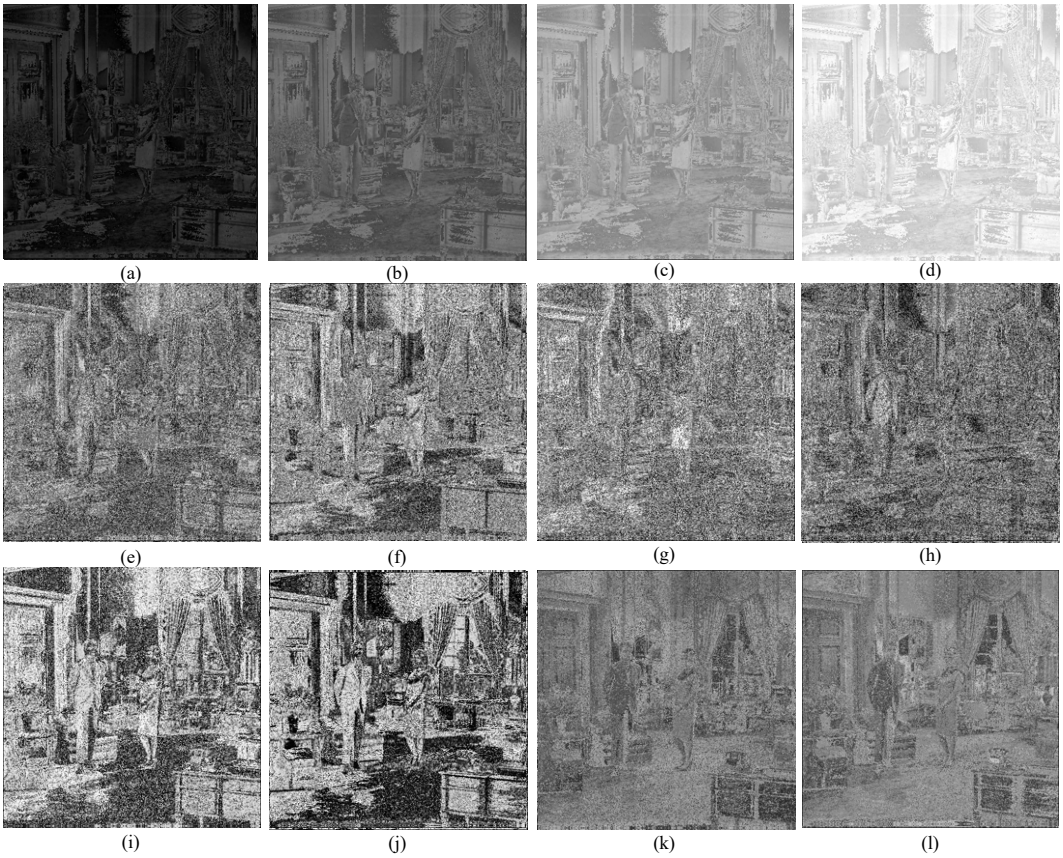


FIGURE A2.7: Eve's reconstruction of Living Room under 2BPE (a) NBC 00, (b) NBC 01, (c) NBC 10, (d) NBC 11, (e) OPTA 00, (f) OPTA 01, (g) OPTA 10, (h) OPTA 11, (i) OPTB_{10^6} 00, (j) OPTB_{10^6} 01, (k) OPTB_{10^6} 10, (l) OPTB_{10^6} 11.

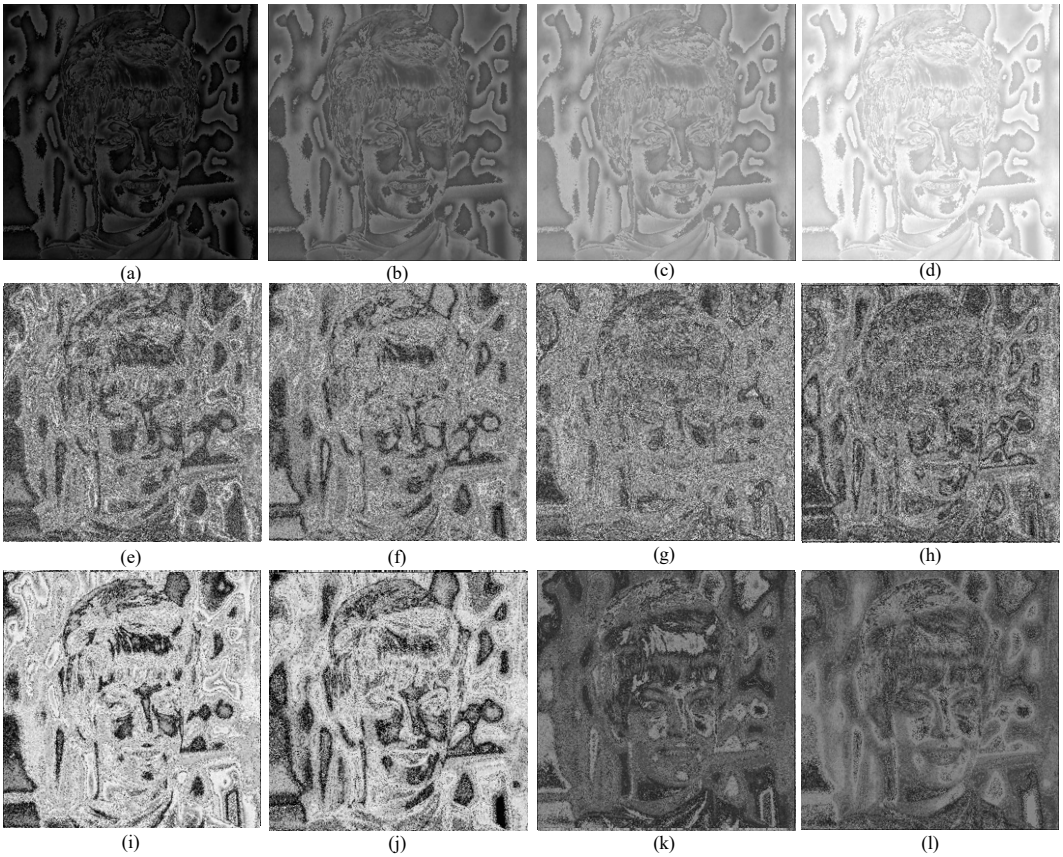
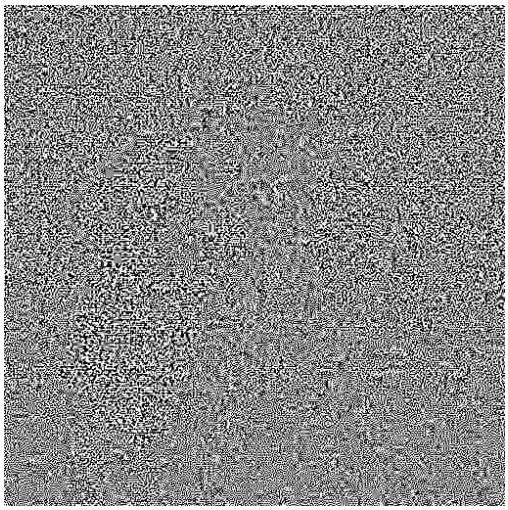


Figure A2.8: Eve's reconstruction of Zelda under 2BPE (a) NBC 00, (b) NBC 01, (c) NBC 10, (d) NBC 11, (e) OPTA 00, (f) OPTA 01, (g) OPTA 10, (h) OPTA 11, (i) OPTB_{10^6} 00, (j) OPTB_{10^6} 01, (k) OPTB_{10^6} 10, (l) OPTB_{10^6} 11.

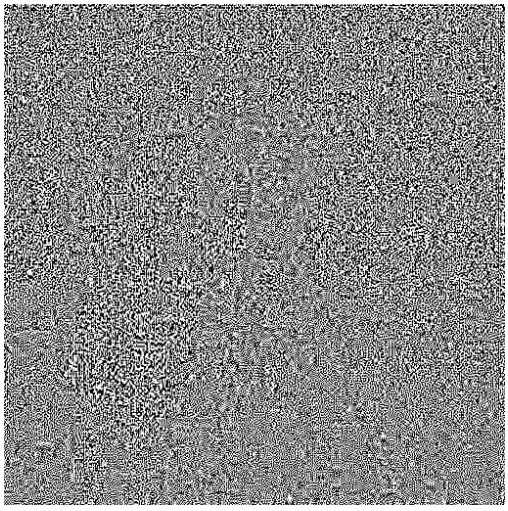
Appendix C

Supplementary Materials for Chapter 4

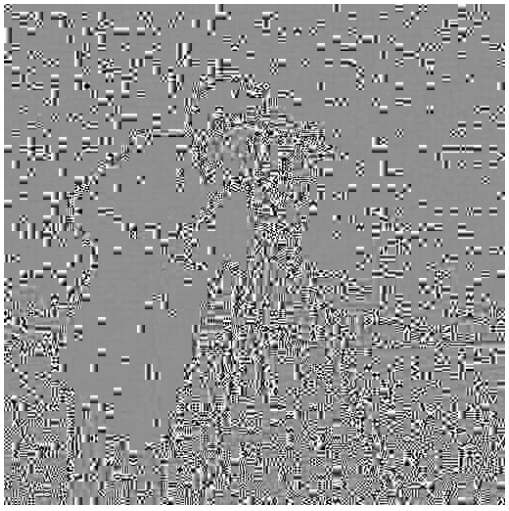


(A) Cameraman, MSB0, λ_{th}

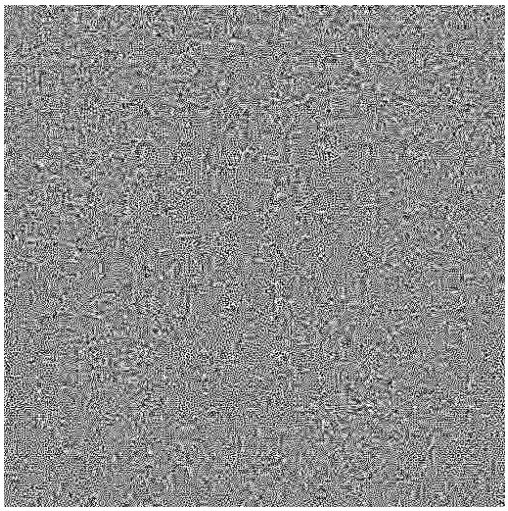
(B) Cameraman, MSB0, $\lambda_{th}+1$



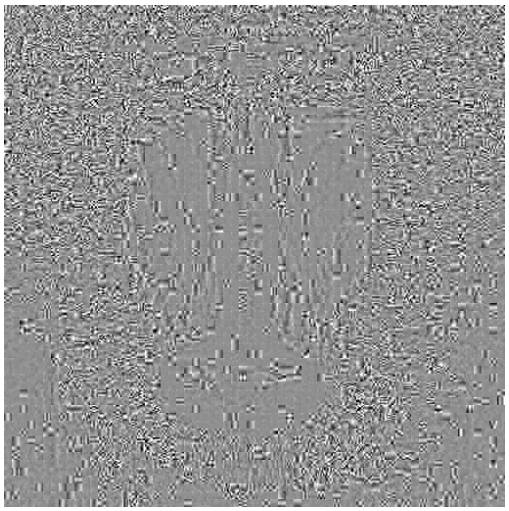
(C) Cameraman, MSB1, λ_{th}



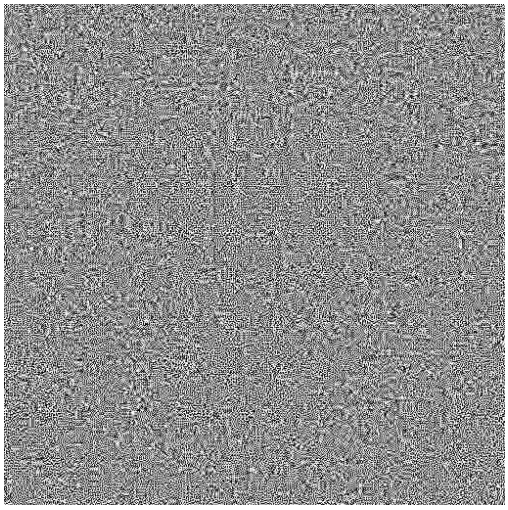
(D) Cameraman, MSB1, $\lambda_{th}+1$



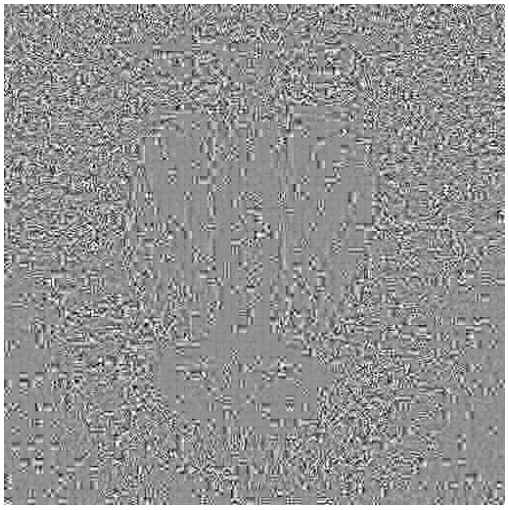
(E) Mandrill, MSB0, λ_{th}



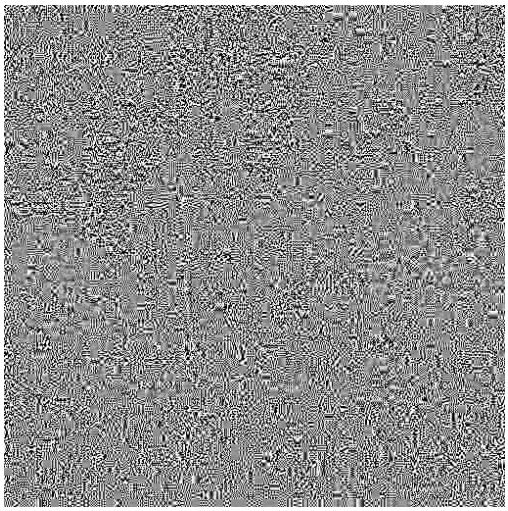
(F) Mandrill, MSB0, $\lambda_{th}+1$



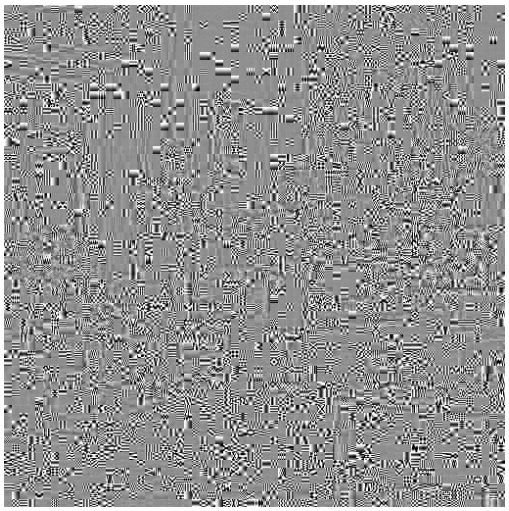
(G) Mandrill, MSB1, λ_{th}



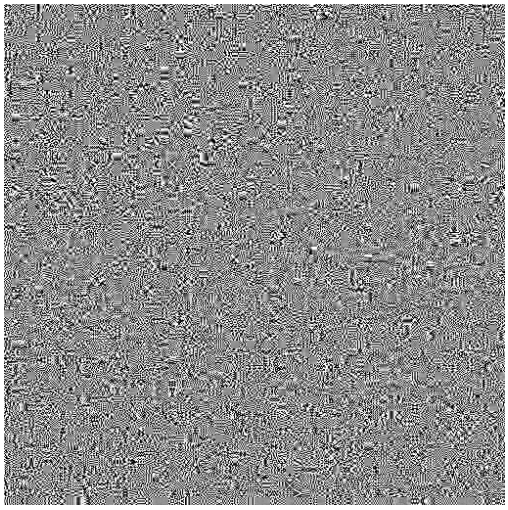
(H) Mandrill, MSB1, $\lambda_{th}+1$



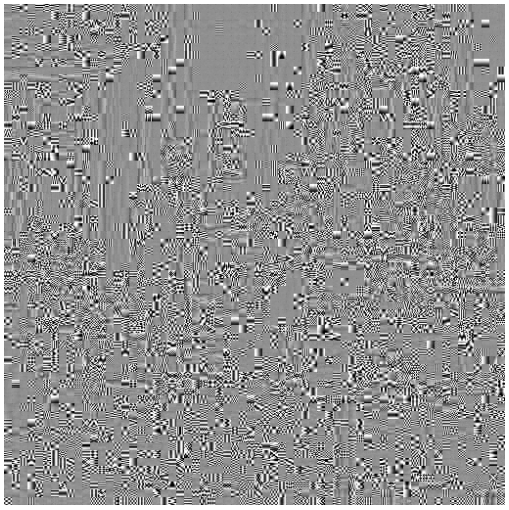
(I) Livingroom, MSB0, λ_{th}



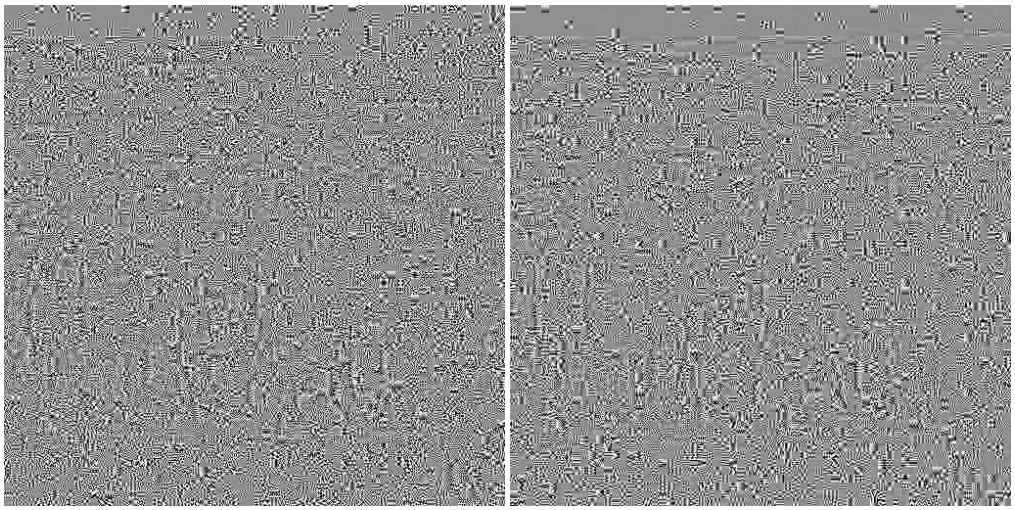
(J) Livingroom, MSB0, $\lambda_{th} + 1$



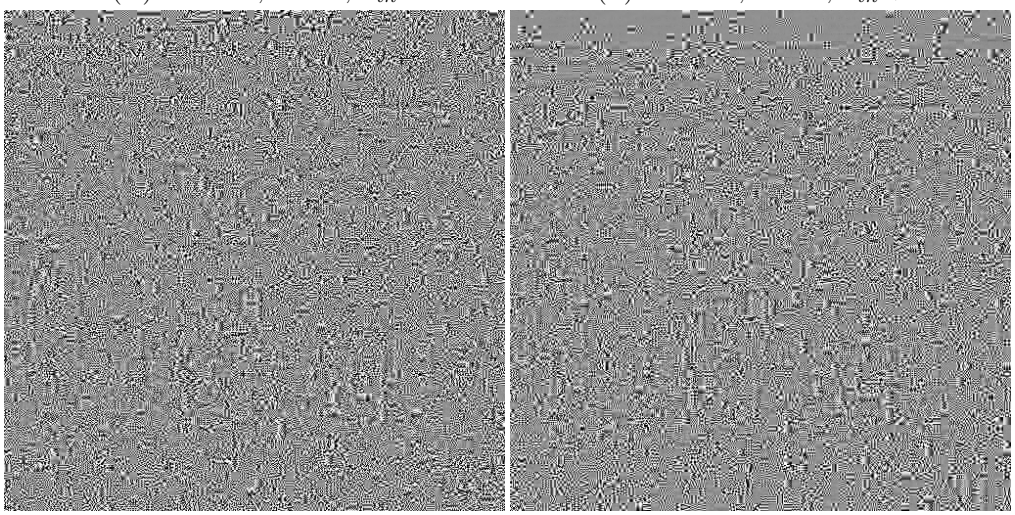
(K) Livingroom, MSB1, λ_{th}



(L) Livingroom, MSB1, $\lambda_{th} + 1$



(M) Goldhill, MSB0, λ_{th} (N) Goldhill, MSB0, $\lambda_{th}+1$



(O) Goldhill, MSB1, λ_{th} (P) Goldhill, MSB1, $\lambda_{th}+1$

FIGURE A3.0: Eve's reconstructions using the replacement attack, after SE with ECOptBC for λ_{th} and $\lambda_{th}+1$.

TABLE A3.1: PSNRs and SSIMs of Eve's reconstructions in SwapBC for Cameraman.

x		+1			+2			+3			+4		
s_H		8	9	10	8	9	10	8	9	10	8	9	10
PSNR	MSB0	8.93	7.20	6.11	10.18	8.60	6.96	10.20	8.74	7.23	10.83	10.14	8.73
	MSB1	9.14	7.31	6.16	10.40	8.92	7.36	10.34	8.71	6.98	11.02	10.44	9.48
SSIM	MSB0	−0.0005	−0.0024	−0.0013	0.0103	−0.0029	−0.0035	0.0318	0.0150	0.0127	0.1165	0.0642	0.0483
	MSB1	0.0003	−0.00006	−0.00007	0.0420	0.0229	0.0190	0.0146	−0.000004	−0.0020	0.2813	0.2586	0.2527

TABLE A3.2: PSNRs and SSIMs of Eve's reconstructions in SwapBC for Mandrill.

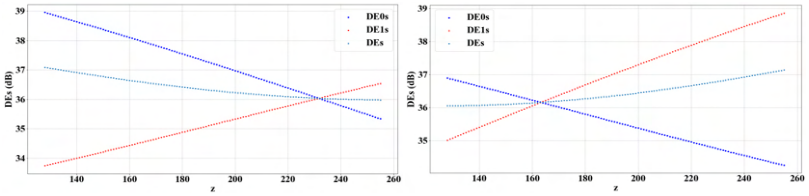
x		+1			+2			+3			+4		
s_H		8	9	10	8	9	10	8	9	10	8	9	10
PSNR	MSB0	10.74	8.33	6.91	12.29	9.79	7.72	12.29	9.85	7.82	13.36	11.61	9.21
	MSB1	10.95	8.43	6.95	12.60	10.15	8.01	12.61	10.10	7.89	13.46	11.72	9.57
SSIM	MSB0	−0.0051	−0.0031	−0.0012	−0.0072	−0.0083	−0.0050	−0.0073	−0.0091	−0.0059	0.0065	−0.0058	−0.0051
	MSB1	−0.0082	−0.0048	−0.0016	−0.0063	−0.0091	−0.0057	−0.0082	−0.0103	−0.0064	0.0103	0.0001	0.0023

TABLE A3.3: PSNRs and SSIMs of Eve's reconstructions in SwapBC for Livingroom.

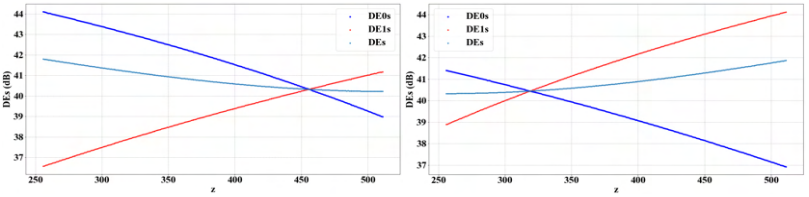
x		+1			+2			+3			+4		
s_H		8	9	10	8	9	10	8	9	10	8	9	10
PSNR	MSB0	11.34	9.16	7.46	11.12	8.85	7.17	11.16	8.96	7.36	12.16	10.19	8.04
	MSB1	11.45	9.22	7.46	11.24	9.00	7.36	11.24	8.91	7.19	12.34	10.62	8.86
SSIM	MSB0	0.0091	0.0003	−0.0006	−0.0058	−0.0088	−0.0070	−0.0005	−0.0049	−0.0042	−0.0020	−0.0140	−0.0124
	MSB1	0.0010	−0.0033	−0.0020	−0.0080	−0.0083	−0.0052	−0.0139	−0.0125	−0.0083	0.0309	0.0212	0.0223

Table A3.4: PSNRs and SSIMs of Eve's reconstructions in SwapBC for Goldhill.

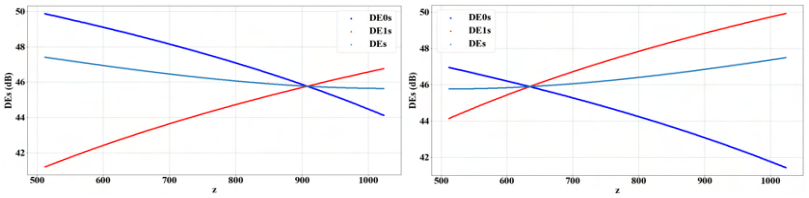
x		+1			+2			+3			+4		
s_H		8	9	10	8	9	10	8	9	10	8	9	10
PSNR	MSB0	9.96	7.99	6.68	10.38	8.34	6.86	10.36	8.35	6.86	11.30	9.57	7.69
	MSB1	10.08	8.05	6.69	10.49	8.41	6.88	10.50	8.42	6.88	11.43	9.85	8.24
SSIM	MSB0	0.0021	−0.0008	−0.0004	−0.0025	−0.0052	−0.0042	−0.0064	−0.0078	−0.0063	0.0074	−0.0032	−0.0027
	MSB1	0.0026	−0.0014	−0.0010	−0.0064	−0.0088	−0.0068	−0.0021	−0.0057	−0.0046	0.0580	0.0476	0.0479



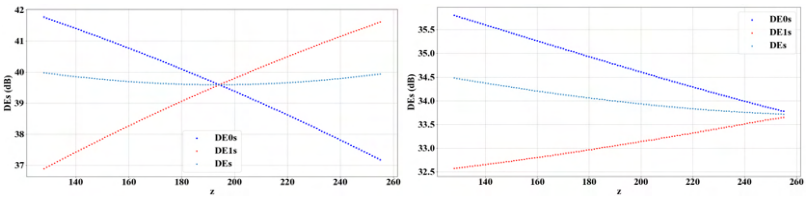
(A) $x = 2, s_H = 8$ (B) $x = 3, s_H = 8$



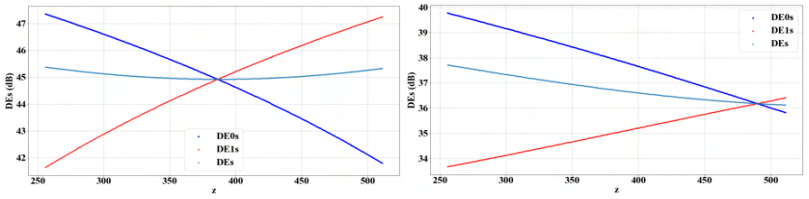
(C) $x = 2, s_H = 9$ (D) $x = 3, s_H = 9$



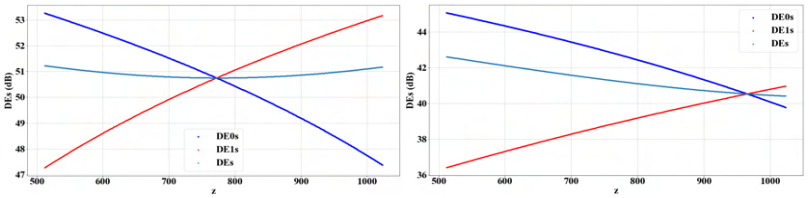
(E) $x = 2, s_H = 10$ (F) $x = 3, s_H = 10$



(G) $x = 1, s_H = 8$ (H) $x = 4, s_H = 8$

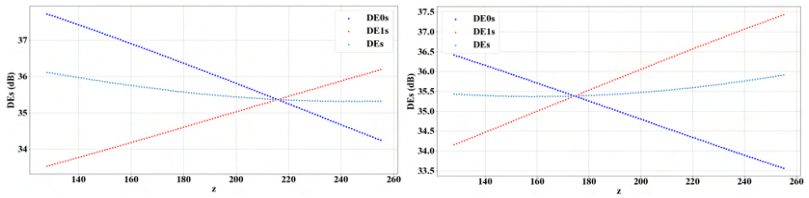


(I) $x = 1, s_H = 9$ (J) $x = 4, s_H = 9$

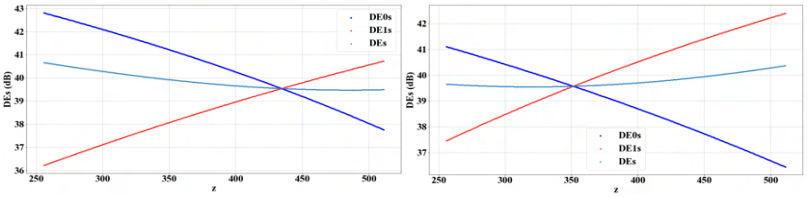


(K) $x = 1, s_H = 10$ (L) $x = 4, s_H = 10$

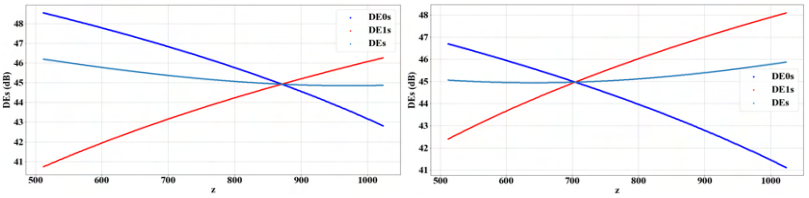
FIGURE A3.1: $D_{E,0}(x, z)$, $D_{E,1}(x, z)$, and $D_E(x, z)$ versus $z \in \mathcal{C}_+(s_H)$ for Cameraman.



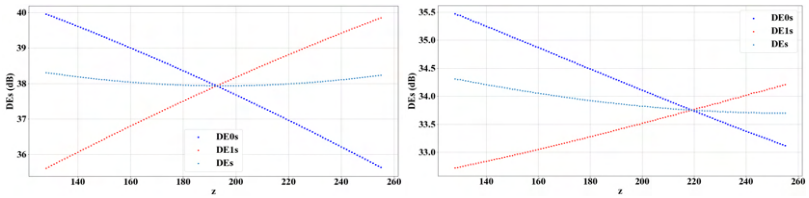
(A) $x = 2, s_H = 8$ (B) $x = 3, s_H = 8$



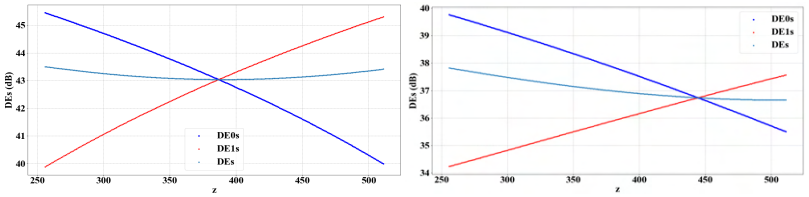
(C) $x = 2, s_H = 9$ (D) $x = 3, s_H = 9$



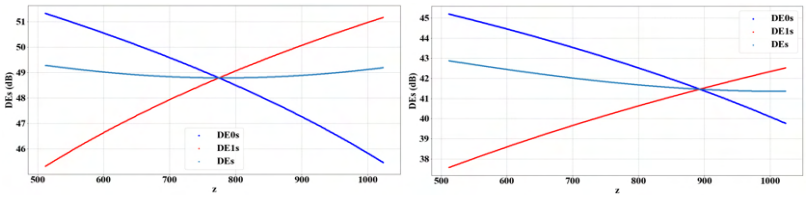
(E) $x = 2, s_H = 10$ (F) $x = 3, s_H = 10$



(G) $x = 1, s_H = 8$ (H) $x = 4, s_H = 8$

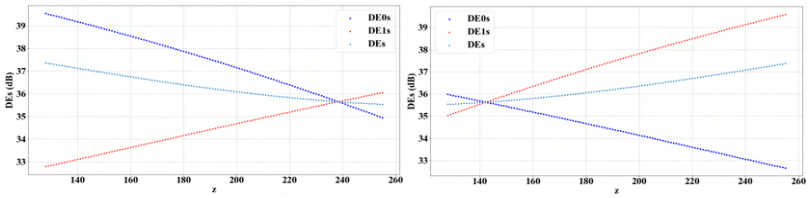


(I) $x = 1, s_H = 9$ (J) $x = 4, s_H = 9$

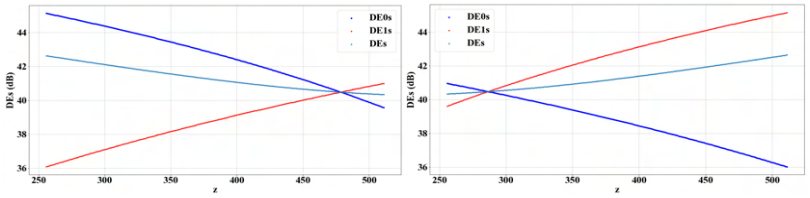


(K) $x = 1, s_H = 10$ (L) $x = 4, s_H = 10$

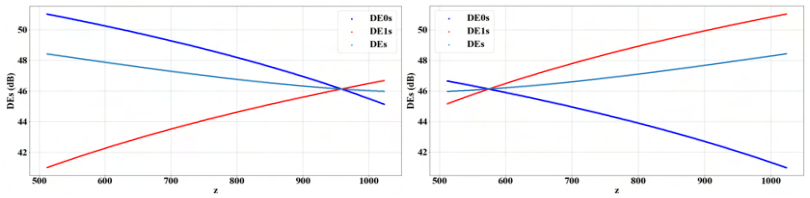
FIGURE A3.2: $D_{E,0}(x, z)$, $D_{E,1}(x, z)$, and $D_E(x, z)$ versus $z \in \mathcal{C}_+(s_H)$ for Mandrill.



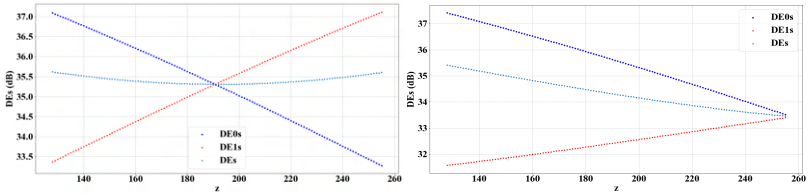
(A) $x = 2, s_H = 8$ (B) $x = 3, s_H = 8$



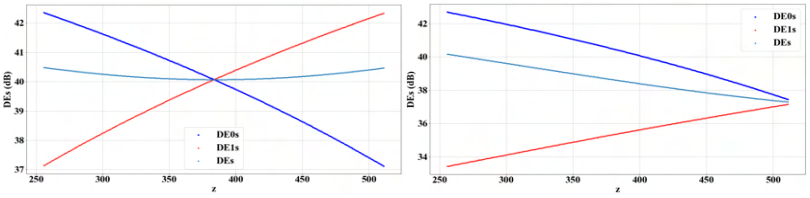
(C) $x = 2, s_H = 9$ (D) $x = 3, s_H = 9$



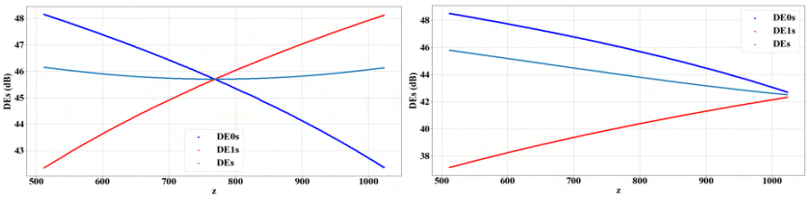
(E) $x = 2, s_H = 10$ (F) $x = 3, s_H = 10$



(G) $x = 1, s_H = 8$ (H) $x = 4, s_H = 8$

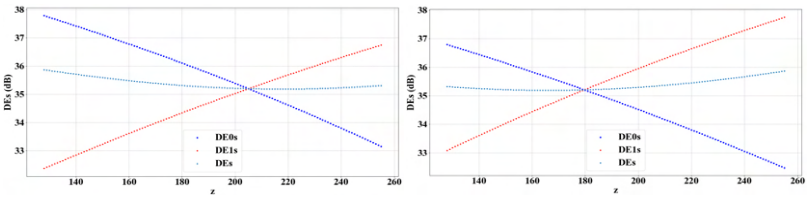


(I) $x = 1, s_H = 9$ (J) $x = 4, s_H = 9$

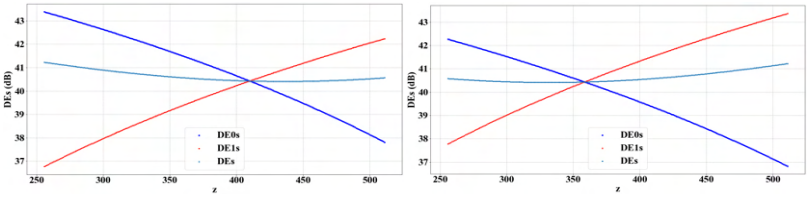


(K) $x = 1, s_H = 10$ (L) $x = 4, s_H = 10$

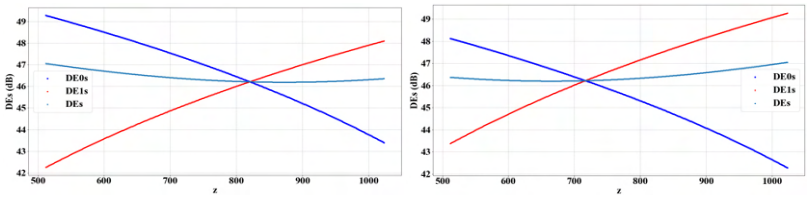
FIGURE A3.3: $D_{E,0}(x, z)$, $D_{E,1}(x, z)$, and $D_E(x, z)$ versus $z \in \mathcal{C}_+(s_H)$ for Livingroom.



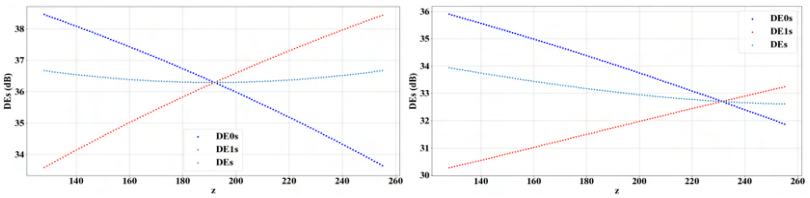
(A) $x = 2, s_H = 8$ (B) $x = 3, s_H = 8$



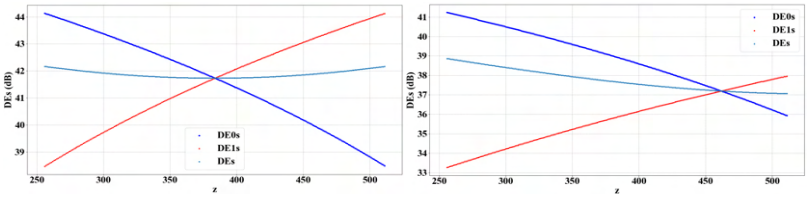
(C) $x = 2, s_H = 9$ (D) $x = 3, s_H = 9$



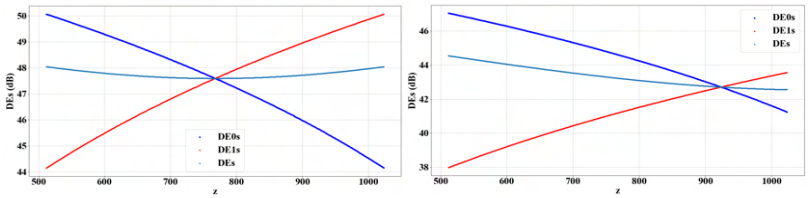
(E) $x = 2, s_H = 10$ (F) $x = 3, s_H = 10$



(G) $x = 1, s_H = 8$ (H) $x = 4, s_H = 8$



(I) $x = 1, s_H = 9$ (J) $x = 4, s_H = 9$



(K) $x = 1, s_H = 10$ (L) $x = 4, s_H = 10$

FIGURE A3.4: $D_{E,0}(x, z)$, $D_{E,1}(x, z)$, and $D_E(x, z)$ versus $z \in \mathcal{C}_+(s_H)$ for Goldhill.

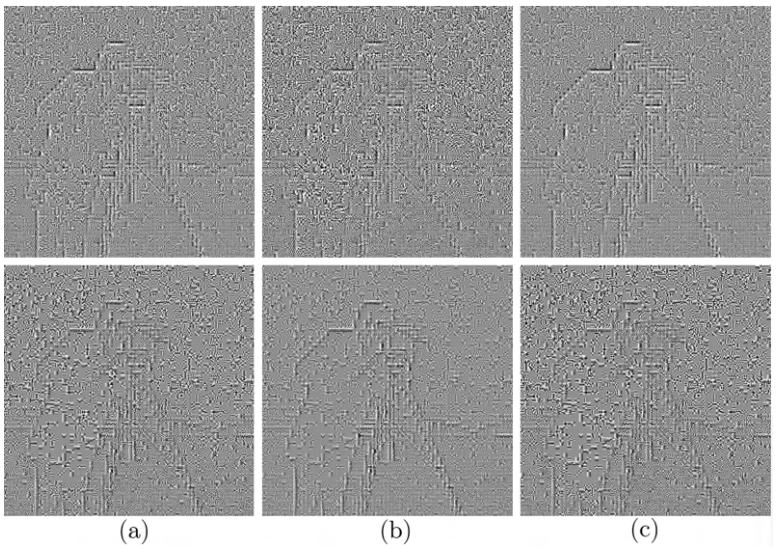


FIGURE A3.5: Comparison of the levels of degradation at Eve's under the MSB0 and MSB1 attacks for SwapBC for Cameraman, when $x = 2$, $s_H = 9$, and z is equal to a) z_{eq}, b) $\alpha(s_H)$, and c) $\beta(s_H)$. Top: MSB0 attack; bottom: MSB1 attack.

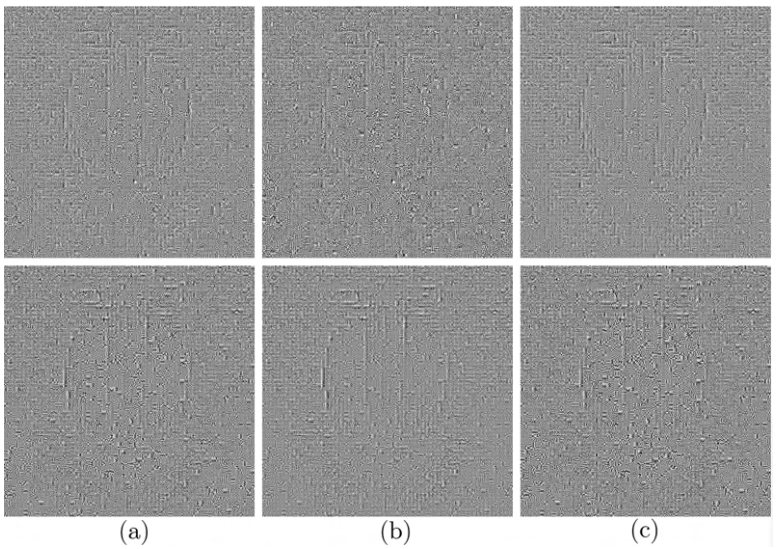


FIGURE A3.6: Comparison of the levels of degradation at Eve's under the MSB0 and MSB1 attacks for SwapBC for Mandrill, when $x = 2$, $s_H = 9$, and z is equal to a) z_{eq}, b) $\alpha(s_H)$, and c) $\beta(s_H)$. Top: MSB0 attack; bottom: MSB1 attack.

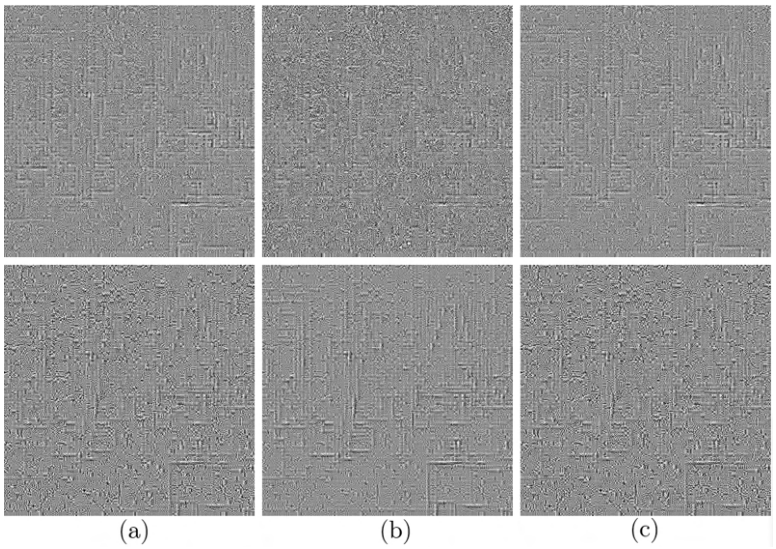


FIGURE A3.7: Comparison of the levels of degradation at Eve's under the MSB0 and MSB1 attacks for SwapBC for Livingroom, when $x = 2$, $s_H = 9$, and z is equal to a) z_{eq}, b) $\alpha(s_H)$, and c) $\beta(s_H)$. Top: MSB0 attack; bottom: MSB1 attack.

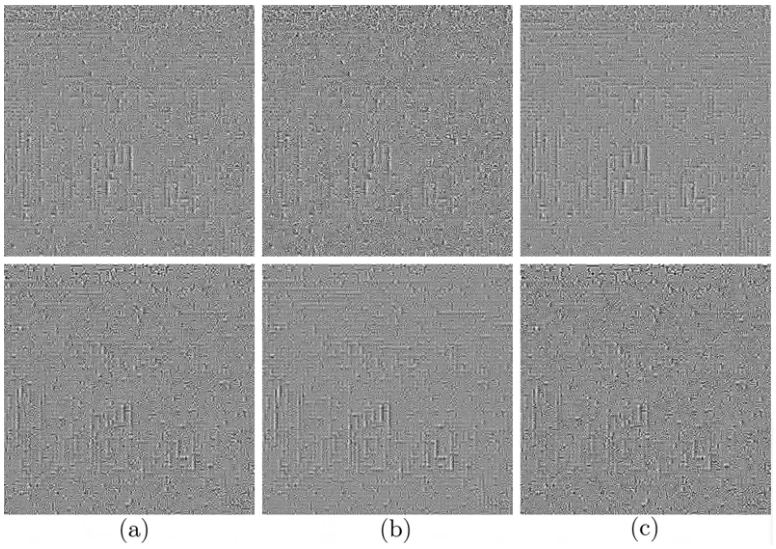
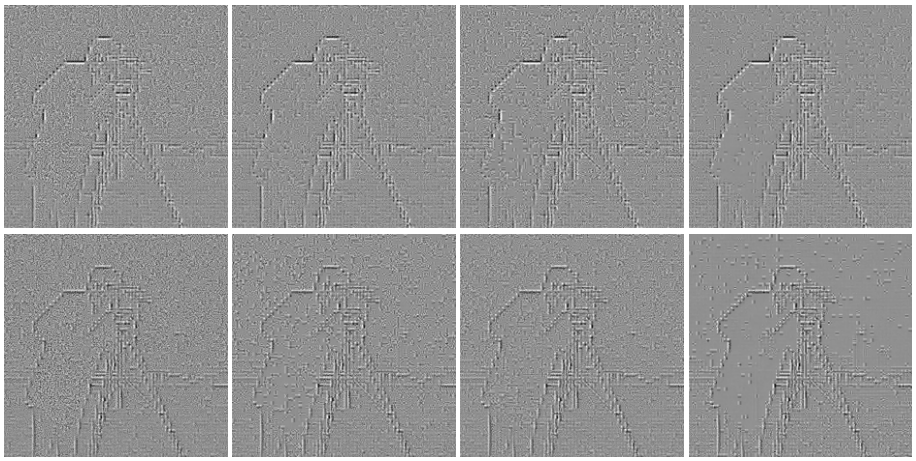
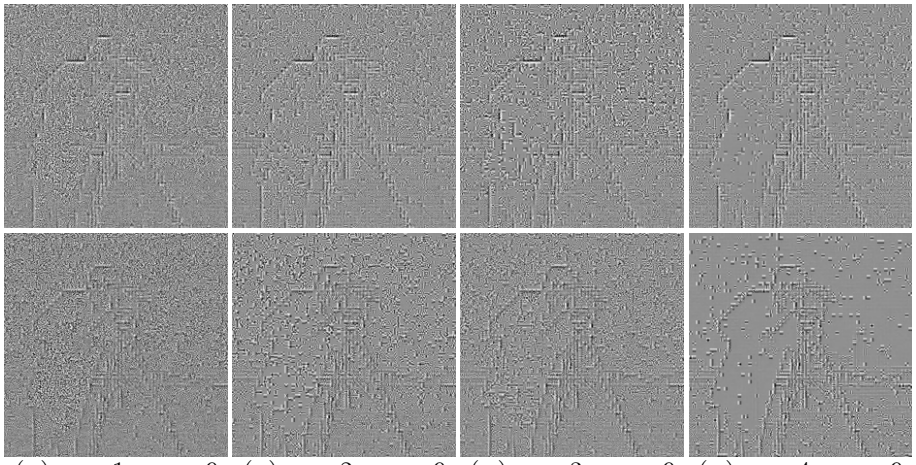


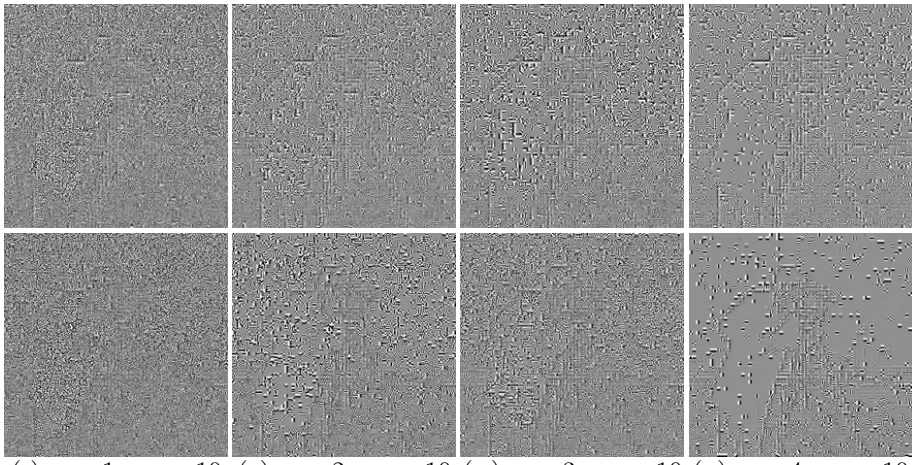
FIGURE A3.8: Comparison of the levels of degradation at Eve's under the MSB0 and MSB1 attacks for SwapBC for Goldhill, when $x = 2$, $s_H = 9$, and z is equal to a) z_{eq}, b) $\alpha(s_H)$, and c) $\beta(s_H)$. Top: MSB0 attack; bottom: MSB1 attack.



(A) $x = 1, s_H = 8$ (B) $x = 2, s_H = 8$ (C) $x = 3, s_H = 8$ (D) $x = 4, s_H = 8$

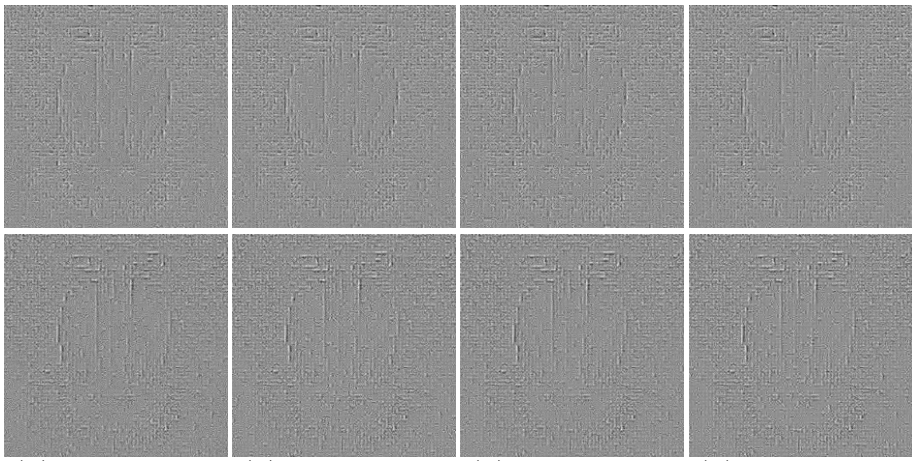


(E) $x = 1, s_H = 9$ (F) $x = 2, s_H = 9$ (G) $x = 3, s_H = 9$ (H) $x = 4, s_H = 9$

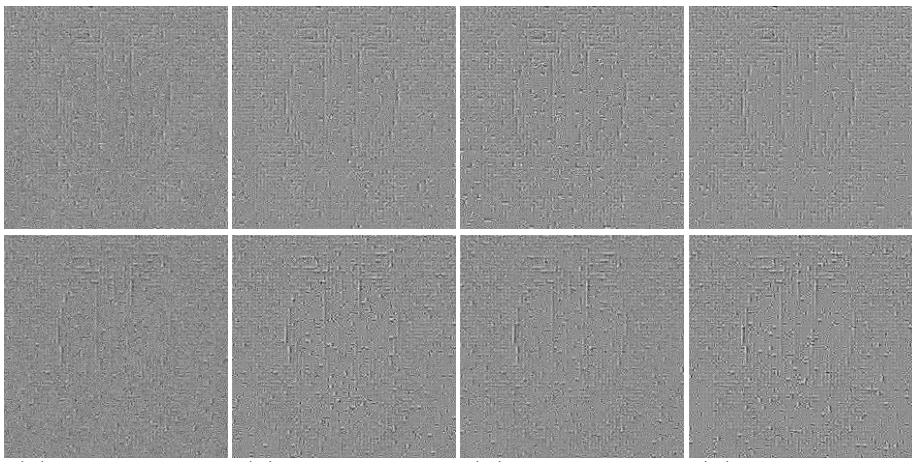


(I) $x = 1, s_H = 10$ (J) $x = 2, s_H = 10$ (K) $x = 3, s_H = 10$ (L) $x = 4, s_H = 10$

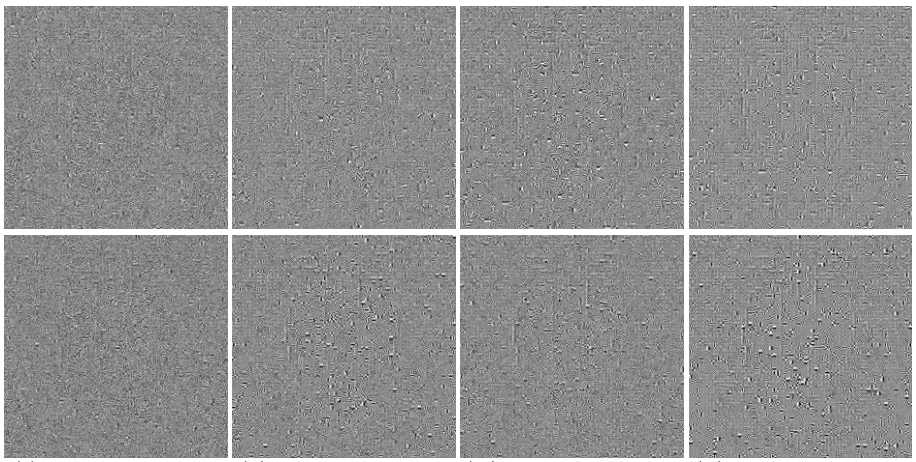
FIGURE A3.9: Eve's reconstructions for Cameraman after SE with SwapBC when $x = 1, 2, 3, 4$, z_{eq} and $s_H = 8, 9, 10$. Top: MSB0 attack; bottom MSB1 attack.



(A) $x = 1, s_H = 8$ (B) $x = 2, s_H = 8$ (C) $x = 3, s_H = 8$ (D) $x = 4, s_H = 8$

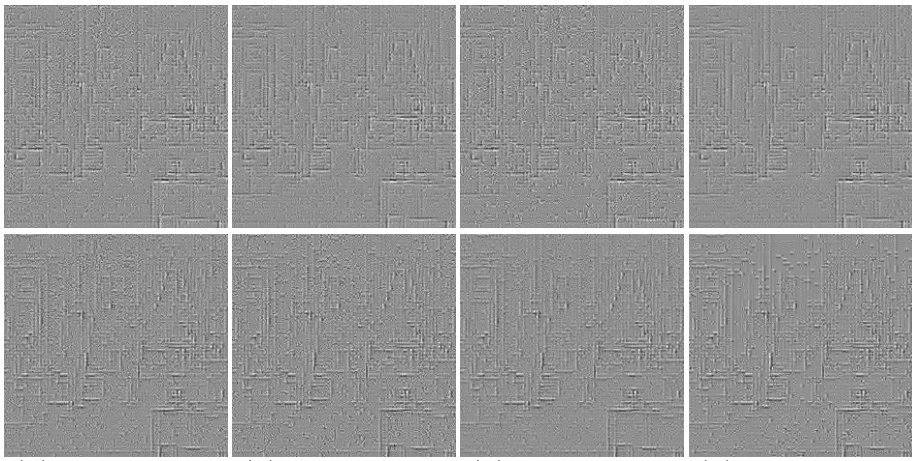


(E) $x = 1, s_H = 9$ (F) $x = 2, s_H = 9$ (G) $x = 3, s_H = 9$ (H) $x = 4, s_H = 9$

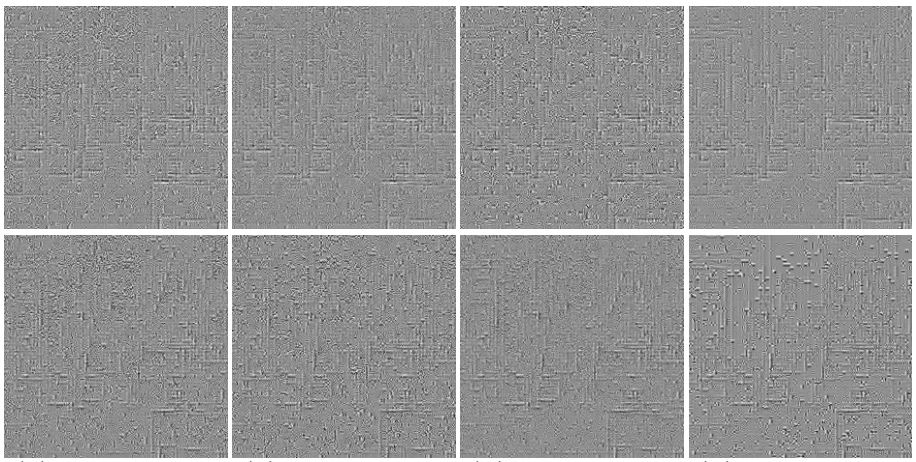


(I) $x = 1, s_H = 10$ (J) $x = 2, s_H = 10$ (K) $x = 3, s_H = 10$ (L) $x = 4, s_H = 10$

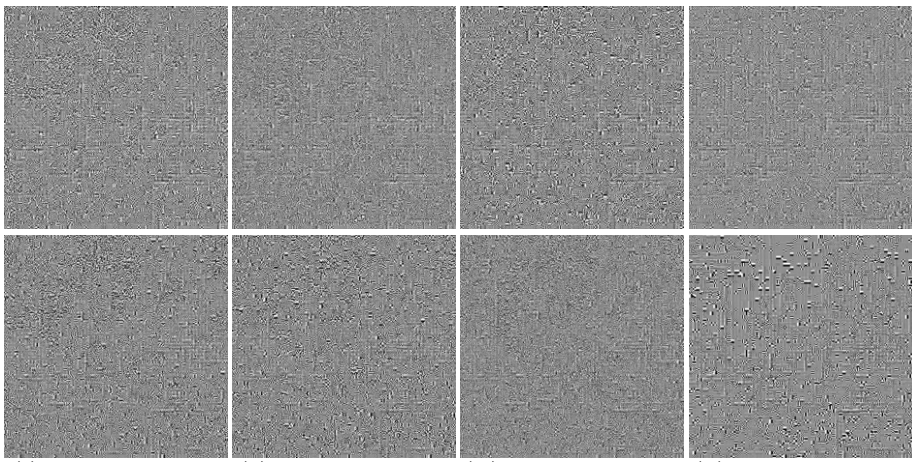
FIGURE A3.10: Eve's reconstructions for Mandrill after SE with SwapBC when $x = 1, 2, 3, 4$, z_{eq} and $s_H = 8, 9, 10$. Top: MSB0 attack; bottom MSB1 attack.



(A) $x = 1, s_H = 8$ (B) $x = 2, s_H = 8$ (C) $x = 3, s_H = 8$ (D) $x = 4, s_H = 8$

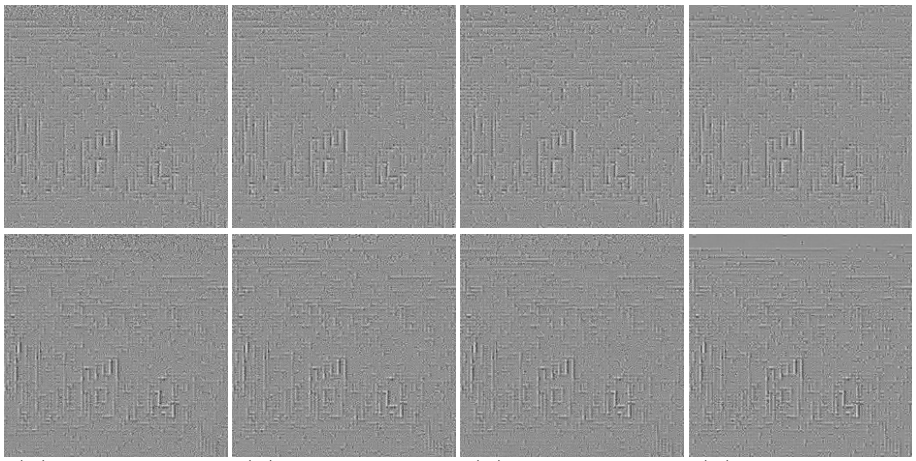


(E) $x = 1, s_H = 9$ (F) $x = 2, s_H = 9$ (G) $x = 3, s_H = 9$ (H) $x = 4, s_H = 9$

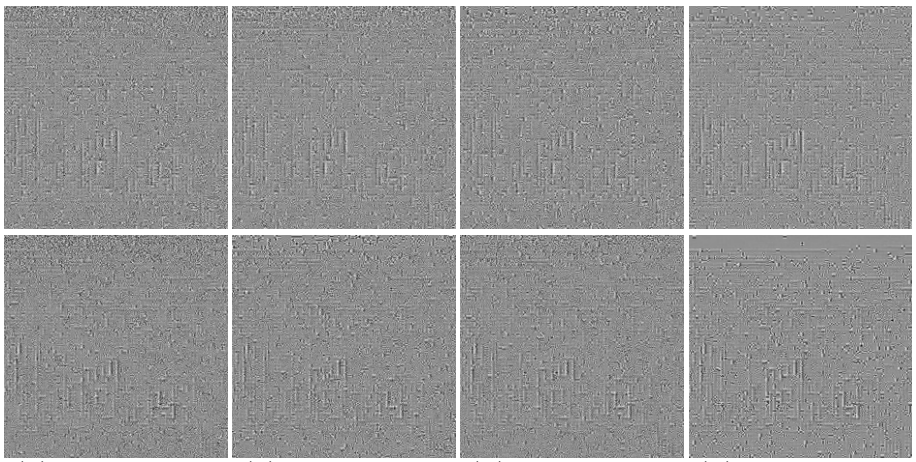


(I) $x = 1, s_H = 10$ (J) $x = 2, s_H = 10$ (K) $x = 3, s_H = 10$ (L) $x = 4, s_H = 10$

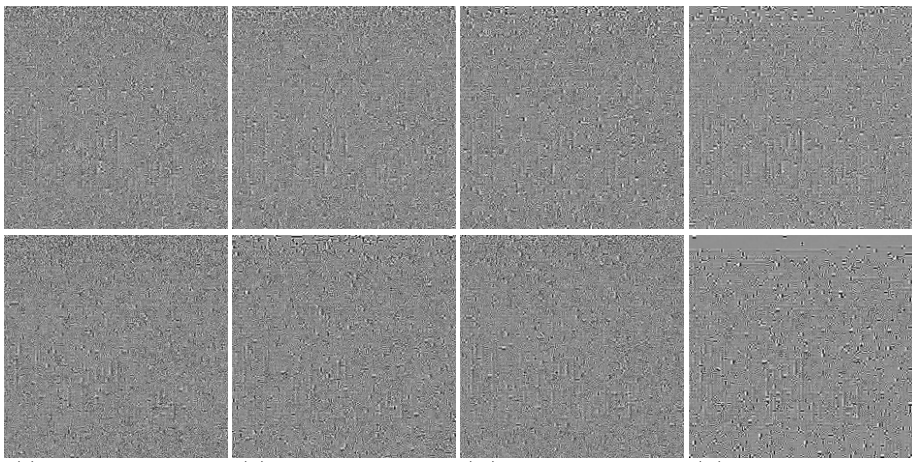
FIGURE A3.11: Eve's reconstructions for Livingroom after SE with SwapBC when $x = 1, 2, 3, 4$, z_{eq} and $s_H = 8, 9, 10$. Top: MSB0 attack; bottom MSB1 attack.



(A) $x = 1, s_H = 8$ (B) $x = 2, s_H = 8$ (C) $x = 3, s_H = 8$ (D) $x = 4, s_H = 8$

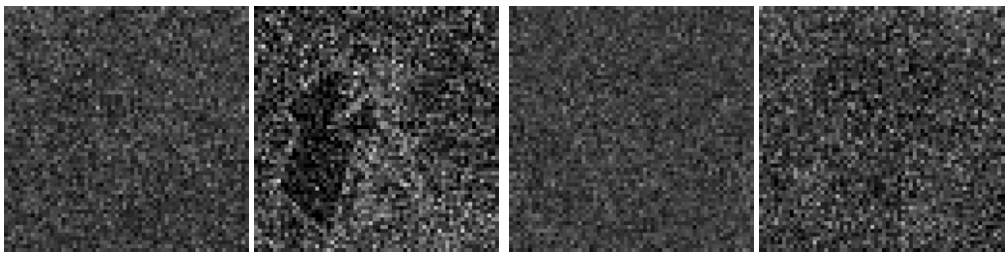


(E) $x = 1, s_H = 9$ (F) $x = 2, s_H = 9$ (G) $x = 3, s_H = 9$ (H) $x = 4, s_H = 9$

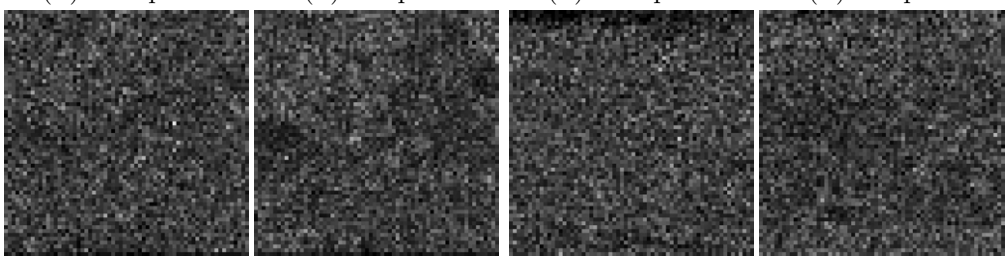


(I) $x = 1, s_H = 10$ (J) $x = 2, s_H = 10$ (K) $x = 3, s_H = 10$ (L) $x = 4, s_H = 10$

FIGURE A3.12: Eve's reconstructions for Goldhill after SE with SwapBC when $x = 1, 2, 3, 4$, z_{eq} and $s_H = 8, 9, 10$. Top: MSB0 attack; bottom MSB1 attack.



(A) ECOptBC (B) SwapBC (C) ECOptBC (D) SwapBC



(E) ECOptBC (F) SwapBC (G) ECOptBC (H) SwapBC

Figure A3.13: Results of EAC attack on original image and encrypted images coded by ECOptBC and SwapBC: A, B) Cameraman, C, D) Mandrill, E, F) Livingroom, G, H) Goldhill

Bibliography

[1] M. K. Abdmouleh, A. Khalfallah, and M. S. Bouhlel. A novel selective encryption DWT-based algorithm for medical images. In: *IEEE 14th Int. Conf. on Comput. Graph., Imaging and Visualization*. Morocco, 2017, 79–84.

[2] M. N. Asghar and M. Ghanbari. An efficient security system for CABAC bin-strings of H.264/SVC. *IEEE Trans. on Circuits and Syst. for Video Technol.* 23 (2013).

[3] S. Auer, A. Bliem, D. Engel, and A. Unterweger. Bitstream-based "JPEG encryption in real-time. *Int. J. Digital Crime Forensics* 5 (2013).

[4] B. Bhargava, C. Shi, and S. Wang. MPEG video encryption algorithms. *Multimedia Tools and Applications* 24 (2004).

[5] B. Boyadjis, C. Bergeron, B. Pesquet-Popescu, and F. Dufaux. Extended selective encryption of H.264/AVC (CABAC)- and HEVC-encoded video streams. *IEEE Trans. Circuits Syst. Video Technol.* 27 (2017).

[6] H. Cheng and X. Li. Partial encryption of compressed images and videos. *IEEE Trans. on Sig. Process.* 48 (2000).

[7] H. Cheng, X. Zhang, J. Yu, and F. Li. Markov process based retrieval for encrypted JPEG images. In: *Proc. 10th Int. Conf. Availability, Reliab. Security*. 2015, 417–421.

[8] H. Cheng, X. Zhang, J. Yu, and Y. Zhang. Encrypted JPEG image retrieval using block-wise feature comparison. *J. Visual Commun. Image Represent.* 40 (2016), 111–117.

[9] T. S. Cho, S. Avidan, and W. T. Freeman. A probabilistic image jigsaw puzzle solver. In: *2010 IEEE Computer Society Conf. on CVPR*. 2010, 183–190.

[10] T. Chuman, K. Kurihara, and H. Kiya. On the security of block scrambling-based ETC systems against jigsaw puzzle solver attacks. In: *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*. 2017, 2157–2161.

[11] T. Chuman, K. Kurihara, and H. Kiya. JPEG-1 standard 25 years: past, present, and future reasons for a success. *J. of Electronic Imaging* 27 (2018).

[12] T. Chuman, W. Sirichotedumrong, and H. Kiya. Encryption-then-compression systems using grayscale-based image encryption for JPEG images. *IEEE Trans. Inf. Forensics and Security* 14 (2019).

[13] D. Engel, T. Stütz, and A. Uhl. A survey on JPEG2000 encryption. *Multimedia Systems* 15 (2009), 243–270.

[14] S. Fong-In, A. Kiattisin, A. Leelasantitham, and W. San-Um. A partial encryption scheme using absolute-value chaotic map for secure electronic health records. In: *The 4th Joint Int. Conf. on Inf. and Commun. Technol., Electron. and Elect. Eng. (JICTEE)*. Thailand, 2014, 1–5.

[15] M. Ghadi, L. Laouamer, and T. Moulahi. Enhancing digital image integrity by exploiting JPEG bitstream attributes. *J. Innovation Digital Ecosyst.* 2 (2015).

[16] M. Grangetto, E. Magli, and G. Olmo. Multimedia selective encryption by means of randomized arithmetic coding. *IEEE Trans. Multimedia* 8 (2006).

[17] N. Hazarika, S. Borah, and M. Saikia. A wavelet based partial image encryption using chaotic logistic map. In: *IEEE Int. Conf. on Advanced Commun., Control and Computing Technol.s*. India, 2014, 1471–1475.

[18] J. He, S. Huang, S. Tang, and J. Huang. JPEG Image Encryption With Improved Format Compatibility and File Size Preservation. *IEEE Trans. on Multimedia* 10 (2018).

[19] H. Hofbauer and A. Uhl. Selective Encryption of the MC EZBC Bitstream for DRM Scenarios. In: *IEEE 18th European Signal Processing Conf.* Denmark, 2010, 2101–2105.

[20] V. Itier, P. Puteaux, and P. W. Recompression of JPEG crypto-compressed images without a key. *IEEE Trans. Circuits and Syst. for Video Tech.* 30(3) (2020), 646–660.

[21] S. Jenisch and A. Uhl. A Detailed Evaluation of Format-compliant Encryption Methdos for JPEG XR-compressed Images. *EURASIP J. Inform. Security* (2014).

[22] M. Kafi and S. Dumitrescu. Binary Code Tailored for the Selective Encryption of JPEG-Compressed Images. *Submitted 2022 to IEEE Trans. Image Process.* ().

[23] M. Kafi and S. Dumitrescu. Index assignment optimized for partial encryption. In: *16th Canadian Workshop on Information Theory (CWIT)*. Canada, 2019.

[24] M. Kafi and S. Dumitrescu. Binary Code Optimized for Partial Encryption. *IEEE Transactions on Communications* 68(11) (2020), 7201–7217.

[25] M. I. Khan, V. Jeoti, and M. A. Khan. Perceptual encryption of JPEG compressed images using DCT coefficients and splitting of DC coefficients into bitplanes. In: *2010 Int. Conf. on Intelligent and Advanced Sys.* 2010, 1–6.

[26] Kiran, B. D. Parameshachari, and H. T. Panduranga. Partial encryption of medical images by dual DNA addition using DNA encoding. In: *Int. Conf. on Recent Innovations in Sig. process. and Embedded Syst. (RISE)*. India, 2017, 310–314.

[27] B. Kishore, B. K. ShreyamshaKumar, and C. R. Patil. FPGA based simple and fast JPEG encryptor. *J. Real-Time Image Process.* 10 (2015).

[28] A. Kulsoom, D. Xiao, and A. U. Rehman. An efficient and noise resistive selective image encryption scheme for gray images based on chaotic maps and DNA complementary rules. *Multimedia Tools and Applications* (75) (2016), 1–23.

[29] K. Kurihara, S. Imaizumi, S. Shiota, and H. Kiya. An encryption-then-compression system for lossless image compression standards. *IEICE Trans. Inf. Syst.* E100-D (2017).

[30] K. Kurihara, M. Kikuchi, S. Imaizumi, S. Shiota, and H. Kiya. An encryption-then-compression system for JPEG/Motion JPEG standard. *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.* 98 (2015).

[31] E. Lam and J. Goodman. A mathematical analysis of the DCT coefficient distributions for images. *IEEE Trans. Image Process.* 9 (2000).

[32] P. Li and K. Lo. A Content-Adaptive Joint Image Compression and Encryption Scheme. *IEEE Trans. on Multimedia* 20 (2018).

[33] P. Li and K. T. Lo. Joint image compression and encryption based on order-8 alternating transforms. *J. of Visual Comm. and Image Representation* 44 (2017), 61–71.

[34] S. Li, G. Chen, A. Cheung, B. Bhargava, and K. T. Lo. On the design of perceptual MPEG-video encryption algorithms. *IEEE Trans. Circuits Syst. Video Technol.* 17 (2007).

[35] S. Li, R. Ma, and H. Zhang. Enhancing Security for JPEG Image Against Mosaic Attack Using Inter-Block Shuffle Encryption. *IEEE Access* 7 (2019).

[36] S. Li and Y. Zhang. Quantized DCT coefficient category address encryption for JPEG image. *KSII Trans. on Internet and Information Systems (TIIS)* 10 (2016).

[37] W. Li, Weihai, and Y. Yuan. A leak and its remedy in JPEG image encryption. *Int. J. of Computer Mathematics* 84 (2007).

[38] S. Lian, Z. Liu, Z. Ren, and W. H. Secure advanced video coding based on selective encryption algorithms. *IEEE Trans. on Consumer Electron.* 52 (2006).

[39] S. Lian, J. Sun, and Z. Wang. A novel image encryption scheme based-on JPEG encoding. In: *IEEE Proc. of the 8th Int. Conf. on Inf. Visualisation.* UK, 2004, 217–220.

[40] W. Liu, W. Zeng, L. Dong, and Q. Yao. Efficient compression of encrypted grayscale images. *IEEE Trans. Image Process.* 19 (2010).

[41] Y. Liu, Z. Qin, and J. Wu. Cryptanalysis and Enhancement of an Image Encryption Scheme Based on Bit-Plane Extraction and Multiple Chaotic Maps. *IEEE Access* 7 (2019).

[42] Y. Mao and M. Wu. A joint signal processing and cryptographic approach to multimedia encryption. *IEEE Trans. Image Processing* 15(7) (2006), 2061–2075.

[43] A. Massoudi, F. Lefebvre, and C. De Vleeschouwer. Secure and low cost selective encryption for JPEG 2000. In: *IEEE Int. Symp. on Multimedia*. USA, 2008, 31–38.

[44] A. Massoudi, F. Lefebvre, and C. De Vleeschouwer. Secure and low cost selective encryption for JPEG2000. In: *IIEEE Int. Symp. on Multimedia*. USA, 2008, 31–38.

[45] K. Minemura, W. KokSheik, Q. Xiaojun, and T. Kiyoshi. A scrambling framework for block transform compressed image. *Multimedia Tools and Applications* 76 (2017).

[46] X. Niu, C. Zhou, J. Ding, and B. Yang. JPEG Encryption with File Size Preservation. In: *2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing*. 2008, 308–311.

[47] S. Ong, K. Wong, X. Qi, and K. Tanaka. Beyond format-compliant encryption for JPEG image. *Signal Processing: Image Comm.* 31 (2015), 47–60.

[48] S. Y. Ong, K. Minemura, and K. S. Wong. Progressive quality degradation in JPEG compressed image using DC block orientation with rewritable data embedding functionality. In: *2013 IEEE Int. Conf. on Image Process.* 2013, 4574–4578.

[49] C. Pak and L. Huang. A new color image encryption using combination of the 1D chaotic map. *Signal Process.* 138 (2017).

[50] C. H. Papadimitriou and K. Steiglitz. *Combinatorial optimization: algorithms and complexity*. Englewood Cliffs, New Jersey: Prentice-Hall, 1982.

[51] W. Puech and J. M. Rodrigues. Crypto-compression of medical images by selective encryption of DCT. In: *Proc. IEEE Eur. Signal Process. Conf. (EUSIPCO)*. 2015.

[52] Z. Qian, X. Zhang, and S. Wang. Reversible Data Hiding in Encrypted JPEG Bitstream. *IEEE Trans. on Multimedia* 16 (2014).

[53] Z. Qian, H. Zhou, X. Zhang, and W. Zhang. Separable reversible data hiding in encrypted JPEG bitstreams. *IEEE Trans. on Dependable and Secure Computing* 15 (2016).

[54] Z. Qian, H. Zhou, X. Zhang, and W. Zhang. eparable Reversible Data Hiding in Encrypted JPEG Bitstreams. *IEEE Trans. on Dependable and Secure Computing* 15 (2018).

[55] C. Qin, J. Hu, F. Li, Z. Qian, and X. Zhang. JPEG Image Encryption with Adaptive DC Coefficient Prediction and RS Pair Permutation. *IEEE Trans. on Multimedia* doi: 10.1109/TMM.2022.3148591 (2021).

[56] Q. U. Rehman, H. Wang, M. M. A. Shahid, S. Iqbal, Z. Abbas, and A. Firdous. A Selective Cross-Substitution Technique for Encrypting Color Images Using Chaos, DNA Rules and SHA-512. *IEEE Access* 7 (2019), 162786–162802.

[57] I. E. G. Richardson. *H.264 and MPEG-4 Video Compression. Video Coding for Next-generation Multimedia*. The Atrium, Southern Gate, Chichester, West Sussex, England: John Wiley and Sons Ltd, 2003.

[58] A. I. Sallam, O. S. Faragallah, and E. M. El-Rabaie. HEVC selective encryption using RC6 block cipher technique. *IEEE Trans. on Multimedia* 20 (2018).

[59] A. Shafique and J. Shahid. Novel image encryption cryptosystem based on binary bit planes extraction and multiple chaotic maps. *Eur. Phys. J. Plus* 133 (2018).

[60] Z. Shahid and W. Puech. Visual protection of HEVC video by selective encryption of CABAC binstrings. *IEEE Trans. on Multimedia* 16 (2014).

[61] C. Shi and B. Bhargava. A fast MPEG video encryption algorithm. In: *IProceedings of the sixth ACM international conference on Multimedia*. 1998, 81–88.

[62] K. Shimizu and T. Suzuki. Finely Tunable Bitcuboid-Based Encryption With Exception-Free Signed Binarization for JPEG Standard. *IEEE Trans. on Information Forensics and Security* 16 (2021).

[63] B. ShreyamshaKumar and C. R. Patil. JPEG image encryption using fuzzy PN sequences. *Signal, Image and Video Process* 4 (2010).

[64] L. Tang. Methods for encrypting and decrypting MPEG video data efficiently. In: *Proceedings of the fourth ACM international*. 1997.

[65] J. Ting, K. Wong, and S. Ong. Format-Compliant Perceptual Encryption Method for JPEG XT. In: *2019 IEEE Inter. Conf. on Image Process. (ICIP)*. 2019, 4559–4563.

[66] A. S. Tosun and W. Feng. Efficient multi-layer coding and encryption of MPEG video streams. In: *IEEE Int. Conf. on Multimedia and Expo.ICME2000. Proc. Latest Advances in the Fast Changing World of Multimedia (Cat. No.00TH8532)*. USA, 2000, 119–122.

[67] A. Uhl and A. Pommer. Image and Video Encryption. From Digital Rights Management to Secured Personal Communication. *Advances in Information Security* 15 (2005), 1–4.

[68] A. Unterweger and A. Uhl. Length-preserving bit-stream-based JPEG encryption. In: *2012 Proc. 14th ACM Multimedia Security Workshop*. 2012, 85–90.

[69] M. Van Droogenbroeck and R. Benedett. Techniques for a selective encryption of uncompressed and compressed images. In: *In ACIVS Advanced Concepts for Intelligent Vision Systems, Proceedings*. 2002, 90–97.

[70] G. K. Wallace. The JPEG still picture compression standard. *IEEE Trans. on Consumer Electronics* 38 (1992).

[71] W. Wang, M. Hempel, D. Peng, H. Wang, H. Sharif, and H. H. Chen. On energy efficient encryption for video streaming in wireless sensor networks. *IEEE Trans. on Multimedia* 12 (2010).

[72] Y. Wang, M. Ob-Neill, and F. Kurugollu. A tunable encryption scheme and analysis of fast selective encryption for CAVLC and CABAC in H.264/AVC. *IEEE Trans. Circuits Syst. Video Technol.* 23 (2013).

[73] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Trans. Image Proc.* 13 (2004).

[74] *WebP*. https://developers.googleblog.com/2012/08/lossless-and-transparency-modes-in-webp.html..

[75] J. Wen, M. Severa, W. Zeng, M. H. Luttrell, and W. Jin. A format-compliant configurable encryption framework for access control of video. *IEEE Trans. Circuits Syst. Video Technol.* 12 (2002).

[76] C. P. Wu and C. C. J. Kuo. Fast encryption methods for audiovisual data confidentiality. *Proc. SPIE, Multimedia Systems and Applications III* 4209 (2001).

[77] C. P. Wu and C. C. J. Kuo. Design of integrated multimedia compression and encryption systems. *IEEE Trans. on Multimedia* 7 (2005).

[78] T. Xiang, S. Guo, and X. Li. Perceptual visual security index based on edge and texture similarities. *IEEE Trans. Information Forensics and Security* 11 (2016).

[79] T. Xiang, K. Wong, and X. Liao. Selective image encryption using a spatiotemporal chaotic system. *Chaos* 17 (2007).

[80] S. K. A. Yeung, S. Zhu, and B. Zeng. Perceptual video encryption using multiple 8×8 transforms in H.264 and MPEG-4. In: *2011 IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP)*. 2011, 2436–2439.

[81] H. Yin, C. Lin, F. Qiu, J. Liu, G. Min, and B. Li. CASM: A content-aware protocol for secure video multicast. *IEEE Trans. on Multimedia* 8 (2006).

[82] J. Zhou, X. Liu, O. C. Au, and Y. Y. Tang. Designing an efficient image encryption-then-compression system via prediction error clustering and random permutation. *IEEE Trans. Inf. Forensics Security* 9 (2014).

[83] B. B. Zhu, C. Yuan, Y. Wang, and L. S. Scalable protection for MPEG-4 fine granularity scalability. *IEEE Trans. Multimedia* 7 (2005).