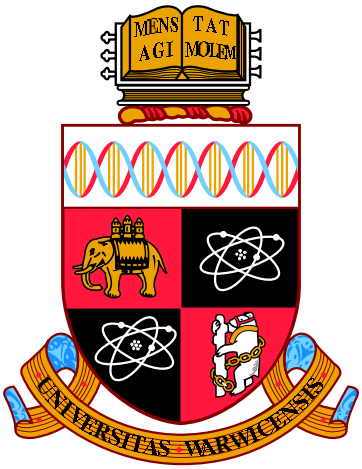
Please contact Warwick Branding and request an A4 keyline and replace this placeholder



Privacy-Preserving Information Hiding and Its Applications

by

Ching-Chun Chang

Thesis

Submitted to the University of Warwick

for the degree of

Doctor of Philosophy

Department of Computer Science

September 2019

Contents

List of Tables

List of Figures

Acknowledgments

First and foremost, I would like to express my sincere thanks to my supervisor Professor Chang-Tsun Li who has introduced me to this wonderful world of science, provided me with immense help with life and study, and granted me considerable opportunities to expand knowledge, gain international experience and build academic networks. His guidance and encouragement have been invaluable throughout my research journey. His expertise and enthusiasm have greatly enlightened me and made me become not only a better scientist but also a better person.

I am deeply indebted to my familiy for all their constant support and selfless care throughout my life. My parents, Professor Chin-Chen Chang and Mrs Ling-Hui Huang, and sisters, Dr Ching-Chieh Chang and Dr Ching-Yun Chang, are my role models and will always be my guidance towards a greater accomplishment.

I would like to humbly extend my gratitude to my advisors Professor Victor Sanchez and Professor Nasir Rajpoot for their professional insight and help throughout the course of this work as well as their kind comments and suggestions which allowed me to keep improving. I would equally like to acknowledge all the research scholars concerned during my academic visits, including Professor Janna Dittmann and Dr Christian Krätzer at the Otto von Guericke University Magdeburg, Germany, Professor Yun-Qing Shi at the New Jersey Institute of Technology, USA, Professor Jianjun Li and Professor Li Li at Hangzhou Dianzi University, China, and many other visiting research fellows.

Special thanks to Dr Xufeng Lin, Dr Rayzhe Li, Dr Ning Jia, Dr Qiang Zhang, Dr Xin Guan, Dr Alaa Khadidos, Dr Roberto Fernandez, Mr Shan Lin, Mr Yijun Quan, and everyone in the laboratory for scientific discussions and technical advice,

which have been proved particularly beneficial and also their kindness without which my life and study would have been much tougher.

Last but not the least, I wish to thank my mentor Professor Jung-San Lee, guitar teachers Ms Yu-Jung Chiu and Dr Harold Gretton, English tutors Mr James Brown and Mrs Anna Churchman, Japanese tutor Ms Yuri Misono, and every friend for bringing colour to my journey.

I would like to dedicate this thesis to the memory of my grandparents whose love and blessings are the light through the darkness and the shelter from the storm.

Declarations

I hereby declare that this thesis entitled *Privacy-Preserving Information Hiding and Its Applications* is my original work and has not been submitted elsewhere for the purpose of obtaining a degree, diploma, or other qualification.

Ching-Chun Chang
23$^{\text{th}}$ April 2019

Publications

- **Ching-Chun Chang**, Chang-Tsun Li and Kaimeng Chen, Privacy-preserving reversible information hiding based on arithmetic of quadratic residues, *IEEE Access*, April 2019.
- **Ching-Chun Chang** and Chang-Tsun Li, Algebraic secret sharing using privacy homomorphisms for IoT-based healthcare systems, *Mathematical Biosciences and Engineering*, April 2019.
- **Ching-Chun Chang**, Chang-Tsun Li, and Yun-Qing Shi, Privacy-aware reversible watermarking in cloud computing environments, *IEEE Access*, November 2018.
- **Ching-Chun Chang** and Chang-Tsun Li, Privacy-preserving reversible watermarking for data exfiltration prevention through lexicographic permutations, in *Proceedings of International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*, Sendai, Japan, November 2018.
- **Ching-Chun Chang** and Chang-Tsun Li, Secure secret sharing in the cloud, in *Proceedings of IEEE International Symposium on Multimedia (ISM)*, Taichung, Taiwan, December 2017.
- **Ching-Chun Chang** and Chang-Tsun Li, Reversible data hiding in JPEG images based on adjustable padding, in *Proceedings of International Workshop on Biometrics and Forensics (IWBF)*, Coventry, UK, April 2017.

Abstract

The phenomenal advances in cloud computing technology have raised concerns about data privacy. Aided by the modern cryptographic techniques such as homomorphic encryption, it has become possible to carry out computations in the encrypted domain and process data without compromising information privacy. In this thesis, we study various classes of privacy-preserving information hiding schemes and their real-world applications for cyber security, cloud computing, Internet of things, *etc.*

Data breach is recognised as one of the most dreadful cyber security threats in which private data is copied, transmitted, viewed, stolen or used by unauthorised parties. Although encryption can obfuscate private information against unauthorised viewing, it may not stop data from illegitimate exportation. Privacy-preserving Information hiding can serve as a potential solution to this issue in such a manner that a permission code is embedded into the encrypted data and can be detected when transmissions occur.

Digital watermarking is a technique that has been used for a wide range of intriguing applications such as data authentication and ownership identification. However, some of the algorithms are proprietary intellectual properties and thus the availability to the general public is rather limited. A possible solution is to outsource the task of watermarking to an authorised cloud service provider, that has legitimate right to execute the algorithms as well as high computational capacity. Privacy-preserving Information hiding is well suited to this scenario since it is operated in the encrypted domain and hence prevents private data from being collected by the

cloud.

Internet of things is a promising technology to healthcare industry. A common framework consists of wearable equipments for monitoring the health status of an individual, a local gateway device for aggregating the data, and a cloud server for storing and analysing the data. However, there are risks that an adversary may attempt to eavesdrop the wireless communication, attack the gateway device or even access to the cloud server. Hence, it is desirable to produce and encrypt the data simultaneously and incorporate secret sharing schemes to realise access control. Privacy-preserving secret sharing is a novel research for fulfilling this function.

In summary, this thesis presents novel schemes and algorithms, including:

- two privacy-preserving reversible information hiding schemes based upon symmetric cryptography using arithmetic of quadratic residues and lexicographic permutations, respectively.
- two privacy-preserving reversible information hiding schemes based upon asymmetric cryptography using multiplicative and additive privacy homomorphisms, respectively.
- four predictive models for assisting the removal of distortions inflicted by information hiding based respectively upon projection theorem, image gradient, total variation denoising, and Bayesian inference.
- three privacy-preserving secret sharing algorithms with different levels of generality.

Chapter 1

Introduction

The past decades have witnessed the worldwide popularisation of social networks and the phenomenal prevalence of public cloud services. Online social networking platforms, such as Facebook, Instagram, and Twitter, have expanded exponentially with a considerable number of posts and a massive influx of personal information. The seemingly unlimited storage space and computational capacity offered by the cloud service providers, such as Amazon, Apple, Dropbox, Google, and Microsoft, have also opened up opportunities for numerous practical applications and appealed to individuals and businesses to entrust an increasing amount of data to the environments out of the control of the data owner. The legality and morality of the use of such personal data by third parties came into question. In many current privacy policies, the sharing of personal information with law enforcement agencies without a warrant is, however, permitted.

Although the public are becoming increasingly reliant upon the Internet, there are serious concerns regarding privacy of sensitive personal information. For instance, a person's social media footprint could affect employment opportunities since employers may screen prospective candidates through social networks and search engines. Moreover, a person's real-time location data might further entail risks of home burglary since it reveals whether the person is at home or on leave. There are risks of intentional or unintentional data leakage to untrustworthy third parties for improper and dishonourable purposes. These issues have raised public awareness about privacy and the sharing of personal data.

Personal information can be exploited in both positive and negative ways depending on the purposes and how they are perceived. On the one hand, sensitive personal information would enable the desirable personalised services and functionalities, for instance, health advice, news feed, product recommendations, and public transit options. The systems will not be able to function without such personal information. On the other hand, although private personal data can be harvested and exploited to understand, engage and influence individuals' opinions, preferences, and decisions, the use of such data in political campaigns is under debate. For example, it was reported that in the 2012 US presidential election, the Obama campaign employed data analytics and micro-targeting tactics to track individual voters and identify potential supporters [1]. It has also been alleged that micro-targeted political advertising was used to influence voters in the 2016 UK Brexit referendum [2]. The UK Parliament's Department for Digital, Culture, Media and Sport (DCMS) condemned that Facebook, Cambridge Analytica, and Aggregate IQ had been involved in the dissemination of disinformation and fake news during the referendum [3]. The point has been made that citizens can only make truly informed decisions about who to vote for if they are certain that those decisions have not been unduly influenced. Hence, when personal data is exploited to target political messages and advertisements, the use should be both transparent and lawful.

In view of the discussed issues, we conclude that there is an urgent need for finding a solution to bridge the gap between privacy and functionality.

1.1 Privacy-Preserving Signal Processing

Privacy-preserving signal processing is an emergent discipline, born as a possible solution addressing privacy concerns in cloud computing [4–10]. The aim is to allow the processing of data and in the meanwhile preserve information privacy even when it is exposed in untrusted environments. The realisation of this aim is often achieved through cooperation with cryptographic schemes, particularly the *homomorphic encryption* schemes which allow computations to be performed in the encrypted domain. Accordingly, this research area is also referred to as *signal processing in the*

encrypted domain in most cases.

Although privacy-preserving signal processing is built upon deep theoretical grounds of cryptography, the research outcomes from the applied aspects are also abundant [11–19]. Apart from the multimedia contents in cloud computing systems, it may also be applied to protect other privacy-sensitive information such as biometric data in access control systems [20–22], criminal databases in forensic investigation systems [23–25], medical records in healthcare systems [26], order history in recommendation systems [27], video footages in surveillance security systems [28] *etc.* If such techniques for processing encrypted data exist, we can then entrust encrypted data to some service providers in order to fulfil the given tasks efficiently without offending certain privacy policies. For instance, the International Criminal Police Organisation, also known as *Interpol*, has built extensive criminal databases containing millions of records related to criminals and crimes. This police network links law enforcement in all member countries and enables authorised parties to track crime trends around the world in real time. To analyse and employ such data, external assistance from academic communities may also be beneficial. However, these research institutions, unlike law enforcement, may not have a sufficiently strong security system to protect sensitive data from leakage during storage and communications. Furthermore, the content of data might be found rather overwhelming due to the lack of professional training. In view of these issues, privacy-preserving signal processing could be a potential solution. For example, the Interpol may request a researcher institution to perform cluster analysis in such a way that criminal evidences of similar attributes are linked together [29]. The Interpol encrypts and entrusts the data to the research institution to be clustered in the encrypted domain and then the clustered results are returned to the Interpol. The whole process of clustering is carried out in the encrypted domain. Therefore, the clustered results are also encrypted and can only be decrypted by the Interpol. In this way, the task is fulfilled without compromising data privacy.

Amongst various topics of privacy-preserving signal processing, this thesis focuses on the techniques and applications of privacy-preserving information hiding. Compared with conventional techniques subsumed under the term 'information hid-

ing', for example, steganography and watermarking, privacy-preserving information hiding has a rather short history. Nevertheless, the rapid growth in the number of academic research outcomes suggests that it is a very promising research area. Multimedia content protection is with no doubt one of the most important topics lies at the heart of the forensics and security communities. The adoption of conventional information hiding techniques for multimedia content protection would, however, face a number of practical issues. In a cloud-based information hiding scenario, sensitive multimedia contents as well as other private metadata are submitted to a cloud service provider by which the message embedding process is carried out. In general, the aim is to protect the given multimedia content via the embedded messages that describe the ownership, authentication code, and other useful information. The cloud is usually assumed to be honest and trustworthy and yet this assumption can only be applied to an ideal world. In reality, the cloud may collect and store personal information, illegally distribute the multimedia content, or even suffer from disastrous security breaches. In 2014, a collection of private photographs of more than hundreds of individuals and celebrities stored in iCloud were compromised and leaked by a malicious hacking attack [30]. This devastating invasion of privacy evoked increasing concerns surrounding security issues of cloud computing services. The solutions for constructing a secure cloud-based information hiding system are based upon the possibility of processing encrypted data [31].

1.2 Issues of Privacy-Preserving Information Hiding

Privacy-preserving information hiding is regarded as one of the most promising and intriguing subdisciplines of privacy-preserving signal processing. It deals with the problem of hiding information into encrypted carrier data and can be applied to fulfil many real-life requests such as data exfiltration prevention, data origin authentication, and electronic data management. It can also be of a substitution strategy when an information hiding algorithm is registered as a proprietary property and thus the access to the algorithm can only be made through authorised third parties. An overview of privacy-preserving information hiding via cloud computing

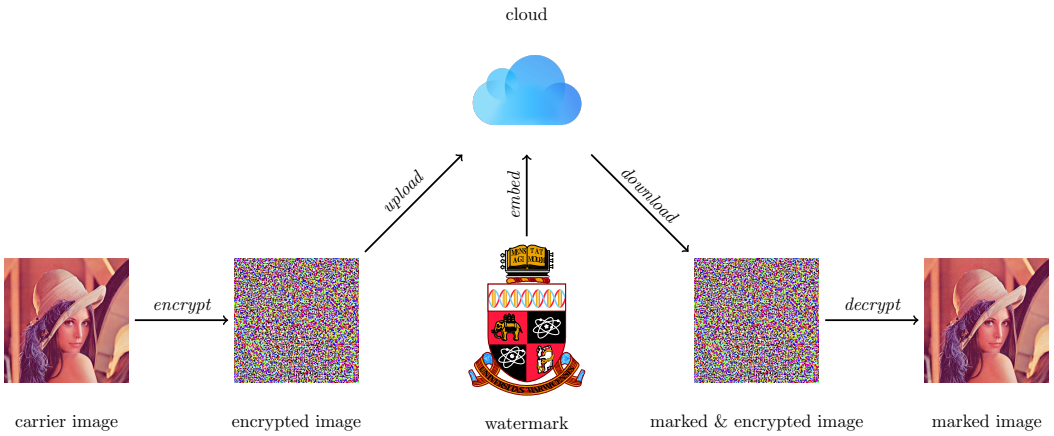


Figure 1.1: Privacy-preserving information hiding via cloud computing.

is illustrated in Fig. 1.1.

Information hiding is a general term encompassing a wide range of research problems including steganography and watermarking. In general, information hiding is defined as the practice of imperceptibly altering a carrier signal to embed a message. The challenge of information hiding in the encrypted domain is a rather difficult one for the reason that when protected by encryption, the carrier signals become semantically unintelligible and cannot be analysed. And the randomness due to the encryption makes it difficult to exploit data redundancy, prohibiting conventional methods of information hiding from being deployed. In addition to this, if a cryptosystem is perfectly secure, it is theoretically not possible to foresee how the change in the encrypted domain would result in the change to carrier signals. The true consequences can only be known after decryption; in other words, whether the distortions caused by information hiding is perceptible or not becomes hardly manageable and difficult to predict before decryption. The main issues concerning privacy-preserving information hiding can be summarised as follows:

- *Cryptographic algorithms:* The choice of cryptographic algorithms forms the basis of a privacy-preserving information hiding scheme. On the one hand, modern public-key cryptosystems, particularly those possess homomorphic properties, are powerful in terms of the ability to perform mathematical operations in the encrypted domain. Homomorphic encryption opens up the possibility to transform a ciphertext into another ciphertext which decrypts to

the same plaintext. This class of cryptosystems is generally secure due to the inherent advantage of circumventing the risks of key exchange. On the other hand, traditional symmetric-key ciphers are comparatively easier to implement since they are usually of low computational complexity and do not have the drawbacks commonly seen in homomorphic cryptosystems such as ciphertext expansion.

- *Data structures:* The data structure in which carrier signals are organised, managed, and stored needs to be taken into account. It is often that cryptographic algorithms cannot be directly applied to multimedia data since many of those algorithms were originally proposed to deal with large integers while multimedia data were not considered or standardised. That is, multimedia data is not structured in a way to facilitate efficient encryption when cryptographic algorithms were initially developed. There are a variety of ways and options to format multimedia data structures for the purposes of encryption and operations that follow. Some structures may, for instance, partition carrier signals into successive blocks based upon the spatial or temporal positions, and some may divide signals according to semantic significance. Some arrangements may involve the padding of additional bits for security purposes or for accommodating privacy-preserving information hiding schemes.

- *Information hiding schemes:* The designs of privacy-preserving information hiding schemes are application-oriented and can be characterised by various features and qualities. Some of the schemes allow the extraction of messages in the encrypted domain and yet the messages will be filtered out along with decryption. This type of schemes may be employed to manage the encrypted files stored online or to monitor the transfer and exportation of encrypted documents. Some of the schemes preserve the embedded information so that the protection furnished by information hiding lasts even after decryption. This type of schemes can be used for the task of information hiding that can be outsourced to authorised third parties, in the sense that the returned result is equivalent to that produced from a conventional information hiding algorithm.

It is also possible to design schemes with a mixture of properties.

- *Content-adaptive predictors:* The decoding process of some privacy-preserving information hiding schemes involves the use of content-adaptive predictors, especially in the schemes that requires the original carrier signals. The precision of content-adaptive predictors could have a crucial effect on the ability and quality of recovery. The designs of predictors are often based upon sophisticated theories and advanced techniques of statistics and signal processing.

1.3 Objectives of This Thesis

The research of privacy-preserving information hiding encompasses a wide range of technical principles and has found many real-world applications, but, be that as it may, the current trend of research focuses predominantly on a specific class of information hiding, namely reversible information hiding. As aforementioned, a widely accepted definition of information hiding is the practice of imperceptibly altering a carrier signal to embed a piece of message. In some highly sensitive applications such as military reconnaissance or medical diagnosis, however, even an imperceptible alteration of data may be strictly prohibitive. This problem has become non-negligible especially when it comes to modern artificial intelligence aided autonomous systems that may have not been trained to be robust against such noise inflicted by information hiding. As such, reversible information hiding is regarded as one of the most favourable solutions. This class of techniques permits the restoration of original carrier signals if desired.

A privacy-preserving reversible information hiding scheme can be categorised by the cryptosystem on which it is based. The possible cryptographic systems to be applied include symmetric-key cryptosystems, asymmetric-key cryptosystems, and other possibly insecure cryptosystems without strict security proofs. By holding the provable security in high regard, we exclude the adoption of risky cryptosystems from consideration. There are merits and demerits of symmetric and asymmetric ciphers. A comprehensive evaluation of props and cons of various cryptosystems is beyond the scope of this thesis. Thus, we briefly conclude that symmetric-key algorithms are

generally of higher computationally efficiency, whereas asymmetric-key algorithms eliminate nearly all risks of secret key exchange. Both class of ciphers are used widely today and hence it is of significance to develop privacy-preserving reversible information hiding schemes for various types of encrypted data with remarkable improvements over the prior art.

Another closely related topic is secret sharing, which is an importance cryptographic technology for access control. A classic secret sharing scheme describes a method for distributing a secret to a group of participants with an access rule that the secret can be reconstructed only when a sufficient number of participants present their shares of secret. The sharing process is carried out by a dealer, who is assumed to be trustworthy. In Internet of things enabled healthcare applications, the secret can be conceived as health data recorded by wearable devices, and the dealer is a centre hub that collects and aggregates the data. The dealer shares the health data amongst a group of authorised medical practitioners and the data can only be retrieved when a sufficient number of practitioners consent to do so. The threshold may be defined and agreed by the patient. However, the dealer's privilege to access patient's private information might come into question. Even if it is permitted by patient himself or herself, the risk of data leakage due to malicious cybersecurity attacks still remains. A potential solution is to encrypt the health data at the very beginning when it is recorded by the wearable devices and sent to the centre hub. In order to enable the dealer to distribute data in the encrypted domain, there is a need for developing privacy-preserving secret sharing techniques.

Recognising the issues argued previously and the need to tackle them, the objectives of this thesis are summarised as follows:

- To develop privacy-preserving reversible information hiding techniques for data encrypted respectively by efficient symmetric-key ciphers and secure asymmetric-key cryptosystems (refer to Chapter 3 and 4).

- To demonstrate how different homomorphic properties can be utilised to construct information hiding schemes for encrypted data (refer to Chapter 4).

- To devise different types of predictive models in order to achieve better per-

formance in carrier signal recovery (refer to Chapter 3 and 4).

- To make improvements over the state-of-the-art with regard to capacity, fidelity, and reversibility (refer to Chapter 3 and 4).
- To introduce the novel research of privacy-preserving secret sharing and its applications (refer to Chapter 5).

1.4 Outline of This Thesis

The rest of this thesis is organised as follows.

- Chapter 2 discusses the fundamental knowledge of related disciplines and gives a literature review. It begins with an introduction of steganography, watermarking, reversible information hiding, and cryptography. Then, some of the most representative works in the research area of privacy-preserving reversible information hiding are categorised and reviewed.
- Chapter 3 presents privacy-preserving reversible information hiding schemes based upon symmetric-key ciphers. It begins with an information hiding scheme that utilises the arithmetic of quadratic residues to embed additional message and a content-adaptive predictive model derived from the projection theorem to remove distortions caused by message embedding. It then discusses another information hiding scheme using lexicographic permutations and a content-adaptive predictor based upon image edge gradient. At the end of each scheme construction, an experimental analysis is provided and a comparison with the state-of-the-art is made.
- Chapter 4 investigates privacy-preserving reversible information hiding schemes compatible to asymmetric-key cryptosystems. It presents information hiding schemes based upon multiplicative and additive homomorphisms respectively. In order to satisfy different operational requirements, it introduces an online predictors based upon total variation denosing model and an offline predictor based upon Bayesian inference. At the end of it, simulation results are analysed and discussed.

- Chapter 5 introduces a novel research of privacy-preserving secret sharing and showed how to apply this technology to the Internet of things based healthcare systems. It starts with a discussion of simplest solutions and points out their deficiencies in practical aspect. Three schemes with an ascending generality have been proposed to address the issues with naïve solutions. The proposed schemes permit secrets to be encrypted and shared amongst a group of authorised users, while the reconstruction of secrets is intended to be as efficient as possible. Moreover, it has been shown that the proposed privacy-preserving secret sharing schemes can achieve the same security level and access structure as Shamir's secret sharing scheme, which is an ideal scheme and yet is only applied to unencrypted messages.

- Chapter 6 concludes the thesis and suggests some potential directions for future research.

Chapter 2

Background

Privacy-preserving information hiding is a multidisciplinary study combining cryptography, steganography, watermarking, signal processing, statistical analysis, *etc.* A good grasp of these disciplines is essential for a clear understanding of the advancement of privacy-preserving information hiding technologies. With the present chapter, we begin our exploration of the fundamental knowledge of information hiding. We provide a brief history of two main branches of information hiding, namely steganography and watermarking, as well as a discussion of their characteristics and applications. Then we introduce reversible information hiding, on which a large part of this thesis focuses. In some applications, the ability to remove alterations inflicted by information hiding would be desirable and hence the notion of reversible information hiding was introduced to tackle restoration of the original carrier signals. After that, we provide a brief introduction of modern cryptography including symmetric-key cryptography, asymmetric-key cryptography, and homomorphic encryption. Different cryptosystems have very distinct characteristics and thus shall be handled and used differently in privacy-preserving information hiding systems. This chapter is ended by a literature review of some of the most representative privacy-preserving reversible information hiding schemes.

2.1 Information Hiding

Information hiding is a general term comprehending the disciplines of steganography and watermarking [32–35]. Some definitions of steganography and watermarking were given in the textbook *Digital Watermarking and Steganography* by Ingemar J. Cox *et al.* as [36]:

> *Steganography is defined as the practice of undetectably altering a work to embed a secret message.*
> *Watermarking is defined as the practice of imperceptibly altering a work to embed a message about that work.*

Although steganography and watermarking are closely related fields that to some extent share a great deal of similar techniques, the divergence of essential principles has led to quite different requirements and thus the technical approaches.

2.1.1 Steganography

The word *steganography* comes from the Greek words steganos, which means 'covered or concealed', and graphein, which means 'writing' [37]. The first recorded use of the term was in Trithemius's book *Steganographia* written in 1499 [38]. Yet, one of the first documented evidences and oft-cited examples of steganography can be traced back to the *Histories of Herodotus* written in 440 BCE [39]. It recorded a steganographic method which is to shave the head of a messenger and tattoo a message on the messenger's head. After the hair grew, the message would be covered and become undetectable until the head was shaved again. Another common form of steganography is through the use of invisible inks, which can be revealed only under certain circumstances such as being heated, or being mixed with appropriate chemical substances, or being viewed under ultraviolet light. With the development of photographic reduction techniques, messages were substantially reduced into tiny photographs, which can be read only with a specially designed magnifying viewer. During World War I and World War II, these tiny photographs, called *microdots*, were stuck on top of printed periods or commas in innocuous cover material such as postcards and magazines, which were then delivered through insecure postal channels

[40]. Public awareness of modern digital steganography increased drastically after the 9/11 terrorist attacks. It was suggested that steganographic softwares might have been applied to coordinate the intrigues.

In 1984, Simmons formulated steganography as the *Prisoner's Problem* [41]. Two accomplices in a crime have been arrested and kept confined to separate cells. Their only means of communication is to convey messages through the warden, who will only permit to pass the messages if the information is innocuous. In order to coordinate their escape plan as well as to deceive the warden, they will have to think of a way to establish a *subliminal channel*. This problem could be seen as an analogy for many other real-world applications of steganography [42–50].

2.1.2 Watermarking

Although the art of papermaking can date back to about 105 BCE in China, it was not until the end of 13th century that watermarks appeared in Italy. The purpose of watermarks at the time of their invention might be untraceable. They may have been served as trademarks of paper maker, or simply as decoration. The first specific watermark in an attempt to thwart counterfeiting of paper currency appeared in 1661 when the first European banknotes were issued in Sweden. A traditional watermark is a design stamped on wet pulp during the papermaking process and only reveals itself when the paper is held up to light. It is almost invisible to the naked eye under normal viewing conditions and is difficult to be reproduced.

The term 'digital watermark' was coined in early 90s when the feasibility of encoding watermarks into digital images was investigated [51]. Digital watermarking techniques can be used for a wide variety of applications, including but not limited to access control, broadcast monitoring, copyright protection, data annotation, forgery detection, message authentication, ownership identification and traitor tracing [52–54]. The development of digital watermarking can be further categorised into a number of subfields such as *robust watermarking* and *fragile watermarking*.

As its name would suggest, a robust watermark is designed to survive incidental or deliberate removal and is widely used for copyright protection and many other real-life applications [55–64]. Attacks against robust watermarks often involve

desynchronisation operations to paralyse watermark detection process. For digital images, desynchronisation can be realised by geometrical transformations including cropping, flipping, padding, rescaling, rotating, shifting, and transposing [65].

By contrast, a fragile watermark would fail to be detected if the cover content is altered in any way and its fragility can be a favourable characteristic for authentication purposes [66–73]. After answering whether a content has been modified, the *localisation* and *restoration* of tampered areas may be taken into further consideration. The knowledge of which part of the content has been altered and which part is authentic could be beneficial to, say, the understanding of motivations behind tampering, while the possibility of recovering the altered parts can, for example, save the cost of retransmissions or be even more valuable when retransmissions are not possible. To enable the function of localisation, the watermark components should be embedded in such a way that failing to verify one component should not affect the verification of others [74–79]. To further recover the altered areas, one may resort to the *self-embedding* approach in which the watermark payload contains a reconstruction reference such as a compression code of the cover content itself [80–93].

The properties of watermarking schemes and their relative importance is dependent upon the application in hand. Amongst various watermarking systems, most common properties are *capacity* and *fidelity*. The capacity refers to the amount of data payload can be encoded into digital media. It may be measured by the number of bits per pixel for images, per second for the audios, per frame for videos, or per sentence for texts. The fidelity refers to the perceptual similarity between the original content and its watermarked counterpart. It can be evaluated by objective quality assessment metrics such as mean square error (MSE) or signal-to-noise ratio (SNR) [94–96]. It is also possible to produce the rate by subjective methods, for example, by involving human observers or applying perceptual models of human visual system (HVS) and human auditory system (HAS) to automatically predict human judgement [97].

2.2 Reversible Information Hiding

In digital forensic science, one of the most critical issues would be the authentication of digital evidence against illegitimate manipulations [98]. *Digital signature* schemes serve as one of the most effective solutions towards message authentication [99]. Typically, a digital signature is the encryption of a *hash value*, or a digest of a file, in a sense that:

- It cannot be forged as long as the private key (for encryption) remains secure.
- It is verifiable as long as the public key (for decryption) is available.

In 1993, Friedman proposed a trustworthy digital camera that contains a microprocessor for generating digital signatures when photos are taken [100]. The private key only known to the camera manufacturer is programmed into the microprocessor, whereas the public key is stored as image files' metadata and also engraved on the camera body. To verify the image file in question, the verification software decrypts the stored signature with the public key to obtain the hash value, and compares it with the hash value produced from the image in question. The verification is passed if both values match; otherwise, the image fails to be authenticated and is judged as tampered.

This construction, however, requires additional storage space for signatures and furthermore has risks of data loss and mismanagement during storage, transmission, or format transformation. Although watermarking can be considered as a potential solution addressing the problem of mislaying digital signatures as well as other auxiliary information, the modifications by the act of watermarking itself may violate the initial objective of integrity protection. In some cases, these non-malicious modifications and imperceptible distortions could be admissible and tolerable. In some sensitive scenarios, however, such alterations would be strictly forbidden, especially in the cases such as military reconnaissance or medical diagnosis. It might be argued that the noise introduced is too faint to be a possible cause of misinterpretation of medical images in a malpractice suit. Yet, this argument may not be persuasive in a courtroom.

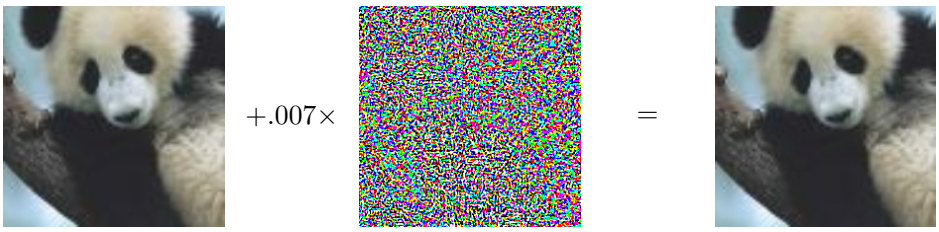
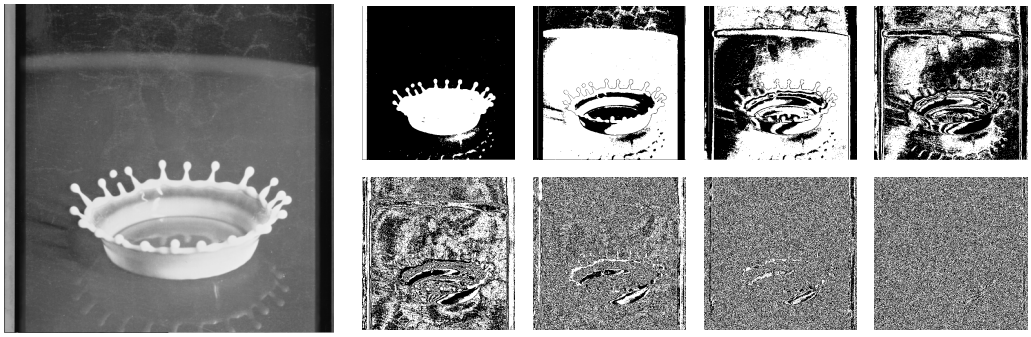


Figure 2.1: A demonstration of adversarial perturbations. By adding an imperceptible colour noise to an image 'panda' from ImageNet, GoogLeNet mistakenly categorised it as a 'gibbon' with 99.3% confidence.

Another critical concern is accompanied by the recent development of artificial intelligence aided automated systems, such as autonomous vehicle systems and autonomous diagnostic systems. It is evident that many current deep neural networks would not able to sustain some *adversarial perturbations* in a sense that some imperceptible noise would chance to or intend to mislead the model in a wrong or even chosen direction, as a demonstration shown in Fig. 2.1 provided by Goodfellow *et al.* [101]. It is also of a practical possibility that a few distorted samples of data collected and used in the training process would poison and eventually compromise the whole model. Hence, we conclude that the ability to preserve *perfect copies* of original images is not only an academic pursuit but also of great significance in real-life applications.

In order to fulfil the requirement of preserving the original carrier signals, the notion of *reversible information hiding* was introduced and has continued to advance over the last two decades [102]. Reversible information hiding is a special class of information hiding techniques that permits the restoration of original carrier signals once the embedded messages are extracted. To the best of our knowledge, the very first reversible information hiding algorithm was invented by Barton and issued as a US patent in 1997 [103]. The invention relates to an information hiding method and apparatus for verifying whether the digital data has been modified from its intended form and allowed restoration of data to its original state if desired. Relevant literature suggested that reversible information hiding schemes are favourable for many authentication applications, and thus we would argue that reversible



Greyscale image Bit planes

Figure 2.2: An example of eight bit planes of a greyscale image. The significance of bit planes decreases from top left to bottom right.

information hiding is more associated with watermarking, instead of steganography, from the application aspect. Accordingly, we may also refer to reversible information hiding as *reversible watermarking* and message payloads as *watermarks*. In the rest of this section, we review some of the most representative techniques of reversible information hiding.

2.2.1 Lossless Compression

To some extent the problem of reversible information hiding can be conceived as lossless data compression with a fidelity constraint [104–109]. In other words, if we are able to compress some perceptually insignificant components of the carrier signal, then we are able to accommodate additional information while preserving the similarity between the original and modified signals. This idea is intuitive and to some extent feasible, but it would encounter an inherent problem of a rather limited embedding capacity due to the nature of signals. It can be observed that those imperceptible parts of signals in most cases would look like random noise signals, which is nearly incompressible. Let us take digital images of 256 pixel intensities as example. It can be seen in Fig. 2.2 that the randomness of each bit plane increases as the visual significance decreases. It indicates that if one wants to embed more amount of information, some significant components of a signal might inevitably be modified, which inflicts visible distortions.

The problem of reversible information hiding can be considered as a matter

of striking a good fidelity-capacity trade-off. The overall objective is to reach a more efficient balance in a sense that higher capacity is achieved under a fixed fidelity constraint or *vice versa*. A theoretical aspect of this problem was investigated by Kalker and Willems, who formulated it as a rate-distortion problem.

Theorem. *Let X and Y denote the memoryless random variables of carrier signal and its marked transition respectively, H denote the entropy function, and D denote a preferred distance function. The upper bound of embedding rate ρ under a distortion constraint Δ is given by*

$$\rho(\Delta) = \max\{H(Y)\} - H(X), \tag{2.1}$$

where the maximum is over all transition probabilities $P(y|x)$ satisfying the distortion constraint

$$\sum_{x,y} P(x)P(y|x)D(x,y) \leq \Delta. \tag{2.2}$$

Based upon the theoretical formulation, Kalker and Willems presented a recursive reversible embedding approach aiming at approaching the theoretical upper bound [110]. The idea is straightforward: First, we segment the carrier bitstream into n disjoint subsequences $\mathbf{x} = x_1||x_2||\ldots||x_n$. Second, we encode the first piece of payload into x_1 through an arbitrary irreversible embedding technique and obtain the result y_1. Third, we compute the auxiliary information needed to reconstruct x_1 when y_1 is available. Theoretically, the amount of information required is equal to $H(x_1|y_1)$. Then, we embed the auxiliary information for reconstructing x_1 as well as the next piece of payload into x_2 via the chosen irreversible embedding technique, resulting in y_2. We repeat this process recursively until the one but the last subsequence x_n, for which we simply deploy the lossless compression strategy. However, this pioneering study only considered independent and identically distributed (i.i.d.) random variables of carrier signal and therefore, efficient approaches to cope with non i.i.d. sequences and to narrow the gap between theoretical and practical performance have played a pivotal role in this line of research [111–117].

2.2.2 Difference Expansion

Difference expansion is a branch of techniques based upon reversible integer transform. This idea was first introduced by Tian [118], who proposed to transform a pair of adjacent pixels of the carrier image into a pair of moving average or *trend* and moving difference or *fluctuation* via integer Haar wavelet transfrom, and embeds one bit into the expanded difference term. Let x and y be a pair of adjacent pixels of an 8-bit greyscale image such that $0 \leq x, y \leq 255$. Their average μ and difference δ are defined as

$$\begin{aligned} \mu &= \left\lfloor \frac{x+y}{2} \right\rfloor, \\ \delta &= x - y, \end{aligned} \tag{2.3}$$

where $\lfloor . \rfloor$ denotes the floor function. The inverse transform from μ and δ to x and y is computed by

$$\begin{aligned} x &= \mu + \left\lfloor \frac{\delta+1}{2} \right\rfloor, \\ y &= \mu - \left\lfloor \frac{\delta}{2} \right\rfloor. \end{aligned} \tag{2.4}$$

Let b denote one bit of information to be embedded such that $b \in \{0, 1\}$. The message embedding procedures are operated as follows:

1. Transform x and y into μ and δ.

2. Expand δ by a scale of 2 and embed b by

$$\delta' = 2 \times \delta + b. \tag{2.5}$$

3. Transform μ and δ' inversely into x' and y'.

Given that most of the energy is preserved in the trend signal which has been kept changed during the embedding process, only tiny fluctuations in visual quality would occur. The message extraction and image recovery procedures are demonstrated as follows:

1. Transform x' and y' into μ and δ'.

2. Extract b by
$$b = \delta' \mod 2. \tag{2.6}$$

3. Recover δ from δ' by
$$\delta = \left\lfloor \frac{\delta'}{2} \right\rfloor. \tag{2.7}$$

4. Transform μ and δ inversely into the original x and y.

Albeit its simplicity, the above scheme leaves the problems of overflow and underflow unsolved. In other words, the modified pixels x' and y' went out of bounds of the colour depth (256 possible grey levels). To rectify this flaw, the encoder needs to record the location of changeable pixel pairs on a *location map*. There are many possible ways to communicate this map to the decoder. For instance, we can employ the *LSB replacement* method to replace least significant bits (LSBs) of some pre-agreed pixels between the encoder and decoder with the compressed bitstream of the location map, and then make the original bits as a part of the message payloads.

Example. *We illustrate the difference expansion approach with an example provided in Tian's article. Assume that we have two pixels* $x = 206$ *and* $y = 201$*, and we would like to embed a bit* $b = 1$*. First, we compute the average* μ *and difference* δ *by*

$$\mu = \left\lfloor \frac{206 + 201}{2} \right\rfloor = 203,$$
$$\delta = 206 - 201 = 5.$$

Then, we expand the difference δ *by a scale of* 2 *and add the bit* b *to it, resulting*

$$\delta' = 2 \times 5 + 1 = 11.$$

Finally, we transform μ *and* δ' *inversely into new pixel values* x' *and* y' *by*

$$x' = 203 + \left\lfloor \frac{11 + 1}{2} \right\rfloor = 209,$$
$$y' = 203 - \left\lfloor \frac{11}{2} \right\rfloor = 198.$$

To extract the bit b, we compute

$$b = 11 \mod 2 = 1.$$

We then obtain the original average μ and difference δ by

$$\mu = \left\lfloor \frac{209 + 198}{2} \right\rfloor = 203,$$
$$\delta = \left\lfloor \frac{11}{2} \right\rfloor = 5.$$

Finally, we use μ and δ to restore x and y as

$$x = 203 + \left\lfloor \frac{5 + 1}{2} \right\rfloor = 206,$$
$$y = 203 - \left\lfloor \frac{5}{2} \right\rfloor = 201.$$

Followed by this paradigm, there have been rapid developments in reversible integer transform. Improvements over Tian's original method include a generalised integer transform by Alattar [119], a reversible contrast mapping by Coltuc and Chassery [120], and other efficient integer transforms [121–123]. To further reduce the size of location map, a sorting technique was proposed by Kamstra and Heijmans [124].

2.2.3 Histogram Shifting

One of the most significant breakthrough in reversible information hiding techniques is the *histogram shifting* approach proposed by Ni *et al.* [125]. This idea had laid the groundwork for a variety of improved constructions. An overview of histogram shifting technique is shown in Fig. 2.3. In general, an embedding algorithm consists of three steps:

1. Generate the histogram of the host signal.

2. Split the histogram into an inner region and an outer region.

3. Embed message by modifying the inner bins.

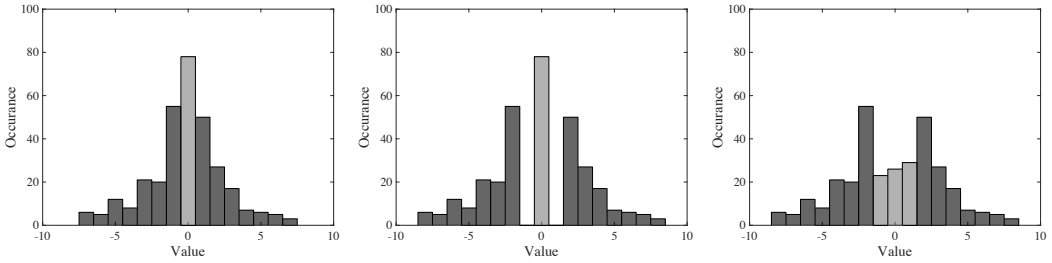


Figure 2.3: An illustration of the histogram-shifting reversible watermarking technique. Given a designated bin (coloured in light grey), the adjacent bins are emptied out by shifting the bins at both sides (coloured in dark grey) outwards. To represent the payload information, the sample values of the designated bin are mapped to the adjacent values.

In order to make room for data embedding, or equivalently to disambiguate the outer bins and the modified inner bins, the outer bins are to be shifted outwards. In more detail, the adjacent bins of a designated bin, usually the peak bin, are emptied out by shifting outwards. Then, each sample value of the designated bin is mapped to its adjacent values to represent the payload information. The process can be reversed by simply shifting the bins inwards and mapping the values back. If the bins at two ends of the histogram are not empty in the first place, a small amount of additional information will be needed to record the overflow.

Among various extensions, one of the most renowned method is *prediction-error expansion* by Thodi and Rodriguez [126]. This method takes account of correlations inherent in the neighbourhood of a pixel, rather than merely the correlation between two adjacent pixels. This technique has greatly influenced the subsequent development of reversible information hiding schemes. Improvements over this method mainly focus on exploiting more advanced prediction techniques to generate a more sharply distributed *prediction-error histogram*. Sachnev *et al.* utilised rhombus prediction, double-layered embedding mechanism, and sorting strategy to improve the performance [127]. Luo *et al.* devised an interpolation-based predictive model which calculates the estimated pixel value as the weighted sum of its four adjacent neighbours [128]. Qin *et al.* employed an inpainting technique to assist the prediction [129]. Fallahpour adopted gradient adjusted predictor (GAP) to improve the estimation accuracy [130]. Coltuc investigated and analysed a variety of

predictors and their performances within scope of reversible information hiding [131]. Ou *et al.* proposed a idea of using a two-dimensional prediction-error histogram [132].

Furthermore, adaptive embedding and pixel-selection strategies were developed to classify pixels according their local complexity, and by exploiting only pixels in the smooth regions, a sharply-distributed prediction-error histogram can be generated to carry a larger payload [133–137].

Another type of improved methods aims to find optimal histogram bins for data embedding. Fallahpour and Sedaaghi devised a block-based mechanism that embeds data by modifying the peaks of histograms generated from distinct blocks of the carrier image [138]. Xuan *et al.* suggested to modify the histogram generated from high frequency coefficients of the integer wavelet transform (IWT) of the host image [139]. Li *et al.* proposed a general framework that concludes a number of histogram shifting based schemes as its special cases [140]. Wang et al. modelled the multiple-layered histogram shifting as a rate-distortion problem and employed the genetic algorithm (GA) to search for the nearly optimal bins [141]. Efficient methods using multiple histograms were also investigated [142–145].

2.3 Modern Cryptography

Cryptography is the study of secure communications in the presence of malicious adversaries [146]. In a narrower sense, it is referred to the study of a pair of encryption and decryption algorithms such that the sender enciphers a message, or a *plaintext*, into a disguised message called *ciphertext* in order to prevent the secret information from being revealed to an eavesdropper, and only the intended recipient is able to decipher and read the message. In the literature of cryptography, for convenience and to aid comprehension, the name Alice is often used to refer to the sender, Bob to the intended recipient, and Eve to the adversary, especially the eavesdropper. We will use these names of fictional characters extensively in discussions about a variety of systems and protocols.

The modern study of cryptography can be generally divided into *symmetric-*

key cryptography and *asymmetric-key cryptography* [147]. As their names suggest, the symmetric-key cryptography is referred to the cryptosystems that use the same key for both enciphering and deciphering, whereas the asymmetric-key cryptography, also known as *public-key cryptography*, indicates the cryptosystems that utilise different keys for encryption and decryption of the messages. In response to privacy concerns in cloud computing environments and other privacy-aware applications, there has been a surge of research in the development of *homomorphic cryptosystems* which permit computations on ciphertexts. We begin this section with basic cryptographic principles and attacks. Then, we discuss some representative cryptosystems of different categories. This section ends with a non-technical introduction of homomorphic cryptosystems.

2.3.1 Kerckhoffs's Principles

In 1883, A. Kerckhoffs stated six principles for the design of military ciphers, which have been regarded the basic tenet of cryptography [148]:

i The system must be practically, if not mathematically, indecipherable.

ii It must not require secrecy and must be able to fall into the hands of the enemy without inconvenience.

iii Its key must be communicable and retainable without the aid of written notes, and must be changeable or modifiable at the discretion of the correspondents.

iv It must be applicable to telegraphic correspondence.

v It must be portable, and its usage and function must not require the concourse of several people.

vi Finally, it is necessary, given the circumstances that control the application, that the system is easy to use, requiring no mental strain, or the knowledge of a long series of rules to be observed.

Although some of the rules are no longer applicable due to the emergence of modern computers, the second rule is still of great significance today. This rule was

echoed and rephrased by C. Shannon as 'the enemy knows the system', which is also widely recognised as *Shannon's maxim* [149]. In other words, if the messages were eavesdropped and copied while en route, the messages should be indecipherable even though the enemy has full knowledge of the algorithms and apparatus used.

2.3.2 Basic Cryptographic Attacks

A cryptographic attack is a method for discovering the weaknesses in algorithms, protocols, or implementations of cryptosystems with the eventual goal to determine the key and thus the secret message in its entirety. This process is also known as *cryptanalysis*. Attacks can be classified based upon what information is available to the attacker. The most common and basic types of cryptanalytic attacks are listed and discussed in order of severity as follows [150].

- *Ciphertext-only attack (COA):* The attacker is assumed to have only access to a set of ciphertexts and attempts to deduce some information about the key or the plaintext. A cryptosystem that is vulnerable to this type of attack would be considered as completely insecure. In the history of cryptography, one of the simplest and most widely known encryption techniques is Caesar's cipher, which simply shifts the plaintext alphabet left or right by some number of positions. This classical cipher can be easily broken in a ciphertext-only scenario by frequency analysis, that matches up the frequency distribution of the enciphered letters with the statistics of the plaintext language.

- *Known-plaintext attack (KPA):* The attacker has access to a set of plaintexts and their enciphered counterparts, which can be used to deduce the secret keys or codebooks. One of most well-known practice of this method is the breaking of the German Enigma machine by Alan Turing and other cryptanalysts during World War II. To decipher the Enigma messages, British cryptanalysts enticed the Germans to include particular words, called 'cribs', in their enciphered messages. This process of planting cribs was called 'gardening'. Chunks of known plaintext enabled the exhaustive search space to be narrowed down to an amount that eventually can be solved with the help of systematically

constructed electro-mechanical devices, such as the Bombe.

- *Chosen-plaintext attack (CPA):* This model presumes that the attacker is able to acquire the corresponding ciphertexts of arbitrarily chosen plaintexts. The target is to reveal the plaintext that was encrypted to give some other ciphertext, or the *challenge ciphertext*. It is usually conceptualised by allowing the attacker to interact with an *encryption oracle*, or less pedantically a black box. Yet, this attack is a realistic threat in many scenarios. During the Pacific War, US Navy cryptanalysts has intercepted a message from Japan containing the ciphertext fragment 'AF' and they believed that it might correspond to the plaintext 'Midway Island'. In order to prove this hypothesis, they devised a ruse by instructing the US forces at Midway to broadcast an uncoded radio message stating that Midway's freshwater supplies were low. Within 24 hours, the US cryptanalysts intercepted a message from Japan reporting 'AF was short on water', which confirmed their hypothesis.

- *Chosen-ciphertext attack (CCA):* This model presumes that the attacker is given the capability to obtain the decryption of any chosen ciphertext. The aim is to determine the secret key in whole or in part. In other words, the attacker is allowed unlimited access to a *decryption oracle*. By its very nature, this attack is difficult to mount and yet the vulnerability to such attack is not only a theoretical possibility. In real life, often, the attacker can issue ciphertext queries to a server and fool it into decrypting those chosen ciphertexts. The attacker is also possible to learn partial information about the plaintext from observing server's responses such as 'invalid checksum', 'invalid timestamp', or 'invalid password'.

We conclude by noting that different application scenarios may require resistance to different types of attacks. It is not always the case that a cryptographic scheme secure against the most powerful type of attacks would be favourable. When taking the algorithm efficiency and computational complexity into account, the cryptographic scheme secure against weaker attacks may be preferred provided that it suffices for the applications at hand.

2.3.3 Symmetric-Key Cryptography

Symmetric-key cryptosystems refer to cryptographic algorithms that use the same keys for both encryption of plaintext and decryption of ciphertext. As one of the most notable milestones in the history of cryptography, an unbreakable cipher was invented by Vernam with a patent granted in 1919 [151], which is also referred to as the *one-time pad* today. Yet, it was not until 1949 that the perfect secrecy of Vernam's cipher was proved by C. Shannon in his seminal work *Communication Theory of Secrecy Systems* [149]. It was shown that the one-time pad is unbreakable and anything unbreakable is naturally a one-time pad. Despite its theoretical perfect secrecy, it is by no means a practical system given that it is unbreakable if and only if

- The key is truly random.
- It is only used once and never reused in whole or in part.
- It is kept completely secret.
- Its length is the same as, or longer than, the length of the message.

As might be expected, it is hard to generate truly random keys and, moreover, even the definition of true randomness could be considered as a vague philosophical question. In addition to this, the cost of storing and exchanging such large volumes of keys would be very expensive. To overcome these issues, stream ciphers were innovated as an approximation of the unbreakable one-time pad, which generate keys via a pseudorandom number generator (PRNG). One of the most prominent and widespread stream ciphers is Rivest Cipher 4 (RC4), designed by R. Rivest in 1987 [152]. It has been used and supported in many commercial software packages.

Data Encryption Standard (DES) was a publicly available cipher developed by H. Feistel and the research team of IBM, and became a public standard in 1977 with the involvement of the National Security Agency (NSA) [153]. However, it has been criticised for a relatively short key length (56 bits). As computational speed continued to grow rapidly, this weakness had become more serious and finally been rendered insecure by brute-force attacks. In 2001, Advanced Encryption Standard

(AES) was published as a successor of DES [154]. It is sometimes referred to as Rijndael owing to its original designers Rijmen and Daemen. The main strength of AES rests upon the various options of key sizes (128, 192, or 256 bits), which make it exponentially stronger than DES. There is, as of this writing, no practical attack against AES that could reduce the amount of time to crack it to less than billions of years, or the age of the universe, even on the world's fastest supercomputer.

2.3.4 Asymmetric-Key Cryptography

A common issue with all of the symmetric-key cryptosystems is that the sender and the recipient must agree on a key prior to the communications. In order to pre-share the key, a secure channel resistant to eavesdropping must be established and yet the cost of implementation and maintenance could be very expensive. This major problem has led to a full-blown revolution in cryptography, starting from a seminal paper *New Directions in Cryptography* by W. Diffie and M. Hellman in 1976 [155], credited as one of the most influential papers in the history of cryptography. A brief description of *Diffie–Hellman key exchange* protocol is provided as follows.

- A sender Alice and a receiver Bob publicly agree to use a modulus $p = 23$ and base $g = 5$ (which is a primitive root modulo 23). In other words, these two digits can be known by an enemy without threatening security.
- Alice selects a secret number $a = 4$ and then sends Bob

$$A \equiv g^a \mod p \equiv 54 \mod 23 \equiv 4. \tag{2.8}$$

  Note that a should be remained secret to anyone but Alice herself.

- Bob chooses a secret number $b = 3$ and then sends Alice

$$B \equiv g^b \mod p \equiv 53 \mod 23 \equiv 10. \tag{2.9}$$

  Note that b should only be known to Bob himself.

- Alice computes the shared key

$$k \equiv Ba \mod p \equiv 104 \mod 23 \equiv 18. \tag{2.10}$$

- Bob obtains the shared key

$$k \equiv Ab \mod p \equiv 43 \mod 23 \equiv 18. \tag{2.11}$$

- As a result, Alice and Bob now share a key $k = 18$, which can be used to encrypt messages in their following communications.

The security of Diffie–Hellman key exchange is based upon *discrete logarithm* problem, recognised as one of the most computationally intractable problem in number theory. In other words, to break the scheme is equivalent to solve a very hard mathematical problem and the concept of which lies at the heart of subsequent development of asymmetric-key cryptosystems. In fact, the problem of good cipher design is essentially one of finding difficult problems, subject to certain conditions. As suggested by Shannon [149]:

> *There are two approaches to this problem; (1) We can study the possible methods of solution available to the cryptanalyst and attempt to describe them in sufficiently general terms to cover any methods he might use. We then construct our system to resist this 'general' method of solution. (2) We may construct our cipher in such a way that breaking it is equivalent to (or requires at some point in the process) the solution of some problem known to be laborious. Thus, if we could show that solving a certain system requires at least as much work as solving a system of simultaneous equations in a large number of unknowns, of a complex type, then we would have a lower bound of sorts for the work characteristic.*

In 1978, one of the first, and also the most dominant to date, public-key cryptosystems was invented at MIT by R. Rivest, A. Shamir, and L. Adleman [99]. Unlike Diffie–Hellman protocol, RSA solution does not require any information to

be sent or exchanged prior to the intended ciphertext. The need for exchanging information beforehand opens a security loophole that if without any authentication mechanism the communications might be manipulated and controlled by an attacker between two parties and thus vulnerable to a so-called *man-in-the-middle attack*. Another valuable feature of RSA algorithm is that it can be used to generate *digital signatures* for verifying the authenticity of messages. Furthermore, RSA is one of the earliest schemes built upon *integer factorisation* problem, which together with discrete logarithm problem form the two most important cornerstones of asymmetric-key cryptography. Since then, a large number of data encryption, digital signature, key exchange, and other public-key cryptographic algorithms have been developed, such as the elliptic curve cryptography [156].

2.3.5 Homomorphic Encryption

The notion of *privacy homomorphisms* was introduced by Rivest *et al.* in 1978 [157], which gave birth to the research on homomorphic cryptosystems and their applications. Privacy homomorphisms can be viewed as particular algebraic mappings between the paintext and ciphertext spaces that allow the result of operations upon the ciphertexts, when deciphered, to match the result of operations upon the plaintexts. As yet another precious legacy of RSA cryptosystem, its *multiplicative homomorphic* property allows multiplication to be operated in the encrypted domain. Suppose that we take two encrypted digits and multiply one with another. The product, when decrypted, is equivalent to the product of two plain digits. If a cryptosystem produces the sum of plain digits through arbitrary operations on encrypted digits, then we say that this cryptosystem has an *additive homomorphic* property. Note that the given operation in the ciphertext domain is not necessarily the same as the resultant operation in the plaintext domain.

Schemes that only possess one of the properties, or do not operate completely on ciphertexts, are said to be *partially homomorphic*, including the Rabin cryptosystem [158], Goldwasser–Micali cryptosystem [159], ElGamal cryptosystem [160], Benaloh cryptosystem [161], Okamoto–Uchiyama cryptosystem [162], Naccache–Stern cryptosystem [163], Paillier cryptosystem [164], and Damgård–Jurik cryptosystem

[165]. By contrast, those that support arbitrary computations on ciphertexts are referred to as *fully homomorphic* cryptosystems [166–176]. Although fully homomorphic schemes are apparently far more powerful, they have been criticised by the tremendous consumption of computational resource and thus the utilisation in practice has been questioned. By contrast, partially homomorphic schemes are more compact and arguably adequate for a number of real-world applications. In this following paragraphs, we discuss two typical partially homomorphic cryptosystems, with which this thesis primarily concerns. In particular, we examine the multiplicative homomorphic property offered by the RSA cryptosystem and the additive homomorphic property given by the Paillier cryptosystem.

RSA Cryptosystem

A well-understood example of multiplicative homomorphism would be the RSA cryptosystem whose security strength is known to be equivalent to the difficulty of solving integer factorisation [99]. Suppose that p and q are two large primes and the modulus is computed as $N = p \cdot q$. Let e and d be the public and private keys of the RSA cryptosystem, respectively, such that e and d satisfy the condition that

$$e \cdot d \equiv 1 \pmod{\phi(N)}, \tag{2.12}$$

where ϕ is Euler's phi function, *i.e.* $\phi(N) = (p-1)(q-1)$. The RSA cryptosystem defines an encryption function

$$c \equiv m^e \pmod{N}, \tag{2.13}$$

and a decryption function

$$m \equiv c^d \pmod{N}, \tag{2.14}$$

where m denotes the message and c denotes the cipher. Consider the goal of generating the encrypted result which, when decrypted, matches the product of two messages m_1 and m_2 through the operations upon the ciphers c_1 and c_2. This goal

can be achieved by multiplying two ciphers as

$$\begin{aligned} c_1 \cdot c_2 &\equiv ({m_1}^e) \cdot ({m_2}^e) \pmod{N} \\ &\equiv (m_1 \cdot m_2)^e \pmod{N}. \end{aligned} \tag{2.15}$$

In other words,

$$\mathcal{D}(\mathcal{E}(m_1) \cdot \mathcal{E}(m_2)) = m_1 \cdot m_2, \tag{2.16}$$

where $\mathcal{E}(\cdot)$ and $\mathcal{D}(\cdot)$ denote the encryption and decryption functions, respectively.

Paillier Cryptosystem

The RSA cryptosystem is limited, however, by solely permitting the multiplication of messages. As a consequence, designing homomorphic encryption algorithms that permit various mathematical operations upon ciphers has been one of the active research areas. Among a variety of historical antecedents, the Paillier cryptosystem is one of the most widely used additive homomorphisms [164]. The system consists of three phases: key generation, encryption, and decryption. In the key generation phase, we choose two large primes p and q. Then, we compute $N = pq$ and $\lambda = \text{lcm}(p-1, q-1)$, where 'lcm' stands for least common multiple. Next, we select a random integer $g \in \mathbb{Z}/N^2\mathbb{Z}^*$ and calculate

$$\mu \equiv (L(g^\lambda \pmod{N^2}))^{-1} \pmod{N}, \tag{2.17}$$

where

$$L(x) = \frac{x-1}{N}. \tag{2.18}$$

The public key is (n, g) and the private key is (λ, μ). In the encryption phase, let m be a message to be encrypted and r be a randomly selected integer, where $m, r \in \mathbb{Z}/N\mathbb{Z}$. The ciphertext is then computed as

$$c \equiv g^m \cdot r^N \pmod{N^2}. \tag{2.19}$$

In the decryption phase, the plaintext message is deciphered by

$$m \equiv L(c^{\lambda} \pmod{N^2}) \cdot \mu \pmod{N}. \tag{2.20}$$

Let m_1 and m_2 be two messages and c_1 and c_2 be two ciphers. To produce the cipher of sum of m_1 and m_2, we calculate the product of c_1 and c_2 and obtain

$$\begin{aligned} c_1 \cdot c_2 &\equiv (g^{m_1} \cdot {r_1}^N) \cdot (g^{m_2} \cdot {r_2}^N) \pmod{N^2} \\ &\equiv g^{(m_1+m_2)} \cdot (r_1 \cdot r_2)^N \pmod{N^2}. \end{aligned} \tag{2.21}$$

To yield the cipher of product of m_1 and m_2, we compute the exponentiation of c_1 by m_2 such that

$$\begin{aligned} c_1^{m_2} &\equiv (g^{m_1} \cdot {r_1}^N)^{m_2} \pmod{N^2} \\ &\equiv g^{(m_1 \cdot m_2)} \cdot (r_1^{m_2})^N \pmod{N^2}. \end{aligned} \tag{2.22}$$

As an alternative expression, we write

$$\mathcal{D}(\mathcal{E}(m_1) \cdot \mathcal{E}(m_2)) = m_1 + m_2, \tag{2.23}$$

and

$$\mathcal{D}(\mathcal{E}(m_1)^{m_2}) = m_1 \cdot m_2. \tag{2.24}$$

It can be observed that the size of encrypted data is expanded as the message space is $\mathcal{M} = \mathbb{Z}/N\mathbb{Z}$ and the ciphertext space is $\mathcal{C} = \mathbb{Z}/N^2\mathbb{Z}^*$.

2.4 Literature Review

The rapid advancement of cloud computing technology has intrigued researchers to explore the possibility of outsourcing the task of reversible watermarking to cloud service providers without compromising the privacy of carrier signals. We have witnessed a dramatic growth in the research of privacy-preserving reversible watermarking over the past few years and a majority of works focused on a particular types of carrier signals, namely digital images. Since reversible watermarking is

usually viewed as a subclass of fragile watermarking, the robustness against attacks such as geometric transformation targeting conventional robust watermarking will not be covered. The focus of privacy-preserving reversible watermarking is primarily the methodology of embedding information into encrypted images or other media. In the rest of this section, we review some of the most representative works in this research area.

2.4.1 Prior Art Based on Symmetric Cryptography

One of the very first reversible information hiding scheme for encrypted images can be traced back to the work by Puech *et al.* who proposed to use block ciphers to encrypt images and embed one bit of information into a block of 4×4 pixels by substituting a significant bit of a selected carrier pixel with the intended bit of payloads [177]. Since encoding was realised through *bit replacement*, the watermark can be decoded directly in the ciphertext domain. Removing the distortion caused by watermark encoding is equivalent to unraveling whether the bit of the selected pixel is originally a zero or an one. This issue was resolved by analysing the *local standard deviation* of the decrypted image block in the two possible cases. As follow-up studies, this idea was refined by an adaptive local entropy analysis [178] and a most significant bit (MSB) predictor [179]. This series of works, however, has an inherent weakness in the fidelity of marked images due to alterations of significant bits of pixels.

One of the most oft-cited works is Zhang's method, which exploits the three least significant bits (LSB) of a block of encrypted pixels to encode one bit of information and recovers the original pixels by a *fluctuation function* that captures spatial correlations in natural images [180]. This design paradigm effectively resolved the issue of low fidelity owing to the fact that only some imperceptible signals are distorted. It method was improved by many follow-up studies including:

- A *side-match* mechanism that involves recovered blocks in the smoothness estimation process [181].
- An evaluation function for estimating pixel distributions of a given block through

analysing the *absolute mean difference* of a given pixel and its adjacent pixels [182].

- A support vector machine (SVM) which handles the problem of image recovery as a *binary classification problem* [183].
- An *elaborate selection* of changeable pixels, instead of altering a whole block of pixels, in order to enhance the visual quality of marked images and a content-based adaptive judging function in light of the fact that pixel fluctuations are minimal along the *isophote direction* [184].
- A *double-round embedding* approach based upon cyclic-shifting and data-swapping for encoding more information [185].
- A *sub-block division* strategy that enables multiple bits, instead of a single bit, to be embedded per block and thus augments the embedding rate [186].

This class of schemes encodes messages mostly through *bit flipping* and therefore the message decoding process was accompanied by the image recovery process in the plaintext domain. In view of this, these schemes are often referred to as *joint schemes*. Message detection in the plaintext domain might, however, limit some potential applications such as encrypted data management that utilises the embedded annotations to supervise and administrate the storage and transfer of encrypted files, and even to protect files against cybersecurity breaches.

In contrast to the class of joint schemes, the notion of *separable schemes* emphasises the separability of message extraction process and signal recovery process. In response to this required property, Zhang proposed to compress encrypted images and append additional messages to the vacated space [187]. The possibility of compressing encrypted data was investigated by Johnson *et al.* ahead of the invention of privacy-preserving reversible information hiding techniques [188], and has undergone profound development over years [189–193]. At first glance, it appears that only a minimal amount of information could be compressed due to the fact that encrypted signals are in general of high randomness and thus high entropy. Nevertheless, the problem of compressing encrypted data can be modelled and solved by *distributed*

source coding. Improvements over this pioneering compression-based scheme include:

- A use of *low-density parity-check* (LDPC) code for approaching an optimal compression rate [194].
- A *three-round embedding* strategy for increasing the embedding rate and a *progressive recovery* mechanism that exploits previously recovered pixels in the future round of recovery [195].
- An extension to deal with encrypted *JPEG bitstreams* [196].

In order to comprehend the concept of distributed source coding in a more concrete sense, let us depict it with the following analogy. Suppose there are two geographically close towns in which a resident of one town, Alice, wishes to compress and communicate the local weather record to a resident of the other town, Bob, using as few bits of information as possible. Since two towns are geographically close to each other, their daily weathers are assumed to be strongly correlated with merely a small probability that they are not of the same weather. We suppose that the weather is either good or bad, which can be represented by 0 and 1, and each day the weather is independent to the past weather. Hence, Alice's weather record can be viewed as an analogy to the encrypted binary image under our assumption of independence of everyday's weather. In other words, the weather record is as random as an encrypted image. Moreover, Bob's weather record can be considered as an analogy to the cryptographic key since it is strongly correlated to Alice's weather record. It may seem a little counter-intuitive and yet it is comprehensible when we take signal redundancy into account and think in a way that the plain binary image is smooth enough so that most of its elements are the same. Thus, when a stream cipher is applied to encrypt the image, the resultant encrypted image would be very similar to the cryptographic key.

Let us exemplify the idea as follows. Suppose that the weathers between Alice's town and Bob's town are correlated in such a way that the difference is no more than one day in every successive three days. In other words, for successive three days, the possible 3-bit difference sequences are: $(0, 0, 0)$, $(1, 0, 0)$, $(0, 1, 0)$, and $(0, 0, 1)$. Suppose Alice weather record is $(1, 1, 0)$ and Bob's weather record is

$(1, 1, 1)$. Let us construct a codebook in which each pair of farthest 3-bit vectors are jointly represented by a 2-bit codeword, as illustrated in Table 2.1. According to the codebook, Alice encode, or compress, her record into a codeword $(0, 1)$ and send it to Bob. At the receiving end, Bob knows that Alice's record must be either $(0, 0, 1)$ or $(1, 1, 0)$ by referring to the codebook. Given the prior assumption that the differences between Alice's and Bob's records are no more than one bit, Bob can confidently determine that Alice's record is $(1, 1, 0)$. It should be kept in mind that this toy example only serves as a conceptual model. In reality, natural signals would be much more complex and therefore we need to apply more powerful codes and make some adjustments to this conceptual construction.

Table 2.1: Codebook

codeword	*vectorpair*
$(0, 0)$	$\{(0, 0, 0), (1, 1, 1)\}$
$(0, 1)$	$\{(0, 0, 1), (1, 1, 0)\}$
$(1, 0)$	$\{(0, 1, 0), (1, 0, 1)\}$
$(1, 1)$	$\{(1, 0, 0), (0, 1, 1)\}$

In literature, the above schemes are often referred to as the class of *vacating room after encryption* (VRAE) in contrast to that of *reserving room before encryption* (RRBE), which is characterised by some *compulsory preprocessing* steps contributing to a comparatively enormous payload capacity. Ma *et al.* utilised *self-embedding* approach to embed some insignificant bits of a part of the image into another part of the image in a reversible manner so that the reserved space can be used to carry additional information even after encryption [197]. The improvements in this line of research mainly focused on more efficient representations of images, including:

- A *sparse coding* technique that encodes images into sparse representations via a pre-trained *K-means singular value decomposition* (K-SVD) based dictionary so that the original images can be reconstructed in presence of the sparse codes and residuals [198].

- An *extended run length code* that efficiently compresses the most significant bit

(MSB) plane [199].

However, this class of schemes would have rather restricted range of applications taking into account the fact that an individual may have too limited computational resource to execute preprocessing algorithms in the first place. In other words, it may constitute a violation of the chief purpose of accessing cloud computing services, in spite of the fact that it could be of good value in some other circumstances and applications.

2.4.2 Prior Art Based on Asymmetric Cryptography

Schemes based upon public-key cryptography involve relatively high computational complexity and often induce non-negligible *ciphertext expansion* problem. It is especially the case when attempting to operate the encryption process pixel by pixel [200–203], due to the fact that public-key cryptography usually involves modular arithmetic with large numbers and thus it would not be efficient to project a small pixel space onto a large ciphertext space. To some extent, schemes adopting ill-constructed encryption procedures would be of limited practical value considering that the expanded file size would be far greater than the payload size in most cases, excluding the standard and inevitable expansion inheres in the given cryptosystem. Furthermore, this class of schemes usually cannot produce a marked plaintext since the payloads are often embedded into the *expanded space* offered by encryption, which evaporates along with decryption. In other words, an intended watermark will be filtered out after decipherment and thus from that point onwards the carrier data will no longer be under the protection of the watermark. Despite the fact that schemes of this type can be suitable for some other potential applications, they would not be applicable when the aim is simply to outsource the task of watermarking to a cloud service provider with expectation of receiving a marked content.

The reserving room before encryption (RRBE) paradigm has also been adopted by schemes compatible with asymmetric-key cryptosystems and has been realised by

- A pre-computation of *reversible integer transform* and a pre-recording of over-

flow/underflow locations enabling the *different expansion* techniques to be operated in the encrypted domain [204].

- A pre-shrinking of image histogram to reconcile the statistical distribution of carrier signals with the subsequent *histogram-shifting* encoding in the encrypted domain [205].

- A *self-embedding* technique to reserve space and an encoding technique based upon *mirroring ciphertext group* strategy to prevent over-saturation of pixels in the plaintext domain [206].

2.5 Summary

In this chapter, we have introduced three fundamental disciplines – information hiding, reversible information hiding, modern cryptography – that form the building blocks of privacy-preserving information hiding systems. Their histories, applications, properties, and other aspects have been concisely reviewed. Although these disciplines have been mostly discussed at an elementary and nontechnical level, a brief grasp of some notions and terminologies would pave the way for our exploration of more detailed and technical principles. We have also reviewed the previous art of privacy-preserving information hiding and classified it by the cryptosystem on which it is based. In the following chapters, we will present our proposed schemes based upon a variety of theoretically secure cryptosystems.

Chapter 3

Information Hiding Based on Symmetric Cryptography

With the global growth in the usage of social media and cloud storage, many companies and organisations are faced with concomitant challenges and security risks. The firms getting compromised will not only suffer from a great financial damage, but also a loss in public credibility. Some of the most dreadful cybersecurity threats would be data breaches, referred to as security incidents in which confidential data is copied, transmitted, viewed, stolen or used by unauthorised parties. ISO/IEC 27040 defines a data breach as:

> *Compromise of security that leads to the accidental or unlawful destruction, loss, alteration, unauthorised disclosure of, or access to protected data transmitted, stored, or otherwise processed.*

Encryption is a potential solution to obfuscate sensitive data against unauthorised viewing and yet it may not stop data from being transferred or exported. Depending upon the importance of the files, it is possible that computationally-intensive cryptanalytic attacks are employed for the decipherments of files after malicious exfiltration. Data saved to the personal cloud accounts of some public celebrities, for example, might be the target of interest. It may also be the case that industrial or political espionages are committed to leak classified documents to adversarial parties. In order to defend against hacking or spying activities, there

is an urgent need for imposing a more secure access control to the encrypted files stored online.

Privacy-preserving information hiding techniques would serve as a potential solution to this issue. It opens up the possibility of annotating cipher data with permission codes in such a way that a message prescribing whether the exportation can take place is embedded into the cipher data and can be detected when transmissions occur. In addition to a permission code, an authentication code can also be embedded for the purpose of tamper proofing. In other words, it offers a two-factor authentication to deter malicious removal of permission codes that attempts to deceive an automated export system to release the files. The system may require the operations of embedding and detection to be carried out in the encrypted domain in order to preserve data privacy throughout storage, retrieval, and transfer.

In this chapter, we present two schemes based upon symmetric cryptography. The proposed schemes permit message extraction in the encrypted domain and therefore are suitable for the applications such as data exfiltration prevention. The first scheme embeds the message by encoding the carrier data as one of the square roots of a quadratic residue, whereas the second scheme embeds the message by encoding the carrier data as one of the possible lexicographic permutations. We would like to note that the schemes are not capable of strictly recovering the images without any loss. We define the reversibility as the degree to which the carrier image can be recovered and measure it by the probability of perfect recovery and the average PSNR of recovered images. Some common properties shared by the two schemes are summarised as follows:

- A synchronous stream cipher is used to encrypted a carrier image. By making computational efficiency a high priority, symmetric-key algorithms are favourable compared to asymmetric-key algorithms.

- Message embedding is based upon a one-to-many mapping. One embedding algorithm utilises square roots of a quadratic residue and the other one utilises lexicographic permutations.

- Message extraction can be carried out in the encrypted domain. This opens up

the possibility to deter unauthorised data transmissions without compromising data privacy to a network administrator.

- Content-adaptive predictors are used to assist the recovery of the carrier image. One predictive model is derived from the projection theorem and the other one is based upon edge gradients.

The remainder of this chapter is organised as follows. Section 3.1 elaborates the proposed method using quadratic residues as well as a predictive model based upon the projection theorem. Section 3.2 presents the proposed method using lexicographic permutations as well as a predictive model based upon image edge gradients. Section 3.4 concludes this chapter and outlines the directions for future research.

3.1 A Scheme Using Quadratic Residues

Privacy-preserving information hiding is a multidisciplinary study that has opened up a great deal of intriguing real-life applications such as data exfiltration prevention, data origin authentication, and electronic data management. Information hiding is a practice of embedding intended messages into carrier signals through imperceptible alterations. In view of some content-sensitive scenarios, however, the ability to preserve perfect copies of signals is of crucial importance, for instance, considering the inadequate robustness of recent artificial intelligence aided automated systems against noise perturbations [101]. Reversibility of information hiding systems is a valuable property that permits a recovery of original carrier signals if desired.

In this section, we present a novel privacy-preserving reversible information hiding scheme inspired by the mathematical concept of quadratic residues [207]. This is one of the first schemes associating reversible information hiding in the encrypted domain with this particular concept in number theory. A quadratic residue has four (not necessarily distinct) square roots and this *one-to-many relationship* between a quadratic residue and its square roots can be utilised to encode payloads in a dynamic fashion. Furthermore, a predictive model based upon the projection theorem is devised to assist the recovery of carrier signals to a perfect state. The design of

cryptographic and watermarking algorithms follows *Shannon's maxim*: 'the enemy knows the system' [149]. In other words, all the detailed construction of algorithms ought to be publicly known and only the keys for decrypting the carrier signal and decoding the watermark remain secret. Experimental results showed significant improvements over the state-of-the-art schemes with respect to three principle factors: capacity, fidelity, and reversibility.

3.1.1 Quadratic Residues

We give a brief introduction to quadratic residues and how to find their square roots [208]. An integer a is called a quadratic residue modulo n if and only if there exists an integer x such that

$$x^2 \equiv a \pmod{n}. \tag{3.1}$$

Otherwise, a is called a quadratic nonresidue modulo n. According to Euler's criterion, if an integer a is relatively prime to an odd prime p, then a is a quadratic residue modulo p if and only if

$$a^{\frac{p-1}{2}} \equiv 1 \pmod{p}. \tag{3.2}$$

and a quadratic nonresidue modulo p if and only if

$$a^{\frac{p-1}{2}} \equiv -1 \pmod{p}. \tag{3.3}$$

This can be expressed concisely by the Legendre symbol:

$$\frac{a}{p} \equiv a^{\frac{p-1}{2}} \pmod{p}. \tag{3.4}$$

The Jacobi symbol generalises the Legendre symbol by considering an odd positive integer modulus n which is not necessary an odd prime. Suppose that n has the prime factorisation $n = p_1 \times p_2 \times \cdots \times p_k$. Then the Jacobi symbol is defined as

$$\frac{a}{n} = \frac{a}{p_1} \times \frac{a}{p_2} \times \cdots \times \frac{a}{p_k}. \tag{3.5}$$

If a is a quadratic residue modulo n, then $(a|n) = 1$. However, the converse does not hold. In other words, a is not necessary a quadratic residue modulo n even if $(a|n) = 1$. Consider the case that an odd positive integer n is factorised into two odd prime p and q. The Jacobi symbol of 1 is possibly the product of two Legendre symbols of -1. That means both $x^2 \equiv a \pmod{p}$ and $x^2 \equiv a \pmod{q}$ have no solutions and thus $x^2 \equiv a \pmod{n}$ has no solutions even though the value of Jacobi symbol is 1. To solve this ambiguity, we need to check whether the value of each Legendre symbol is 1.

We have discussed how to determine whether a quadratic congruence equation is solvable and now we want to find its solutions. It is widely known that factoring a large composite integer is of significant difficulty. The hardness of integer factorisation has formed a cornerstone of a variety of modern cryptosystems such as the Rabin cryptosystem. It is known that Eq. (3.1) is very difficult to solve when n is the product of two large primes p and q. However, if p and q are known, then the Chinese remainder theorem (CRT) can be applied to solve for x. First of all, let us consider an example of the CRT. Let p and q be two relatively prime moduli and α and β be two known integers, the CRT states that there exists an integer x such that

$$\begin{aligned} x &\equiv \alpha \pmod{p}, \\ x &\equiv \beta \pmod{q}. \end{aligned} \tag{3.6}$$

and such x is a unique solution modulo $n = pq$. Now, we want to solve Eq. (3.1) by solving

$$\begin{aligned} x &\equiv x_p \pmod{p}, \\ x &\equiv x_q \pmod{q}. \end{aligned} \tag{3.7}$$

where

$$\begin{aligned} x_p^2 &\equiv a \pmod{p}, \\ x_q^2 &\equiv a \pmod{q}. \end{aligned} \tag{3.8}$$

We can use trial and error to solve for x_p and x_q and express the solutions by

$$\begin{aligned} x_p &\equiv \pm\sqrt{a} \pmod{p}, \\ x_q &\equiv \pm\sqrt{a} \pmod{q}, \end{aligned} \tag{3.9}$$

For odd primes $p, q \equiv 3 \pmod 4$, there exists an efficient formula for solving Eq. (3.8). That is,

$$\begin{aligned} x_p &\equiv \pm a^{\frac{p+1}{4}} \pmod{p}, \\ x_q &\equiv \pm a^{\frac{q+1}{4}} \pmod{q}. \end{aligned} \tag{3.10}$$

The CRT has a unique solution for Eq. (3.7) formulated by

$$x \equiv (x_p \cdot q \cdot b_q + x_q \cdot p \cdot b_p) \pmod{n}. \tag{3.11}$$

where b_q is a unique modular multiplicative inverse of q with respect to the modulus p, and b_p is a unique modular multiplicative inverse of p with respect to the modulus q. That is to say,

$$\begin{aligned} q \cdot b_q &\equiv 1 \pmod{p}, \\ p \cdot b_p &\equiv 1 \pmod{q}. \end{aligned} \tag{3.12}$$

Let us rewrite Eq. (3.12) by

$$\begin{aligned} q \cdot b_q + p \cdot y_p &= 1, \\ p \cdot b_p + q \cdot y_q &= 1. \end{aligned} \tag{3.13}$$

where y_p and y_q are unknown and irrelevant to our problem. Since $\gcd(p, q) = 1$, Eq. (3.13) has the form

$$\alpha x + \beta y = \gcd(\alpha, \beta). \tag{3.14}$$

where α and β are arbitrary integers and x and y are solvable with the extended Euclidean algorithm. Thus, both b_q and b_p in Eq. (3.12) can be solved accordingly. By substituting the solutions for x_p and x_q in Eq. (3.10) into Eq. (3.11), the four

square roots for Eq. (3.1) are obtained by

$$\begin{aligned} x_1 &\equiv (+a^{\frac{p+1}{4}} \cdot q \cdot b_q + a^{\frac{q+1}{4}} \cdot p \cdot b_p) \pmod{n}, \\ x_2 &\equiv (+a^{\frac{p+1}{4}} \cdot q \cdot b_q - a^{\frac{q+1}{4}} \cdot p \cdot b_p) \pmod{n}, \\ x_3 &\equiv (-a^{\frac{p+1}{4}} \cdot q \cdot b_q + a^{\frac{q+1}{4}} \cdot p \cdot b_p) \pmod{n}, \\ x_4 &\equiv (-a^{\frac{p+1}{4}} \cdot q \cdot b_q - a^{\frac{q+1}{4}} \cdot p \cdot b_p) \pmod{n}. \end{aligned} \tag{3.15}$$

3.1.2 Fundamental Mechanisms

In this subsection, we present the fundamental mechanisms that form the building blocks of our proposed privacy-preserving reversible watermarking scheme. We begin by introducing the watermark encoding and decoding mechanisms based upon the Rabin cryptosystem and demonstrate them with a simple example. Then, we discuss the content-adaptive prediction mechanism for assisting image recovery.

Encoding and Decoding Mechanisms

Let us start with the encoding and decoding mechanisms based upon Rabin cryptosystem [158]. The goal is to embed information into an encrypted carrier image. We use stream cipher to encrypt an image and exploit properties of the Rabin cryptosystem to encode the watermark into the encrypted image. We refer to the information to be embedded per round of operation as a *watermark symbol* denoted by w and a random variable of the enciphered carrier image as a *cipher symbol* denoted by c. Note that a cipher symbol is not a pixel. Instead, it is an integer convert from certain bits of a group of selected pixels. We shall see the construction of cipher symbols later.

Let p and q be two distinct prime numbers and $n = pq$ be a modulus. The encryption and decryption functions of Rabin cryptosystem are defined as

$$\text{Encryption} : a \equiv c^2 \pmod{n}, \tag{3.16}$$

and

$$\text{Decryption} : \{\rho_0, \rho_1, \rho_2, \rho_3\} \equiv \sqrt{a} \pmod{n}, \tag{3.17}$$

where c is an input plaintext, a is an output ciphertext, and ρ_i, $0 \leq i \leq 3$ is a possible deciphered result. It can be observed that the decipherment of Rabin cryptosystem is unusual is a sense that it produces four possible answers, though it is not necessary that they are all distinct numbers. In number theory, a is called a *quadratic residue* modulo n and ρ_i is one of its *square roots*. Note that any square root can be encrypted into the same quadratic residue. In addition to this, the chosen prime numbers p and q are required for efficiently calculating the square roots.

The watermark encoding process is carried out as follows. To begin with, we apply Rabin cryptosystem to encrypt c into a and subsequently decrypt a into a set of four possible numbers, $\{\rho_0, \rho_1, \rho_2, \rho_3\}$, in which the numbers are assumed to have been sorted in ascending order, that is, $\rho_0 \leq \rho_1 \leq \rho_2 \leq \rho_3$. Then, we embed w by replacing c with ρ_w resulting $c' = \rho_w$. Consider that at a certain time an authorised party wants to extract w for some intended purposes. With the presence of watermarking key, c' is processed with an encryption and immediately followed by a decryption that yields $\{\rho_0, \rho_1, \rho_2, \rho_3\}$. Finally, the watermark w is determined by matching c' with ρ_w. It is worth pointing out that the number of bits that can be carried may vary in each round of the watermarking operation. There are three different cases to be taken into consideration:

1. If there are four distinct values in the set of square roots, two bits of information can be embedded.

2. If there are two distinct values in the set of square roots, one bits of information can be embedded.

3. If there are only one distinct value in the set of square roots, no information can be embedded at all.

In summary, the number of bits able to be carried is equal to $\log_2 \eta$, where η denotes the number of distinct square roots of a given quadratic residue. Due to the fact that the encryption and decryption functions of Rabin cryptosystem are used in conjunction throughout our scheme, we refer to this conjoint operation as *Rabin transform* for simplicity of notation.

Example. *An example of how to encode and decode the watermark is demonstrated as follows. Consider a* 7*-bit carrier cipher symbol*

$$c = (0100100)_2 = 36.$$

By applying Rabin transform, the resultant square roots in ascending order are

$$\{\rho_0 = 8, \rho_1 = 36, \rho_2 = 41, \rho_3 = 69\}.$$

Given that c *yields* 4 *distinct square roots, we can embed* 2 *bits of information into* c*. Suppose that the intended watermark symbol is*

$$w = (10)_2 = 2.$$

To encode the information, we substitute c *with* ρ_2*, resulting*

$$c' = 41.$$

To decode the information, we compute the Rabin transform of c' *and sort the yielded square roots in ascending order. Finally, by matching* c' *to* ρ_2*, we determine* $w = 2$*.*

Prediction Mechanism

A marked cipher symbol can be recovered into four candidate symbols, although they are not necessarily all distinct numbers. These candidate symbols would then result in four sets of possible values of the original pixels, which will be discussed later. The issue to be addressed at the moment is to distinguish which set of pixels amongst some given sets is more likely to be the original set. It can be realised through developing a *predictive model* that is capable of estimating pixels at certain locations by pixels at other locations. By perceiving an image as a *Markov random field*, a predictive model can generate a *denoised image* in a sense that some contaminated pixels are purified by their neighbouring correlated pixels.

Let us divide pixels of an image into a group of changeable pixels and a group

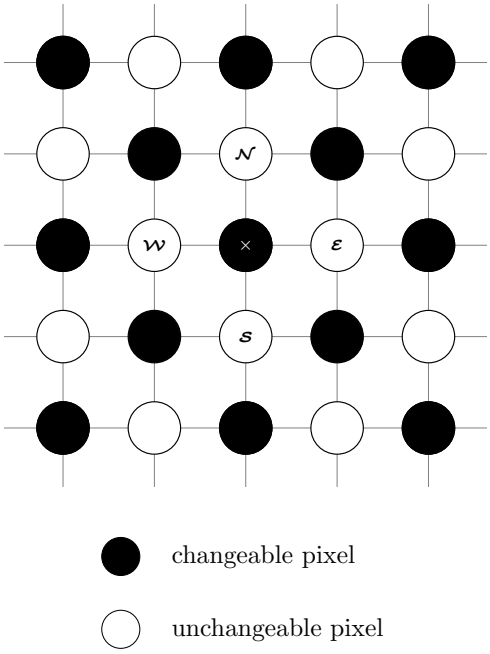


Figure 3.1: Sets of changeable and unchangeable pixels.

of unchangeable pixels in such a fashion that each changeable pixel is encircled by four unchangeable pixels located at its north, south, east and west, as illustrated in Fig. 3.1. The changeable pixels are those used for carrying the payloads and the unchangeable pixels are those used for assisting image recovery. We adopt an efficient but also effective predictive model:

$$\begin{aligned} \widetilde{u} &= \sum_i \psi_i \cdot u_i \\ &= \psi_{\mathcal{N}} \cdot u_{\mathcal{N}} + \psi_{\mathcal{S}} \cdot u_{\mathcal{S}} + \psi_{\mathcal{E}} \cdot u_{\mathcal{E}} + \psi_{\mathcal{W}} \cdot u_{\mathcal{W}}, \end{aligned} \tag{3.18}$$

where $\widetilde{u}$ is an estimated pixel, u_i is an uncontaminated pixel, and ψ_i is a weight of the predictive model. The remaining issue is to compute proper weights that lead to an accurate prediction.

Let $\vec{u}$ denote a column vector of n changeable pixels such that

$$\vec{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}, \tag{3.19}$$

and $\vec{u}_i$ denote a column vector of n unchangeable pixels at corresponding locations, for example,

$$\vec{u}_{\boldsymbol{\mathcal{N}}} = \begin{bmatrix} u_{\boldsymbol{\mathcal{N}},1} \\ u_{\boldsymbol{\mathcal{N}},2} \\ \vdots \\ u_{\boldsymbol{\mathcal{N}},n} \end{bmatrix}. \tag{3.20}$$

An optimal predictive model would be that minimises the L^2 norm:

$$\left\| \vec{u} - \sum_i \psi_i \cdot \vec{u}_i \right\|_2. \tag{3.21}$$

According to Hilbert projection theorem, minimising this norm is equivalent to finding a set of weights such that $\vec{v} = \vec{u} - \sum \psi_i \vec{u}_i$ is orthogonal to $\vec{u}_{\boldsymbol{\mathcal{N}}}$, $\vec{u}_{\boldsymbol{\mathcal{S}}}$, $\vec{u}_{\boldsymbol{\mathcal{E}}}$, and $\vec{u}_{\boldsymbol{\mathcal{W}}}$, respectively. In other words, for a given vector $\vec{u}_{\boldsymbol{\mathcal{N}}}$, we have

$$\vec{u}_{\boldsymbol{\mathcal{N}}}^{\mathsf{T}} \cdot \vec{v} = 0, \tag{3.22}$$

and

$$\vec{u}_{\boldsymbol{\mathcal{N}}}^{\mathsf{T}} \cdot \vec{u} = \vec{u}_{\boldsymbol{\mathcal{N}}}^{\mathsf{T}} \cdot \sum \psi_i \vec{u}_i. \tag{3.23}$$

Let us express the above equation as

$$\begin{bmatrix} u_{\boldsymbol{\mathcal{N}},1} \\ u_{\boldsymbol{\mathcal{N}},2} \\ \vdots \\ u_{\boldsymbol{\mathcal{N}},n} \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} u_{\boldsymbol{\mathcal{N}},1} \\ u_{\boldsymbol{\mathcal{N}},2} \\ \vdots \\ u_{\boldsymbol{\mathcal{N}},n} \end{bmatrix}^{\mathsf{T}} \mathbf{A}_{n\times 4} \begin{bmatrix} \psi_{\boldsymbol{\mathcal{N}}} \\ \psi_{\boldsymbol{\mathcal{S}}} \\ \psi_{\boldsymbol{\mathcal{E}}} \\ \psi_{\boldsymbol{\mathcal{W}}} \end{bmatrix}, \tag{3.24}$$

where

$$\mathbf{A} = \begin{bmatrix} u_{\boldsymbol{\mathcal{N}},1} & u_{\boldsymbol{\mathcal{S}},1} & u_{\boldsymbol{\mathcal{E}},1} & u_{\boldsymbol{\mathcal{W}},1} \\ u_{\boldsymbol{\mathcal{N}},2} & u_{\boldsymbol{\mathcal{S}},2} & u_{\boldsymbol{\mathcal{E}},2} & u_{\boldsymbol{\mathcal{W}},2} \\ \vdots & \vdots & \vdots & \vdots \\ u_{\boldsymbol{\mathcal{N}},n} & u_{\boldsymbol{\mathcal{S}},n} & u_{\boldsymbol{\mathcal{E}},n} & u_{\boldsymbol{\mathcal{W}},n} \end{bmatrix}. \tag{3.25}$$

By deriving the orthogonality for other three vectors $\vec{u}_{\boldsymbol{\mathcal{S}}}$, $\vec{u}_{\boldsymbol{\mathcal{E}}}$, and $\vec{u}_{\boldsymbol{\mathcal{W}}}$ in a similar manner, we have

$$\mathbf{A}^{\intercal}\vec{u} = \mathbf{A}^{\intercal}\mathbf{A}\vec{\psi}, \tag{3.26}$$

where

$$\vec{\psi} = \begin{bmatrix} \psi_{\boldsymbol{\mathcal{N}}} \\ \psi_{\boldsymbol{\mathcal{S}}} \\ \psi_{\boldsymbol{\mathcal{E}}} \\ \psi_{\boldsymbol{\mathcal{W}}} \end{bmatrix}. \tag{3.27}$$

Finally, the weights are given by

$$\vec{\psi} = (\mathbf{A}^{\intercal}\mathbf{A})^{-1}\mathbf{A}^{\intercal}\vec{u}. \tag{3.28}$$

3.1.3 Scheme Constructions

In this subsection, we present detailed procedures based upon the mechanisms discussed previously. The aims and tasks of three parties involved (*i.e.* data owner, cloud server, and end user) are discussed respectively.

Data Owner

Let $\mathbf{m} = \{m_1, m_2, \ldots, m_N\}$ denote an one-dimensional row vector converted from a carrier image such that each element of $\mathbf{m}$ is a pixel. The data owner carries out the encryption procedures as follow:

Step 1. Learn the weight variables $\boldsymbol{\psi} = \{\psi_{\boldsymbol{\mathcal{N}}}, \psi_{\boldsymbol{\mathcal{S}}}, \psi_{\boldsymbol{\mathcal{E}}}, \psi_{\boldsymbol{\mathcal{W}}}\}$ for the predictive model from the given image. The weights are recorded by using 64 bits and may be further compressed into fewer bits.

Step 2. Encrypt each pixel m_i by a stream cipher such that each bit of the pixel is combined with a pseudo-random bit of the keystream via exclusive-or operation. In other words, an enciphered bit is generated by

$$\mathcal{E}(\mathfrak{m}_i^k) = \mathfrak{m}_i^k \oplus \mathfrak{r}_i^k, \tag{3.29}$$

where $\mathfrak{m}_i^k$ denotes the k^{th} bit of the pixel m_i and $\mathfrak{r}_i^k$ denotes a pseudo-random bit defined likewise. In a similar fashion, the cryptographic key is denoted by $\mathbf{r} = \{r_1, r_2, \ldots, r_N\}$ and the enciphered image is denoted by $\mathcal{E}(\mathbf{m}) = \{\mathcal{E}(m_1), \mathcal{E}(m_2), \ldots, \mathcal{E}(m_N)\}$.

Step 3. Encrypt the weights via an arbitrary cipher, resulting $\mathcal{E}(\boldsymbol{\psi})$, and if the watermark is provided by the data owner instead of the cloud server, compress and encrypt also the watermark $\mathcal{E}(\mathbf{w})$. Depending on different applications, the watermark could be the data owner's intended message such as authentication codes or cloud server's auxiliary information such as data annotation.

Step 4. Send the set of encrypted files $\{\mathcal{E}(\mathbf{m}), \mathcal{E}(\boldsymbol{\psi}), \mathcal{E}(\mathbf{w})\}$ to the cloud server for the subsequent task of watermarking.

Cloud Server

Let p and q be two distinct prime numbers chosen by the cloud server and $n = pq$ is the modulus. The cloud server chooses two distinct prime numbers for activating the Rabin transform mechanism, and carries out the message encoding procedures as follows:

Step 1. Sample ℓ changeable encrypted pixels randomly and group them together, yielding $\lambda N/\ell$ groups of pixels in total, where λ denotes the proportion, or the ratio, of changeable pixels. In other words, the total number of groups is equal to the number of changeable pixels divided by the number of pixels sampled each round and thus is represented by $\lambda N/\ell$. The random

sampling is initiated by a random seed, which serves as the *watermarking key*. In other words, the embedded information can only be extracted with the presence of this key; otherwise, an unauthorised party can only employ *brute force attack* to find out the sampling patterns.

Step 2. Extract the t^{th} bit from each pixel in a group to form a cipher symbol in a sense that a symbol is a decimal integer converted from ℓ bits. There are $\lambda N/\ell$ cipher symbols in total, and yet by considering the modular arithmetic involved in Rabin transform, the number of changeable symbols should be represented more precisely by

$$N' = \lambda N/\ell - \epsilon, \tag{3.30}$$

where ϵ denotes the number of symbols whose value exceeds the modulus $n = pq$. Let us denote the changeable cipher symbol used in the current round of watermarking operation by c.

Step 3. Convert $\mathcal{E}(\boldsymbol{\psi})$ and $\mathcal{E}(\mathbf{w})$ into N' watermark symbols in a sense that each watermark symbol w is composed of four, two, or zero bit(s) of information adjusted dynamically according to the capacity of corresponding carrier symbol c. The encoding mechanism is realised by Rabin transform as described previously and produces a marked cipher symbol c'. A collection of all marked cipher symbols is converted in an inverse manner to construct the marked and encrypted image denoted by $\mathcal{E}(\mathbf{m}')$.

Step 4. Send the marked and encrypted image to the intended end user, who could be the data owner if the initial purpose is to outsource the task to the cloud server, or could be the cloud server if the aim is to utilise the annotations to manage encrypted files stored in the cloud and to prevent unauthorised file exportation. It is also possible that the end user is another authorised party with permission to access the image file as well as the watermark information.

It is worth noting that 2^ℓ must not be greater than the modulus n since we

use ℓ bits to form a cipher symbol and feed it into the functions involving modular arithmetic. In practice, we determine ℓ first and then choose proper prime numbers p and q to yield proper modulus n. In addition to this, the parameter ℓ governs the balance between the capacity and reversibility. On the one hand, if a cipher symbol is composed of more pixels, the total number of cipher symbols for carrying message payloads decreases and thus a lower capacity. On the other hand, involving more pixels in the construction of a symbol would reduce the probability of sampling a series of unpredictable pixels. Also note that the parameter t governs the fidelity and is also in charge of the reversibility. Depending on the application in hand, we may choose a smaller value for t if the visual quality of watermarked image is of more concern; otherwise, a larger value for t may be chosen to enhance the probability of recovering the watermarked image to a perfect copy.

End User

There are three different levels of accessibility to be taken into consideration depending on the types of keys available to the end user. Let us demonstrate how the end user can react in three different scenarios:

1) The end user has granted the key $\mathbf{r}$ for decipherment and thus is able to obtain a meaningful marked image denoted by $\mathbf{m}'$.

2) The end user has acquired the watermarking key, namely the locations of pixels in each round of sampling. Hence, by applying the decoding mechanism, the embedded messages $\mathcal{E}(\boldsymbol{\psi})$ and $\mathcal{E}(\mathbf{w})$ can be extracted. We assume that the end user is authorised to decrypt and decompress them into meaningful information.

3) The end user has gained access to both cryptographic and watermarking keys. The aim is not only to obtain the marked image and embedded information, but also to recover the original image. The task of image recovery can be realised with the aid of the previously discussed prediction mechanism. We start by inputing the weights $\boldsymbol{\psi}$ and marked image $\mathbf{m}'$ into the predictive model, which outputs a denoised image $\widetilde{\mathbf{m}}$. We take a copy of the marked and encrypted image $\mathcal{E}(\mathbf{m}')$ and find out $\lambda N/\ell$ groups of changeable pixels via the known sampling patterns.

For a group of ℓ cipher pixels, we convert their t^{th} bits into a cipher symbol c' and apply Rabin transform to generate four corresponding square roots, which are not necessarily all distinct. We replace ℓ bits of each square root with the t^{th} bits of the very group of ℓ pixels respectively, resulting four modified groups of pixels. Then, we decipher them into four groups of plain pixels. Finally, we compare these four candidate groups of pixels with the group of corresponding pixels of the denoised image. The group of pixels that gives the smallest L^1 norm is determined as that of recovered pixels. The recovery procedures are performed iteratively until all the groups of pixels have been properly processed.

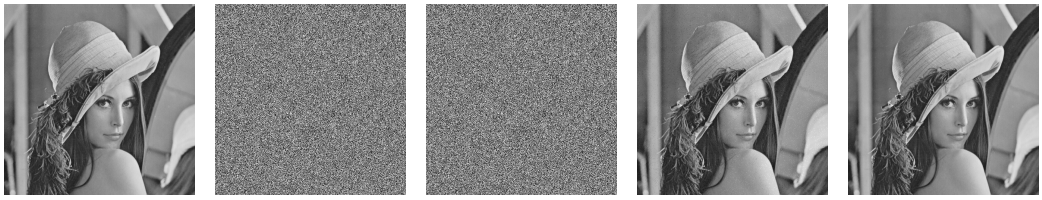
It is worth noting that the reversibility is content-dependent and is also affected by the configurations of parameters ℓ and t. As aforementioned, these parameters play a pivot role in balancing a three-way trade-off between capacity, fidelity, and reversibility. We will examine their impacts in more detail through the following experiments.

3.1.4 Experiments

In this subsection, we examine the scheme's performance with respect to capacity, fidelity, and reversibility. We measure the capacity by the number of bits carried and fidelity by peak signal-to-noise ratio (PSNR). Algorithms are evaluated on standard grayscale test images of size 512×512 widely used across literature, as shown in Fig. 3.2. Images generated from different steps are illustrated in Fig 3.3. This demonstration begins with an original carrier image, whose semantics are later obfuscated by a stream cipher. After that, 4096 bits of information are embedded into it resulting in a marked and encrypted image, which is then decrypted into a meaningful marked image with fidelity of about 33.64 (dB). In the end of this demonstration, the original image is restored. The parameter configurations are $\ell = 8$ and $t = 6$, indicating that each carrier symbol is formed by the 6^{th} bits of 8 changeable pixels. The scheme's performance is evaluated and compared with the prior art including the schemes by Zhang [180], and Liao and Shu [182], and Dragoi et al. [209].

Airplane Lena Peppers Zelda

Figure 3.2: Standard 8-bit test images of size 512×512.



original image encryption watermarking decryption recovery

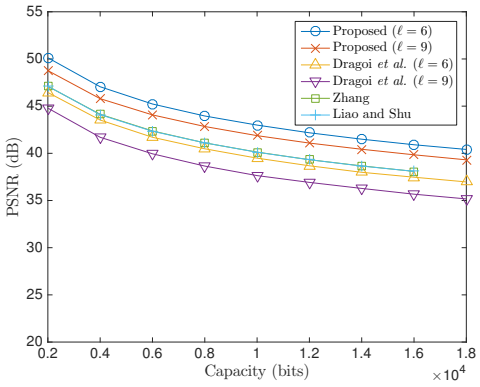
Figure 3.3: Images generated from different steps of process.

Maximum Payload Capacity

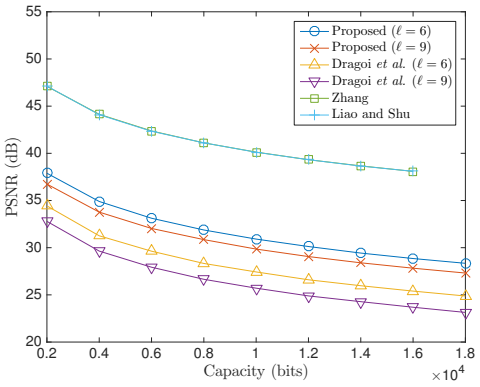
The test results of the watermarking capacity with different configurations of parameters (p, q, and ℓ) are presented in Table 3.1. It is shown that the proposed scheme achieves a larger capacity than [180] and [182] in most cases, and outperforms [209] when the length of symbols (in bits) increases to $\ell \geq 8$. In general, the capacity would decrease as the number of changeable symbols diminishes, namely as ℓ increases. Nonetheless, an interesting observation can be made on the case in which $\ell = 7$. It can be observed that despite a steady decrease of payload bits from $\ell = 6$ to $\ell = 9$, there is a sudden downturn when $\ell = 7$ (*i.e.* 22099 bits). The underlying reason is mainly the infeasible selection of p and q. According to our scheme design, the choice of p and q must satisfy $pq \leq 2^{\ell} - 1$ and a symbol is changeable only if its value in $[0, 2^{\ell} - 1]$ is smaller than the modulus $n = pq$. A large gap between n and $2^{\ell} - 1$ would lead to a great number of unchangeable symbols and hence it is desirable to choose a pair of p and q such that pq is as close to $2^{\ell} - 1$ as possible. However, in the case when $l = 7$, we are not able to find a pair of proper prime numbers that keeps the gap small enough and thus an abrupt drop in terms of the capacity is observed.

Table 3.1: Maximum payload capacity

	Payload (bits)
Proposed ($\ell = 6$, $p = 3$, $q = 19$)	31219
Proposed ($\ell = 7$, $p = 3$, $q = 31$)	22099
Proposed ($\ell = 8$, $p = 11$, $q = 23$)	29998
Proposed ($\ell = 9$, $p = 11$, $q = 43$)	25331
Dragoi *et al.* ($\ell = 6$)	32512
Dragoi *et al.* ($\ell = 7$)	27867
Dragoi *et al.* ($\ell = 8$)	24384
Dragoi *et al.* ($\ell = 9$)	21675
Zhang	16384
Liao and Shu	16384

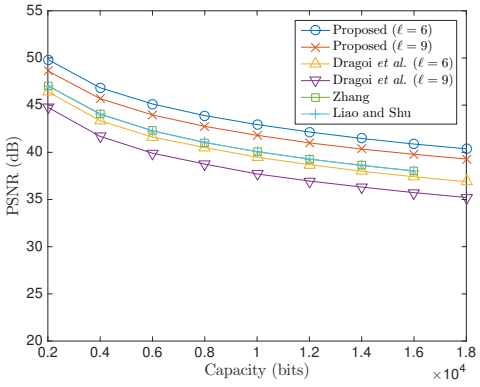


Airplane ($t = 4$)

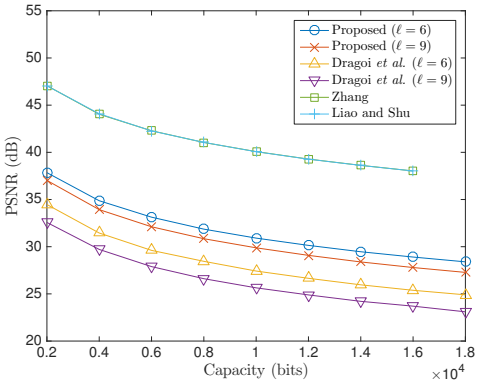


Airplane ($t = 6$)

Figure 3.4: Capacity-fidelity curve (Airplane).

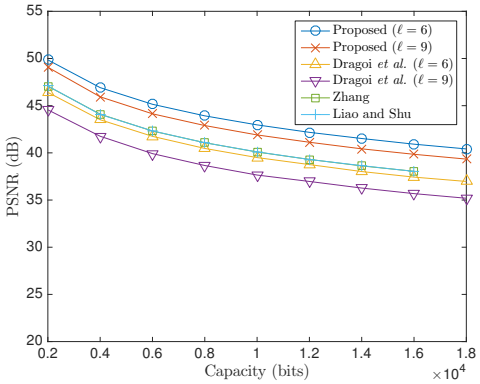


Lena ($t = 4$)

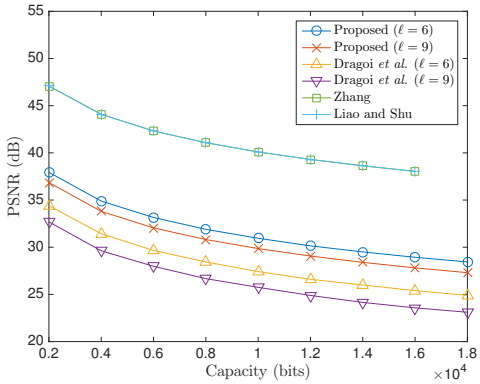


Lena ($t = 6$)

Figure 3.5: Capacity-fidelity curve (Lena).

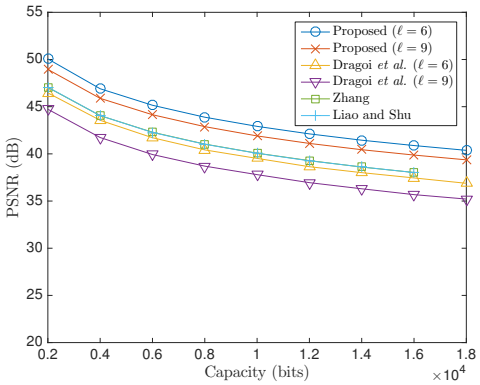


Peppers ($t = 4$)

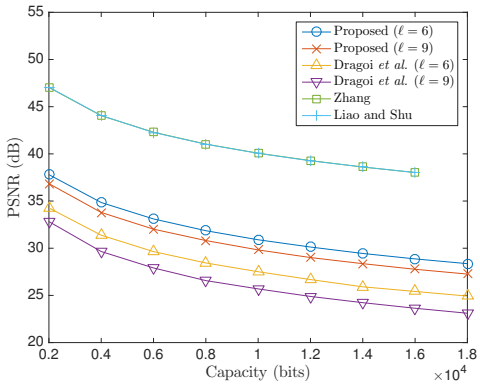


Peppers ($t = 6$)

Figure 3.6: Capacity-fidelity curve (Peppers).



Zelda ($t = 4$)



Zelda ($t = 6$)

Figure 3.7: Capacity-fidelity curve (Zelda).

Capacity-Fidelity Curve

We evaluate the fidelity of marked images under the given capacity. The quality of recovered images will be measured later. The test results of the capacity-fidelity (rate-distortion) curves are shown in Fig. 3.4 to Fig. 3.7. Let the pixels specified by coordinates i and j be denoted by $u_{i,j}$ and its marked version by $u'_{i,j}$. The fidelity of a marked image is measured by

$$\text{PSNR} = 10 \times \log_{10} \frac{255^2}{\text{MSE}}, \tag{3.31}$$

where the mean squared error (MSE) is calculated by

$$\text{MSE} = \frac{\sum_{i=1}^{512}\sum_{j=1}^{512}(u_{i,j} - u'_{i,j})^2}{512 \times 512}. \tag{3.32}$$

It can be seen that a superior rate-distortion performance over the prior art is achieved when the 4^{th} bit plane is set as the *watermarking channel*, namely when $t = 4$. It can also be observed that a striking decline in performance occurs when $t = 6$, causing the proposed scheme and [209] to be inferior to [180] and [182]. The changes in the 6^{th} bit plane cause severe distortions in visual quality of marked images. Nevertheless, choosing a more significant bit plane as the watermarking channel would enhance the reversibility, namely the probability to recover a perfect copy. Overall, our scheme achieves high fidelity due to the fact that the watermarking process does not necessarily alter all the bits of changeable symbols, and attains high capacity by embedding two bits into each changeable symbol for the most part. Hence, under the same fidelity constraint, the proposed scheme reaches a higher embedding rate, and *vice versa*.

Reversibility

The reversibility refers to the ability to recover the altered pixels. As aforementioned, we assumed that the number of changeable pixels is λN, where N is the total number of pixels of a carrier image and λ is the ratio of selected changeable pixels. A more precise value for the number of changeable pixels is calculated by

$$\left\lceil \frac{H-2}{2} \right\rceil \times \left\lceil \frac{W-2}{2} \right\rceil + \left\lfloor \frac{H-2}{2} \right\rfloor \times \left\lfloor \frac{W-2}{2} \right\rfloor, \tag{3.33}$$

where H and W denotes the height and width of an image. Hence, there are in total 130050 changeable pixels for an image of size 512×512.

The reversibility strongly depends on the accuracy of a given predictive model. Table 3.11 presents the weight parameters of the applied predictive model as well as its average prediction error, as calculated by

$$\text{Error} = \frac{\sum_{i=1}^{130050} |u_i - \tilde{u}_i|}{130050}, \tag{3.34}$$

Table 3.2: Average prediction error

	$\psi_{\boldsymbol{\mathcal{N}}}$	$\psi_{\boldsymbol{\mathcal{S}}}$	$\psi_{\boldsymbol{\mathcal{E}}}$	$\psi_{\boldsymbol{\mathcal{W}}}$	Error
Airplance	0.2474	0.2191	0.2786	0.2564	2.84
Lena	0.4046	0.4075	0.0935	0.0954	3.07
Pappers	0.2574	0.2860	0.2461	0.2125	4.03
Zelda	0.4770	0.4760	0.0240	0.0237	2.20

Table 3.3: Recovery rates for each bit plane

	1^{st}	2^{nd}	3^{rd}	4^{th}	5^{th}	6^{th}	7^{th}	8^{th}
Airplance	60.86%	60.57%	75.94%	88.73%	95.42%	98.95%	99.97%	100.00%
Lena	56.78%	56.67%	69.21%	85.72%	95.81%	99.34%	99.98%	100.00%
Pappers	55.14%	55.26%	64.52%	79.26%	93.25%	98.68%	99.87%	100.00%
Zelda	57.50%	57.56%	71.26%	89.92%	99.24%	99.99%	100.00%	100.00%

where u_i denotes the original value of a changeable pixel and $\tilde{u}_i$ denotes its estimated value. Overall, the predictive model is content-adaptive with low prediction error on average. Table 3.3 shows the rates of correct recovery of altered bits by purely considering the *pairwise distances*. The aim is to test and analyse to what extent the predictive model is capable of assisting the recovery of flipped bits.

Let u_i be an original pixel, $\bar{u}_i$ be its altered counterpart with the t^{th} bit being flipped, and $\tilde{u}_i$ be its estimated value generated from the predictive model. We model the problem as to resolve which of the values, u_i or $\bar{u}_i$, is more likely to be the original value. To decide which one is the original pixel value, we calculate their respective distances to $\tilde{u}_i$ and determine the original pixel value as the closer one. In the real context, u_i and $\bar{u}_i$ are two possible candidates to be disambiguated and we do not know which one is the original one. Nonetheless, in order to evaluate the rate of successful recovery, we assume the fact that u_i is the original one is known and hence the rate of successful recovery is computed by

$$\text{Rate} = \frac{\sum_{i=1}^{130050} \tau_i}{130050}, \tag{3.35}$$

where

$$\tau_i = \begin{cases} 1, & \text{if } |u_i - \tilde{u}_i| \leq |\bar{u}_i - \tilde{u}_i|, \\ 0, & \text{otherwise.} \end{cases} \tag{3.36}$$

As anticipated, it is much easier to recover the bits from a significant bit plane than an insignificant bit plane. Note that this recovery rate does not represent the real recovery rate because in the proposed recovery process we calculate the *accumulated distances* for a group of pixels that together form a symbol, instead of considering individually one pixel after another.

Tables 3.4 to 3.7 show the average reversibility from 500 trails of experiments with different payload settings (4096, 8192, and 16384 bits). In each trail, we generate a random payload and assign a new watermarking key for randomly selecting ℓ changeable pixels to form a symbol. The reversibility is measured by the probability of perfect recovery denoted by Pr and the average PSNR of recovered images. In spite of the fact that the watermark extraction process and carrier recovery process are independent in the proposed scheme, it is not the case for the prior art. In the prior art, failing to recover the carrier image causes a mistaken watermark decoding. Thus, we also include the number of incorrect bits being extracted as one of the measurements, as denoted by Bits. It can be observed that the selected t^{th} bit plane plays a pivotal role in the reversibility. It is scarcely possible to restore a perfect copy when embedding payloads into an insignificant bit plane ($t = 4$) and it is more likely to achieve a perfect recovery when a more significant bit plane is used to carry the payloads ($t = 6$). Apart from this, the number of pixels forming a symbol also has substantial impact on carrier signal recovery. By comparing the cases of $\ell = 6$ and $\ell = 9$, it can be seen that the reversibility is significantly higher when more pixels are grouped together to form a symbol. For [180] and [182], a perfect recovery is barely possible. For [209], a superior reversibility can be achieved under small payload setting (4096 to 8192 bits). Nevertheless, the proposed scheme is in general of higher reversibility when large payloads are applied (16384 bits). Furthermore, the proposed scheme is able to extract the watermark bits without any errors.

Table 3.4: Reversibility (Airplane).

Airplane (4096 bits)

	Pr	PSNR	Bits
Proposed ($\ell = 6$, $t = 4$)	0.00	58.10	0.00
Proposed ($\ell = 9$, $t = 4$)	0.00	59.90	0.00
Proposed ($\ell = 6$, $t = 6$)	0.06	64.79	0.00
Proposed ($\ell = 9$, $t = 6$)	0.67	68.51	0.00
Dragoi *et al.* ($\ell = 6$, $t = 4$)	0.00	59.47	50.67
Dragoi *et al.* ($\ell = 9$, $t = 4$)	0.00	64.05	12.16
Dragoi *et al.* ($\ell = 6$, $t = 6$)	0.92	64.43	1.00
Dragoi *et al.* ($\ell = 9$, $t = 6$)	1.00	∞	0.00
Zhang	0.00	51.07	170.86
Liao and Shu	0.00	60.18	16.46

Airplane (8192 bits)

	Pr	PSNR	Bits
Proposed ($\ell = 6$, $t = 4$)	0.00	55.11	0.00
Proposed ($\ell = 9$, $t = 4$)	0.00	56.86	0.00
Proposed ($\ell = 6$, $t = 6$)	0.00	62.37	0.00
Proposed ($\ell = 9$, $t = 6$)	0.40	68.31	0.00
Dragoi *et al.* ($\ell = 6$, $t = 4$)	0.00	56.36	102.86
Dragoi *et al.* ($\ell = 9$, $t = 4$)	0.00	60.20	28.74
Dragoi *et al.* ($\ell = 6$, $t = 6$)	0.90	64.14	1.10
Dragoi *et al.* ($\ell = 9$, $t = 6$)	0.88	62.40	1.09
Zhang	0.00	47.15	1174.80
Liao and Shu	0.00	56.82	130.75

Airplane (16384 bits)

	Pr	PSNR	Bits
Proposed ($\ell = 6$, $t = 4$)	0.00	52.06	0.00
Proposed ($\ell = 9$, $t = 4$)	0.00	53.75	0.00
Proposed ($\ell = 6$, $t = 6$)	0.00	58.82	0.00
Proposed ($\ell = 9$, $t = 6$)	0.18	66.60	0.00
Dragoi *et al.* ($\ell = 6$, $t = 4$)	0.00	52.15	270.65
Dragoi *et al.* ($\ell = 9$, $t = 4$)	0.00	54.84	97.39
Dragoi *et al.* ($\ell = 6$, $t = 6$)	0.00	55.97	7.48
Dragoi *et al.* ($\ell = 9$, $t = 6$)	0.35	60.87	1.69
Zhang	0.00	43.17	3061.40
Liao and Shu	0.00	53.89	258.96

Table 3.5: Reversibility (Lena).

Lena (4096 bits)

	Pr	PSNR	Bits
Proposed ($\ell = 6$, $t = 4$)	0.00	57.47	0.00
Proposed ($\ell = 9$, $t = 4$)	0.00	58.75	0.00
Proposed ($\ell = 6$, $t = 6$)	0.64	67.91	0.00
Proposed ($\ell = 9$, $t = 6$)	0.92	69.26	0.00
Dragoi *et al.* ($\ell = 6$, $t = 4$)	0.00	58.34	65.37
Dragoi *et al.* ($\ell = 9$, $t = 4$)	0.00	62.35	17.67
Dragoi *et al.* ($\ell = 6$, $t = 6$)	0.98	64.43	1.00
Dragoi *et al.* ($\ell = 9$, $t = 6$)	1.00	∞	0.00
Zhang	0.00	54.13	55.30
Liao and Shu	00.0	60.21	12.00

Lena (8192 bits)

	Pr	PSNR	Bits
Proposed ($\ell = 6$, $t = 4$)	0.00	54.47	0.00
Proposed ($\ell = 9$, $t = 4$)	0.00	55.83	0.00
Proposed ($\ell = 6$, $t = 6$)	0.05	65.85	0.00
Proposed ($\ell = 9$, $t = 6$)	0.69	68.85	0.00
Dragoi *et al.* ($\ell = 6$, $t = 4$)	0.00	55.28	132.05
Dragoi *et al.* ($\ell = 9$, $t = 4$)	0.00	58.74	39.98
Dragoi *et al.* ($\ell = 6$, $t = 6$)	0.97	64.43	1.00
Dragoi *et al.* ($\ell = 9$, $t = 6$)	0.89	62.67	1.00
Zhang	0.00	46.58	1232.30
Liao and Shu	0.99	55.47	161.53

Lena (16384 bits)

	Pr	PSNR	Bits
Proposed ($\ell = 6$, $t = 4$)	0.00	51.46	0.00
Proposed ($\ell = 9$, $t = 4$)	0.00	58.81	0.00
Proposed ($\ell = 6$, $t = 6$)	0.00	62.54	0.00
Proposed ($\ell = 9$, $t = 6$)	0.51	68.65	0.00
Dragoi *et al.* ($\ell = 6$, $t = 4$)	0.00	51.54	311.81
Dragoi *et al.* ($\ell = 9$, $t = 4$)	0.00	54.44	106.84
Dragoi *et al.* ($\ell = 6$, $t = 6$)	0.00	56.28	7.06
Dragoi *et al.* ($\ell = 9$, $t = 6$)	0.23	60.69	1.80
Zhang	0.00	43.50	2502.00
Liao and Shu	0.00	52.56	315.26

Table 3.6: Reversibility (Peppers).

Peppers (4096 bits)

	Pr	PSNR	Bits
Proposed ($\ell = 6$, $t = 4$)	0.00	54.40	0.00
Proposed ($\ell = 9$, $t = 4$)	0.00	58.75	0.00
Proposed ($\ell = 6$, $t = 6$)	0.00	62.81	0.00
Proposed ($\ell = 9$, $t = 6$)	0.41	67.65	0.00
Dragoi *et al.* ($\ell = 6$, $t = 4$)	0.00	56.69	95.50
Dragoi *et al.* ($\ell = 9$, $t = 4$)	0.00	59.82	31.31
Dragoi *et al.* ($\ell = 6$, $t = 6$)	0.93	64.23	1.07
Dragoi *et al.* ($\ell = 9$, $t = 6$)	1.00	∞	0.00
Zhang	0.00	52.02	93.40
Liao and Shu	0.00	56.01	35.90

Peppers (8192 bits)

	Pr	PSNR	Bits
Proposed ($\ell = 6$, $t = 4$)	0.00	51.34	0.00
Proposed ($\ell = 9$, $t = 4$)	0.00	51.77	0.00
Proposed ($\ell = 6$, $t = 6$)	0.00	59.25	0.00
Proposed ($\ell = 9$, $t = 6$)	0.26	65.62	0.00
Dragoi *et al.* ($\ell = 6$, $t = 4$)	0.00	53.61	193.58
Dragoi *et al.* ($\ell = 9$, $t = 4$)	0.00	56.04	74.16
Dragoi *et al.* ($\ell = 6$, $t = 6$)	0.87	64.19	1.08
Dragoi *et al.* ($\ell = 9$, $t = 6$)	0.75	62.25	1.14
Zhang	0.00	45.21	1645.5
Liao and Shu	0.00	52.82	282.05

Peppers (16384 bits)

	Pr	PSNR	Bits
Proposed ($\ell = 6$, $t = 4$)	0.00	48.35	0.00
Proposed ($\ell = 9$, $t = 4$)	0.00	48.77	0.00
Proposed ($\ell = 6$, $t = 6$)	0.00	56.07	0.00
Proposed ($\ell = 9$, $t = 6$)	0.04	64.07	0.00
Dragoi *et al.* ($\ell = 6$, $t = 4$)	0.00	49.35	515.57
Dragoi *et al.* ($\ell = 9$, $t = 4$)	0.00	50.94	238.70
Dragoi *et al.* ($\ell = 6$, $t = 6$)	0.00	52.39	16.51
Dragoi *et al.* ($\ell = 9$, $t = 6$)	0.04	57.72	3.52
Zhang	0.00	41.96	3443.70
Liao and Shu	0.00	50.01	537.5

Table 3.7: Reversibility (Zelda).

Zelda (4096 bits)

	Pr	PSNR	Bits
Proposed ($\ell = 6$, $t = 4$)	0.00	63.10	0.00
Proposed ($\ell = 9$, $t = 4$)	0.00	66.17	0.00
Proposed ($\ell = 6$, $t = 6$)	0.96	69.20	0.00
Proposed ($\ell = 9$, $t = 6$)	1.00	∞	0.00
Dragoi *et al.* ($\ell = 6$, $t = 4$)	0.00	61.35	32.91
Dragoi *et al.* ($\ell = 9$, $t = 4$)	0.00	66.92	6.61
Dragoi *et al.* ($\ell = 6$, $t = 6$)	1.00	∞	0.00
Dragoi *et al.* ($\ell = 9$, $t = 6$)	1.00	∞	0.00
Zhang	0.00	60.80	12.46
Liao and Shu	0.00	67.89	3.00

Zelda (8192 bits)

	Pr	PSNR	Bits
Proposed ($\ell = 6$, $t = 4$)	0.00	60.10	0.00
Proposed ($\ell = 9$, $t = 4$)	0.00	63.18	0.00
Proposed ($\ell = 6$, $t = 6$)	0.93	70.06	0.00
Proposed ($\ell = 9$, $t = 6$)	1.00	∞	0
Dragoi *et al.* ($\ell = 6$, $t = 4$)	0.00	58.35	65.41
Dragoi *et al.* ($\ell = 9$, $t = 4$)	0.00	63.13	14.82
Dragoi *et al.* ($\ell = 6$, $t = 6$)	1.00	∞	0.00
Dragoi *et al.* ($\ell = 9$, $t = 6$)	0.92	62.48	1.06
Zhang	0.00	46.80	1122.00
Liao and Shu	0.00	56.74	122.62

Zelda (16384 bits)

	Pr	PSNR	Bits
Proposed ($\ell = 6$, $t = 4$)	0.00	57.10	0.00
Proposed ($\ell = 9$, $t = 4$)	0.00	60.01	0.00
Proposed ($\ell = 6$, $t = 6$)	0.92	70.71	0.00
Proposed ($\ell = 9$, $t = 6$)	0.99	70.20	0.00
Dragoi *et al.* ($\ell = 6$, $t = 4$)	0.00	54.68	151.57
Dragoi *et al.* ($\ell = 9$, $t = 4$)	0.00	59.05	37.31
Dragoi *et al.* ($\ell = 6$, $t = 6$)	0.01	57.05	5.97
Dragoi *et al.* ($\ell = 9$, $t = 6$)	0.40	61.32	1.50
Zhang	0.00	44.03	2142.30
Liao and Shu	0.00	53.90	233.25

3.2 A Scheme Using Lexicographic Permutations

Privacy-preserving reversible watermarking, as a subfield of secure signal processing, has received a growing research attention in the recent years due to privacy concerns in cloud computing. In general, the cloud is assumed to be an honest-but-curious or semi-honest party that is interested in learning the information from the protocol (*e.g.* the plaintext), but does not deviate from the protocol specification. This research problem is challenging since an imperceptible alteration in the ciphertext domain may cause a nontrivial distortion in the plaintext domain. If a cryptosystem is perfectly secure, it is theoretically not possible to foresee how a change in the ciphertext domain would result in a change in the plaintext domain.

There are a variety of possible applications of privacy-preserving reversible watermarking. Consider a network administrator whose responsibility is to monitor data transmissions. It is of crucial importance to prevent classified documents from leakage beyond this point, and yet the privilege to read the documents may not be granted to the administrator. To address this issue, we may encrypt the document and embed a watermark in the ciphertext that indicates the confidentiality and prohibits unauthorised transmission when detected. In the meantime, we also design an algorithm that is able to determine the existence of such a watermark, and if detected, refuses to transmit the ciphertext unless the privilege is demonstrated.

In this section, we present a novel reversible watermarking scheme for data exfiltration prevention. This scheme enables the cloud to embed labels that indicate the degree of confidentiality into the encrypted documents in such a way that the network administrator can monitor the document exfiltration through detecting the labels in the encrypted domain without compromising data privacy. An efficient watermarking algorithm is devised primarily based upon the concept of lexicographic permutations. In addition to this, a content-adaptive signal estimation mechanism is constructed for assisting host media recovery. Experimental results show that the proposed scheme outperforms the state-of-the-art with regards to watermarking capacity, fidelity, and recoverability.

3.2.1 Lexicographic Permutations

To begin with, we introduce a privacy-preserving reversible watermarking scheme based upon lexicographic permutations and then present an updated scheme with detailed discussions. An overview of the proposed scheme is illustrated in Fig. 3.8. The encoding process utilises lexicographic permutations to embed watermarks into encrypted signals, whereas the decoding process extracts the watermarks and recovers the signals in aid of a content-adaptive signal estimation mechanism. In more detail, the transmitter encrypts a host array of symbols $\mathbf{s}$ with a key $\mathbf{k}$ and then uploads the encrypted array of symbols $\mathbf{e}$ to the cloud. A watermark w is encoded into $\mathbf{e}$ producing a marked array $\boldsymbol{\epsilon}_w$ which is then downloaded to the receiver. At the receiving end, w is extracted and a permutation group containing the original encrypted array, denoted by $\{\boldsymbol{\epsilon}_i\}_{i=0}^{t}$, is generated. After decryption, a group containing the original array, denoted by $\{\boldsymbol{\sigma}_i\}_{i=0}^{t}$, is yielded. Eventually, the original array is restored with the assistance of additional information, denoted by $\tilde{\mathbf{s}}$, obtained by a signal estimation mechanism. We would like to emphasise that the restoration is not perfectly lossless and the aim is to achieve the restoration quality as high as possible.

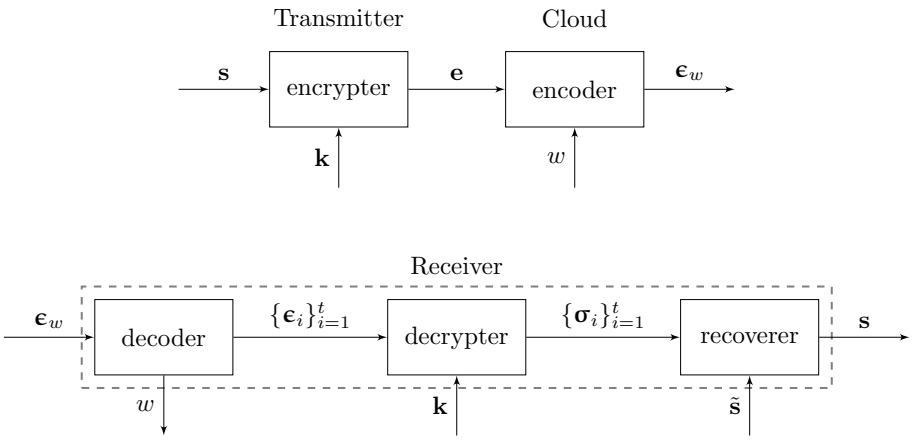


Figure 3.8: An overview of the proposed scheme based on lexicographic permutations.

Consider the host signal as an 8-bit greyscale image. In order to satisfy the fidelity requirement, significant bit-planes should not be modified during watermark embedding process. We assume that the four most significant bit-planes are un-

modifiable, though the scheme permits variations in implementations depending on different fidelity requirements. Let us refer to the remaining four insignificant bit-planes as a nybble-volume, where the basic unit is a nybble, namely, a four-bit aggregation. While this nybble-volume is generally modifiable, only a portion of the nybbles are selected for carrying the payload and the rest part is kept intact for the purpose of reversing watermarking distortions. The selection follows a rule that each selected nybble is encircled by eight unselected immediate neighbours. The unselected nybbles will remain intact and be exploited for estimating the selected nybbles during the reverse process. A nybble can be represented by an integer between 0 and 15. Let a sequence of r modifiable nybbles be converted into an integer, referred to as a host symbol, between 0 and $N-1$, where $N = 2^{4r}$.

Let us divide the host symbols into non-overlapping arrays of length n and each array can be processed independently. Let $\mathbf{s} = (s_1, s_2, \ldots, s_n)$ be an array of modifiable host symbols and $\mathbf{k} = (k_1, k_2, \ldots, k_n)$ be an array of randomly generated key symbols. The transmitter encrypts the former with the latter by

$$\mathbf{e} \equiv \mathbf{s} + \mathbf{k} \pmod{N}. \tag{3.37}$$

Note that the array arithmetic operations are carried out element by element. Then, the transmitter uploads the enciphered array $\mathbf{e} = (e_1, e_2, \ldots, e_n)$ along with the watermark w to the cloud, in which the watermark encoding is realised through lexicographic permutations. Before proceeding further, let us define the number of permutations of a given set. If the set of size n consists of n distinct elements, the number of permutations is simply the factorial of n, denoted by $n!$. If the set consists of repeated elements, then the multiplicity of each element shall be taken into account. Let M be a multiset of size n consisting of l distinct elements and the multiplicities of the elements be m_1, m_2, …, m_l. The number of permutations of M is then given by

$$t = \frac{n!}{m_1! m_2! \ldots m_l!}. \tag{3.38}$$

Let $G_{\mathbf{e}} = \{\boldsymbol{\epsilon}_0, \boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_{t-1}\}$ be a group consisting of all the possible permutations of $\mathbf{e}$ sorted with lexicographic order, where $\boldsymbol{\epsilon}_u = \mathbf{e}$ and $0 \leq u \leq t-1$. A possible

watermarking strategy is to encode a payload of $\log_2 t$ bits into one of the possible permutations. For instance, if an encrypted array $\mathbf{e}$ and a message $0 \leq w \leq t-1$ are encoded into $\boldsymbol{\epsilon}_w$, during the decoding process we can efficiently determine w as the lexicographic order of $\boldsymbol{\epsilon}_w$. Note that the watermark extraction process is carried out in the encrypted domain. As a result, the network administrator can inspect the decoded watermark to decide whether the host file has been given approval to be transmit beyond this point without actually inspecting the file itself. In other words, this scheme prevents data exfiltration without compromising data privacy. Apart from knowing the watermark information, we can also be certain about that $\mathbf{e}$ is one of the possible permutations of $\boldsymbol{\epsilon}_w$, though we are not able to recognise which it is in the absence of further information. At the receiving end, one may want to remove the distortions caused by watermarking. Since it is theoretically not possible to make inferences from the encrypted data, we decipher each possible one by

$$\boldsymbol{\sigma}_i \equiv \boldsymbol{\epsilon}_i - \mathbf{k} \pmod{N}, \tag{3.39}$$

and obtain $G_{\boldsymbol{\sigma}} = \{\boldsymbol{\sigma}_0, \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_{t-1}\}$. As a result, we can employ signal processing techniques to analyse each σ_i and draw an inference on the original one in which some distinguishable structures may inhere. However, failed inferences may occur with high probability when we happen to process a sequence of intrinsically similar host symbols and a sequence of intrinsically similar key symbols. For example, consider $\mathbf{s} = (s_1, s_2)$ and $\mathbf{k} = (k_1, k_2)$ such that $s_1 \approx s_2$ and $k_1 \approx k_2$. We encrypt $\mathbf{s}$ into $\mathbf{e}$ and then encode $\mathbf{e}$ into either $\boldsymbol{\epsilon}_0$ or $\boldsymbol{\epsilon}_1$ depending on whether $w = 0$ or $w = 1$. Assume that $\mathbf{e}$ is of the 0-th permutation order, namely, $\boldsymbol{\epsilon}_0 = \mathbf{e}$, and accordingly

$$\begin{aligned} \boldsymbol{\epsilon}_0 &= (e_1, e_2) = (s_1 + k_1, s_2 + k_2), \\ \boldsymbol{\epsilon}_1 &= (e_2, e_1) = (s_2 + k_2, s_1 + k_1). \end{aligned} \tag{3.40}$$

In order to recover the original permutation, we decrypt respectively $\boldsymbol{\epsilon}_0$ and $\boldsymbol{\epsilon}_1$ into

$\boldsymbol{\sigma}_0$ and $\boldsymbol{\sigma}_1$, as given by

$$\begin{aligned} \boldsymbol{\sigma}_0 &= \boldsymbol{\epsilon}_0 - \mathbf{k} = (s_1, s_2), \\ \boldsymbol{\sigma}_1 &= \boldsymbol{\epsilon}_1 - \mathbf{k} = (s_2 + k_2 - k_1, s_1 + k_1 - k_2). \end{aligned} \tag{3.41}$$

If $s_1 \approx s_2$ and $k_1 \approx k_2$, then $\boldsymbol{\sigma}_0 \approx \boldsymbol{\sigma}_1$. We can observe that in this case $\boldsymbol{\sigma}_0$ is indistinguishable from $\boldsymbol{\sigma}_1$.

3.2.2 Invertible Transform

In the previous scheme, we permute $\mathbf{e}$ lexicographically and obtain a lexicon, or a codebook, for watermark encoding. To overcome the ambiguity in some extreme cases, we update the previous scheme by introducing an invertible transform to $\mathbf{e}$ prior to the creation of the lexicon. Let $\phi(N)$ be Euler's totient function which describes the number of positive integers up to N that relatively prime to N. A positive integer that is coprime to N is termed a totative of N. Suppose that $\mathbf{e}$ is the u-th permutation. We multiply $\mathbf{e}$ with the u-th totative, denoted by p_u, and obtain

$$\mathbf{e}' \equiv \mathbf{e} \cdot p_u \pmod{N}. \tag{3.42}$$

An important property of the above computation is that an inverse transform exists, which is given by

$$\mathbf{e} \equiv \mathbf{e}' \cdot q_u \pmod{N}, \tag{3.43}$$

where q_u is a unique modular multiplicative inverse of p_u with respect to the modulus N, that is,

$$p_u \cdot q_u \equiv 1 \pmod{N}. \tag{3.44}$$

A unique modular multiplicative inverse q_u exists if and only if p_u is coprime to N, that is, $\gcd(p_u, N) = 1$, where gcd stands for greatest common divisor. The number of permutations of $\mathbf{e}'$ is also t since the transform from $\mathbf{e}$ to $\mathbf{e}'$ is a bijective mapping.

We sort the permutations of $\mathbf{e}'$ lexicographically and form an ordered lexicon $G_{\boldsymbol{\epsilon}} = \{\boldsymbol{\epsilon}_0, \boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_{t-1}\}$ and encode a watermark w of $\log_2 t$ bits into one of the permutations yielding the marked result $\boldsymbol{\epsilon}_w$. In the decoding phase, w can be

efficiently recognised by the order of $\boldsymbol{\epsilon}_w$. Let $\{p_0, p_1 \ldots, p_{t-1}\}$ be the first t totatives in $[0, N]$, and $\{q_1, \ldots, q_t\}$ be their respective modular multiplicative inverses. To restore the original array, we choose any $\boldsymbol{\epsilon}_i$ from the lexicon and multiply it with each multiplicative inverse yielding $G_{\boldsymbol{\alpha}} = \{\boldsymbol{\alpha}_0, \boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_{t-1}\}$, where

$$\begin{aligned} \boldsymbol{\alpha}_0 &\equiv \boldsymbol{\epsilon}_i \cdot q_0 \pmod{N}, \\ \boldsymbol{\alpha}_1 &\equiv \boldsymbol{\epsilon}_i \cdot q_1 \pmod{N}, \\ &\ldots \\ \boldsymbol{\alpha}_{t-1} &\equiv \boldsymbol{\epsilon}_i \cdot q_{t-1} \pmod{N}. \end{aligned} \tag{3.45}$$

Then, we sort the elements in each array with the lexicographic order in accordance to the array index and yield an updated group of arrays $G_{\boldsymbol{\beta}} = \{\boldsymbol{\beta}_0, \boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_{t-1}\}$. For instance, the elements in $\boldsymbol{\alpha}_i$ is sorted with the i-th lexicographic order yielding $\boldsymbol{\beta}_i$. Note that the choice of $\boldsymbol{\epsilon}_i$ does not affect the resultant $G_{\boldsymbol{\beta}}$; in other words, any $\boldsymbol{\epsilon}_i$ yields the same group of results. The u-th array in $G_{\boldsymbol{\alpha}}$ is the original encrypted array with scrambled elements, whereas the u-th array in $G_{\boldsymbol{\beta}}$ is exactly the original encrypted array, namely $\boldsymbol{\beta}_u = \mathbf{e}$. Since it is not possible to distinguish the original array in the encrypted domain, we decipher each array in $G_{\boldsymbol{\beta}}$ and obtain $G_{\boldsymbol{\sigma}} = \{\boldsymbol{\sigma}_0, \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_{t-1}\}$. With the aid of signal analysis techniques, we can retrieve the original one, namely $\boldsymbol{\sigma}_u$, with relatively low error rate. Let us see how this updated scheme is able to resolve the aforementioned ambiguity. Again, consider two host symbols and two key symbols such that $s_1 \approx s_2$ and $k_1 \approx k_2$. Assume that $\mathbf{e}$ is of the 0-th permutation order and accordingly $\mathbf{e}' = \mathbf{e} \cdot p_0$. Then, we encode $\mathbf{e}'$ into either $\boldsymbol{\epsilon}_0$ or $\boldsymbol{\epsilon}_1$ depending on the watermark bit. In the recovering phase, two possible candidates are generated by

$$\begin{aligned} \boldsymbol{\sigma}_0 &= \boldsymbol{\beta}_0 - \mathbf{k} = \mathrm{sort}(\boldsymbol{\alpha}_0, 0) - \mathbf{k} = \mathrm{sort}(\boldsymbol{\epsilon}_i \cdot q_0, 0) - \mathbf{k}, \\ \boldsymbol{\sigma}_1 &= \boldsymbol{\beta}_1 - \mathbf{k} = \mathrm{sort}(\boldsymbol{\alpha}_1, 1) - \mathbf{k} = \mathrm{sort}(\boldsymbol{\epsilon}_i \cdot q_1, 1) - \mathbf{k}, \end{aligned} \tag{3.46}$$

where $\mathrm{sort}(\mathbf{x}, i)$ denote a sorting function that sorts the elements of an array $\mathbf{x}$ according to the i-th lexicographic permutation, and $\boldsymbol{\epsilon}_i$ can be either $\boldsymbol{\epsilon}_0$ or $\boldsymbol{\epsilon}_1$. We

further derive that

$$\text{sort}(\boldsymbol{\epsilon}_i \cdot q_0, 0) - \mathbf{k} = ((s_1 + k_1)p_0q_0 - k_1, (s_2 + k_2) \cdot p_0q_0 - k_2) = (s_1, s_2), \quad (3.47)$$

and $\boldsymbol{\sigma}_1$ equals to either

$$\text{sort}(\boldsymbol{\epsilon}_i \cdot q_1, 1) - \mathbf{k} = ((s_1 + k_1)p_0q_1 - k_1, (s_2 + k_2) \cdot p_0q_1 - k_2), \quad (3.48)$$

or

$$\text{sort}(\boldsymbol{\epsilon}_i \cdot q_1, 1) - \mathbf{k} = ((s_2 + k_2)p_0q_1 - k_1, (s_1 + k_1) \cdot p_0q_1 - k_2). \quad (3.49)$$

In either case, two candidates are not likely to be similar since the term p_uq_v, where $u \neq v$, thoroughly randomise the incorrect candidate; in other words, the original array should be very distinguishable from a sequence of random numbers with high probability.

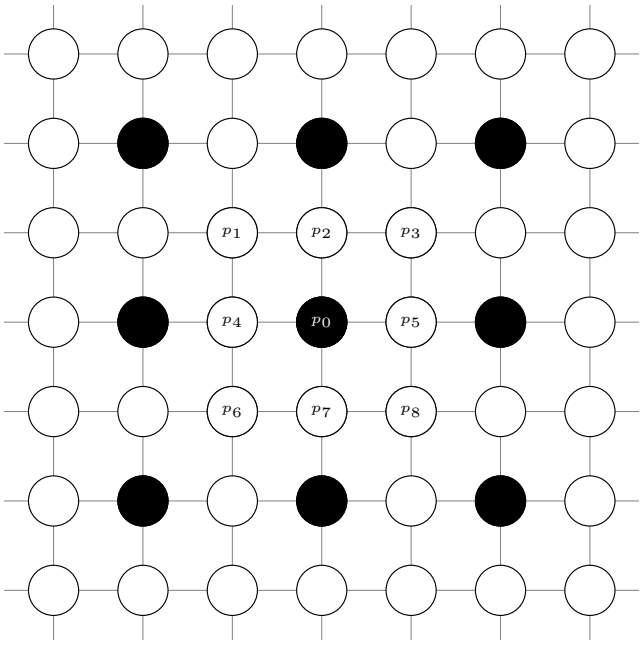


Figure 3.9: Pixels at the block positions are modifiable in terms of their low nybbles, whereas those at the white positions are unmodifiable.

3.2.3 Predictive Model

To complete the proposed scheme, we devise a signal estimation mechanism for assisting host signal recovery. As aforementioned, for an 8-bit greyscale image we embed payloads into selected low nybbles while each of which is encircled by eight unselected immediate neighbours, as illustrated in Fig. 3.9. The aim is to estimate the low nybble of a pixel p_0 with the aid of the high nybble of p_0 and the neighbouring pixels $p_1, p_2, \ldots, p_8$. Image regions can be roughly divided into smooth, edge-carrying, and highly textured patches. Due to the fact that the statistical distribution of pixel values varies a lot in different regions, we have to identify the class the observed p_0 belongs to. Let

$$M(X) = \frac{1}{\gamma} \sum_{i=1}^{\gamma} |x_i - \mu(X)| \tag{3.50}$$

represent the mean absolute deviation (MAD), where $\mu(X)$ represents the arithmetic mean and γ represents the number of elements in a given set X. The score for smooth patches is given by

$$\delta_{smth} = M(p_1, p_2, \ldots, p_8), \tag{3.51}$$

and the scores for different degrees of edges are given by

$$\begin{aligned}
\delta_{0^\circ} &= \frac{M(p_1, p_2, p_3) + M(p_4, p_5) + M(p_6, p_7, p_8)}{3}, \\
\delta_{45^\circ} &= \frac{M(p_2, p_4) + M(p_3, p_6) + M(p_5, p_7)}{3}, \\
\delta_{90^\circ} &= \frac{M(p_1, p_4, p_6) + M(p_2, p_7) + M(p_3, p_5, p_8)}{3}, \\
\delta_{135^\circ} &= \frac{M(p_2, p_5) + M(p_1, p_8) + M(p_4, p_7)}{3}.
\end{aligned} \tag{3.52}$$

Let the minimum value of $\{\delta_{smth}, \delta_{0^\circ}, \delta_{45^\circ}, \delta_{90^\circ}, \delta_{135^\circ}\}$ be denoted by δ_{min}. If δ_{min} is no greater than a threshold θ (empirically $\theta = 15$), then we calculate an anticipated

value for p_0 by

$$\tilde{p}_0 = \begin{cases} \mu(p_1, p_2, \ldots, p_8) & \text{if } \delta_{min} = \delta_{smth}, \\ \mu(p_4, p_5) & \text{if } \delta_{min} = \delta_{0^\circ}, \\ \mu(p_3, p_6) & \text{if } \delta_{min} = \delta_{45^\circ}, \\ \mu(p_2, p_7) & \text{if } \delta_{min} = \delta_{90^\circ}, \\ \mu(p_1, p_8) & \text{if } \delta_{min} = \delta_{135^\circ}. \end{cases} \tag{3.53}$$

Otherwise, an anticipated value for p_0 is determined by the closest value in the neighbouring area, that is,

$$\tilde{p}_0 = \arg\min_{p_i} \mid p_i - p_0 \mid, \tag{3.54}$$

where $i \in \{1, 2, \ldots, 8\}$. Finally, we estimate p_0's low nybble in such a way that the resultant pixel value approaches the anticipated value $\tilde{p}_0$, as formulated by

$$\tilde{p}_0^* = \arg\min_{p_{0,j}} \mid p_{0,j} - \tilde{p}_0 \mid, \tag{3.55}$$

where $j \in \{0, 1, \ldots, 15\}$ and $p_{0,j}$ denotes a value generated by setting p_0's low nybble to one of the possible patterns.

3.2.4 Experiments

A series of experiments were carried out for validating and evaluating the proposed scheme. Images of size 512×512 with 256 greylevels are used as the host media, as shown in Fig. 3.10. The scheme utilises a synchronous stream cipher to encrypt images and is therefore semantically secure, as presented in Fig. 3.11. Let the number of host symbols in each array be fixed to $n = 4$. Each symbol is formed by r modifiable nybbles, where r is set from 2 to 5, and correspondingly the symbol values lie in $[0, 31]$, $[0, 63]$, $[0, 127]$, $[0, 255]$, and $[0, 511]$, respectively. The experimental results are shown in Table 3.8. Since the total number of modifiable nybbles is invariable, the total number of host symbols decreases when each symbol is formed by more nybbles, which explains the inverse proportionality between watermarking

capacity and r. We can also observe that the error rate of image recovery reduces and the fidelity of recovered images improves as r increases. When a symbol is made of more nybbles, it is more likely that it consists largely of well-estimated nybbles, which is of crucial importance to the recovery process. Let $p_{i,j}$ denote the pixel at the i-th row and the j-th column, and $\hat{p}_{i,j}$ denote its noisy counterpart. We evaluate the image quality by peak signal-to-noise ratio (PSNR) defined as

$$\text{PSNR} = 10 \times \log_{10} \frac{255^2}{\text{MSE}}, \tag{3.56}$$

where mean square error (MSE) is calculated by

$$\text{MSE} = \frac{1}{512 \times 512} \sum_{i=1}^{512} \sum_{j=1}^{512} (p_{i,j} - \hat{p}_{i,j})^2. \tag{3.57}$$

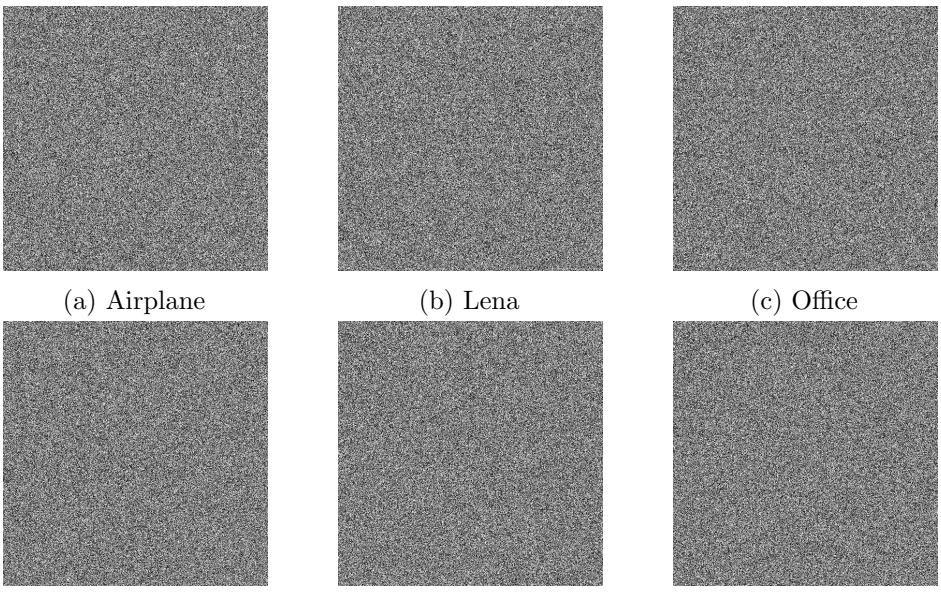
Fig. 3.12 shows the marked images generated by embedding the payload into all the modifiable nybbles. It is shown that the PSNRs of various test images are all no less than 40.7 dB. Fig. 3.13 shows the recovered images generated by adjusting the embedding rate to 0.057 bits per pixel (bpp). The results show that the PSNRs are all greater than 51.9 dB.

We compare the proposed scheme with the state-of-the-art schemes [182, 187, 209] in terms of watermarking capacity, fidelity, and recoverability. As can be seen from Fig. 3.14, the proposed scheme outperforms the previous methods with regard to the fidelity of marked images under the same embedding rate. As reported in Fig. 3.15, the proposed scheme also achieves the best results with respect to the fidelity of recovered images given the same amount of payload. Overall, it is evident that the proposed scheme achieves a substantial improvement in algorithm performance.

(a) Airplane (b) Lena (c) Office

(d) Sail (e) Wine (f) Zelda

Figure 3.10: Test images.



(a) Airplane (b) Lena (c) Office

(d) Sail (e) Wine (f) Zelda

Figure 3.11: Encrypted images.

(a) Airplane (b) Lena (c) Office

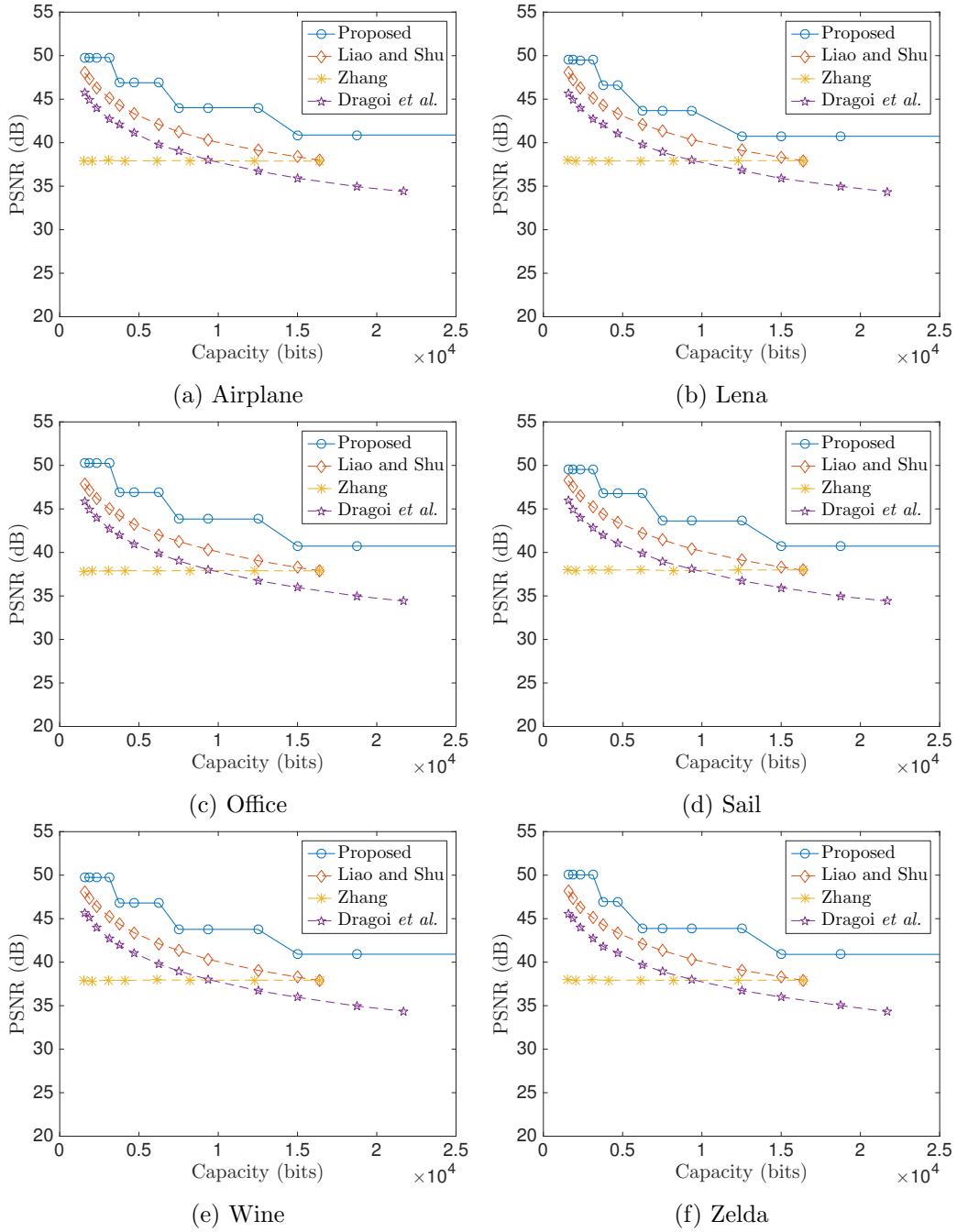
(d) Sail (e) Wine (f) Zelda

Figure 3.12: Marked images: (**a**) Airplane, PSNR= 40.88 dB. (**b**) Lena, PSNR= 40.74 dB. (**c**) Office, PSNR= 40.72 dB. (**d**) Sail, PSNR= 40.71 dB. (**e**) Wine, PSNR= 40.91 dB. (**f**) Zelda, PSNR= 40.90 dB.

(a) Airplane (b) Lena (c) Office

(d) Sail (e) Wine (f) Zelda

Figure 3.13: Recovered images: (**a**) Airplane, PSNR= 52.87 dB. (**b**) Lena, PSNR= 51.96 dB. (**c**) Office, PSNR= 54.01 dB. (**d**) Sail, PSNR= 53.17 dB. (**e**) Wine, PSNR= 53.75 dB. (**f**) Zelda, PSNR= 52.39 dB.

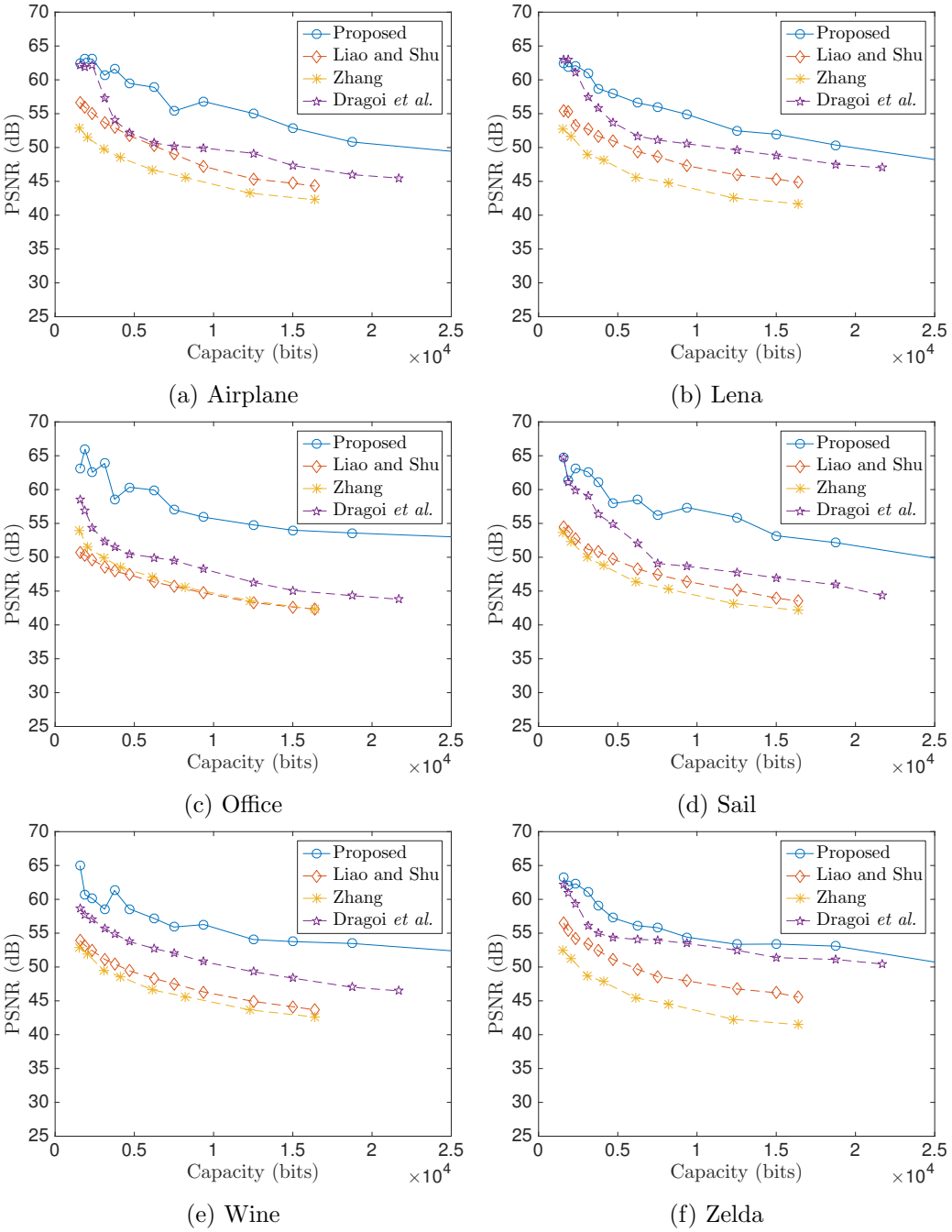


(a) Airplane (b) Lena

(c) Office (d) Sail

(e) Wine (f) Zelda

Figure 3.14: Comparison with the state-of-the-art in terms of fidelity. The horizontal axis displays the watermarking capacity, whereas the vertical axis shows the PSNR of marked images.



(a) Airplane (b) Lena

(c) Office (d) Sail

(e) Wine (f) Zelda

Figure 3.15: Comparison with the state-of-the-art with regard to recoverability. The horizontal axis depicts the watermarking capacity, whereas the vertical axis presents the PSNR of recovered images.

Table 3.8: Scheme performance at different number of nybbles per symbol ($r = 2$, 3, 4, and 5). WR is watermarking rate (in bpp), ER is error rate (in bpp), and PSNR is peak signal-to-noise ratio (in dB).

	$r = 2$			$r = 3$		
	WR	**ER**	**PSNR**	**WR**	**ER**	**PSNR**
Airplane	0.143	0.094	46.86	0.095	0.049	49.43
Lena	0.143	0.120	45.95	0.095	0.066	48.21
Office	0.143	0.041	50.65	0.095	0.023	50.02
Sail	0.143	0.096	46.97	0.095	0.047	49.84
Wine	0.143	0.051	49.84	0.095	0.027	52.39
Zelda	0.143	0.088	47.77	0.095	0.042	50.72
	$r = 4$			$r = 5$		
	WR	**ER**	**PSNR**	**WR**	**ER**	**PSNR**
Airplane	0.072	0.034	50.81	0.057	0.021	52.87
Lena	0.072	0.040	50.36	0.057	0.026	51.96
Office	0.072	0.020	53.59	0.057	0.017	54.01
Sail	0.072	0.027	52.16	0.057	0.020	53.17
Wine	0.072	0.020	53.48	0.057	0.019	53.75
Zelda	0.072	0.022	53.08	0.057	0.020	53.39

Table 3.9: Comparisons of two proposed schemes with respect to PSNR of marked images and recovered images under the same capacity.

Capacity: 4096 bits

	Airplane		Lena		Mandrill	
	QUADRA	LEXICO	QUADRA	LEXICO	QUADRA	LEXICO
	($l = 9$, $t = 6$)	($n = 3$, $r = 3$)	($l = 6$, $t = 6$)	($n = 6$, $r = 3$)	($l = 9$, $t = 6$)	($n = 6$, $r = 1$)
marked images	33.69	44.43	34.73	47.13	33.70	51.47
recovered images	∞	66.27	∞	65.35	59.20	51.89

Capacity: 8192 bits

	Airplane		Lena		Mandrill	
	QUADRA	LEXICO	QUADRA	LEXICO	QUADRA	LEXICO
	($l = 6$, $t = 6$)	($n = 4$, $r = 3$)	($l = 7$, $t = 6$)	($n = 6$, $r = 1$)	($l = 8$, $t = 6$)	($n = 6$, $r = 1$)
marked images	31.79	42.70	31.30	48.31	30.35	48.49
recovered images	67.44	64.28	∞	59.63	54.28	49.08

Capacity: 16384 bits

	Airplane		Lena		Mandrill	
	QUADRA	LEXICO	QUADRA	LEXICO	QUADRA	LEXICO
	($l = 9$, $t = 6$)	($n = 5$, $r = 2$)	($l = 9$, $t = 6$)	($n = 6$, $r = 1$)	($l = 8$, $t = 6$)	($n = 6$, $r = 1$)
marked images	27.73	42.12	27.66	45.41	27.33	45.51
recovered images	67.44	56.75	∞	55.01	52.48	46.19

3.3 Comparisons of Two Proposed Schemes

In this section, we compare the performance of two proposed schemes in terms of the following two aspects:

- visual quality of marked images and recovered images under the same capacity;
- execution time.

The quadratic residue-based method (QUADRA) is parameterised by l and t, representing that a host symbol is formed by the t-th bits of l pixels. The visual quality of marked images decreases when a higher bit-plane is used to form the host symbols (*i.e.* greater t) and on the other hand the visual quality of recovered images increases. The lexicographic permutation-based method (LEXICO) is parameterised by r denoting the number of low nibbles to form a host symbol and n denoting the number of host symbols to form a permutation group. The embedding capacity grows with the number of possible permutations, which is associated with the number of host symbols in a permutation group.

In Table 3.9, it can be observed that under the same capacity constraints (4096, 8192, and 16384 bits respectively), LEXICO achieved higher PSNR of marked images, while QUADRA yielded higher PSNR of recovered images. In addition to this, QUADRA realised perfect recovery in some cases in contrast to LEXICO. For each method, we evaluated the encoding time (watermark embedding) and decoding time (watermark extraction and image recovery) given 4096 payload bits. As shown in Table 3.10, the execution time of QUADRA did not have noticeable fluctuations under different settings of parameters, which contrasted sharply with that of LEXICO. The decoding time of LEXICO grew with n. It can also be seen from Table 3.11 that LEXICO took much more time for watermark extraction and image recovery when embedding more bits of information.

Table 3.10: Evaluation of execution time (in second)

	parameters	encoding time	decoding time
QUADRA	$l = 6$, $t = 4$	0.73	2.22
	$l = 6$, $t = 5$	0.69	2.20
	$l = 6$, $t = 6$	0.69	2.19
	$l = 7$, $t = 4$	0.77	2.47
	$l = 7$, $t = 5$	0.77	2.39
	$l = 7$, $t = 6$	0.78	2.41
	$l = 8$, $t = 4$	0.68	2.59
	$l = 8$, $t = 5$	0.67	2.55
	$l = 8$, $t = 6$	0.69	2.50
	$l = 9$, $t = 4$	0.73	2.65
	$l = 9$, $t = 5$	0.73	2.64
	$l = 9$, $t = 6$	0.73	2.68
LEXICO	$n = 2$, $r = 1$	1.35	2.28
	$n = 2$, $r = 2$	1.13	2.45
	$n = 2$, $r = 3$	1.17	2.79
	$n = 3$, $r = 1$	0.67	1.75
	$n = 3$, $r = 2$	0.56	1.73
	$n = 3$, $r = 3$	0.54	1.96
	$n = 4$, $r = 1$	0.46	2.42
	$n = 4$, $r = 2$	0.38	2.68
	$n = 4$, $r = 3$	0.37	2.88
	$n = 5$, $r = 1$	0.40	7.27
	$n = 5$, $r = 2$	0.33	9.59
	$n = 5$, $r = 3$	0.33	10.01

Table 3.11: Growth of execution time (in second)

parameters	encoding time		decoding time	
capacity	4096	16384	4096	16384
$n = 2$, $r = 1$	1.35	4.72	2.28	8.84
$n = 2$, $r = 2$	1.13	4.60	2.45	9.86
$n = 2$, $r = 3$	1.17	3.10	2.79	7.50
$n = 3$, $r = 1$	0.67	2.51	1.75	6.75
$n = 3$, $r = 2$	0.56	2.20	1.73	7.15
$n = 3$, $r = 3$	0.54	2.29	1.96	7.86
$n = 4$, $r = 1$	0.46	1.77	2.42	9.78
$n = 4$, $r = 2$	0.38	1.55	2.68	10.84
$n = 4$, $r = 3$	0.37	1.52	2.88	11.87
$n = 5$, $r = 1$	0.40	1.55	7.27	30.76
$n = 5$, $r = 2$	0.33	1.34	9.59	39.04
$n = 5$, $r = 3$	0.33	1.32	10.01	40.73
$n = 6$, $r = 1$	0.40	2.41	7.27	226.76
$n = 6$, $r = 2$	0.33	2.28	9.59	307.42
$n = 6$, $r = 3$	0.33	2.28	10.01	317.41

3.4 Summary

In this chapter, two novel reversible information hiding schemes are introduced for embedding watermarks into encrypted images in cloud computing environments. These techniques can be used for data breach protection and also other potential real-world applications. We adopt a secure and efficient symmetric-key cipher for image encryption.

The first scheme utilises the arithmetic of quadratic residues for watermark embedding. In addition to this, a content-adaptive predictive model based upon the projection theorem is devised to fulfil the requirement of recovering the original copy. Experimental results show that in most cases the proposed scheme outperforms the prior art in terms of capacity, fidelity and reversibility.

The second scheme embeds payloads into encrypted images via lexicographic permutations. We derive further an updated version of the scheme in order to minimise the error rate in host recovery. In addition to this, a content-adaptive signal estimation mechanism based upon image edge gradients is devised for assisting the carrier signal recovery process. Experimental results show a remarkable breakthrough over the state-of-the-art in capacity, fidelity, and recoverability.

It is expected that the research in this field will continue to move forwards. From our perspective, privacy-preserving reversible information hiding will find more real-world applications in the near future because it has evolved the classical information hiding to address security and privacy issues in many new technologies. Its reversibility is also a desirable feature for many artificial intelligence aided automated systems in which the available perfect copies are of great significance to the system performance. Minimisation of error rate in carrier signal recovery entails further investigation.

Chapter 4

Information Hiding Based on Asymmetric Cryptography

Sitting in the intersection of watermarking and cryptography, privacy-aware reversible watermarking permits a party to entrust the task of embedding to a cloud service provider without compromising information privacy. The early development of schemes were primarily based upon traditional symmetric-key cryptosystems, which involve an extra implementation cost of key exchange. Although recent research attentions have been drawn to schemes compatible with asymmetric-key cryptosystems, there are notable limitations in the practical aspects. In particular, the host signal must either be enciphered in a redundant way or be pre-processed prior to encryption, which would largely limit the storage efficiency and scheme universality. To relax the restrictions, we propose a novel research paradigm and devise different schemes compatible with different asymmetric-key homomorphic cryptosystems. In the proposed schemes, the encoding function works in a way similar to noise addition, whereas the decoding function can be perceived as a corresponding denoising process. Both online and offline content-adaptive predictors are developed to assist watermark decoding for various operational requirements. A three-way trade-off between the capacity, fidelity and reversibility is analysed mathematically and empirically.

4.1 Privacy-Preserving Reversible Information Hiding

The past decades have witnessed the phenomenal prevalence of public cloud services. The seemingly unlimited storage and computational capacity of the cloud have opened up opportunities for businesses and individuals to entrust their data to cloud service providers. Meanwhile, the rapid development of cloud computing technology has also raised privacy and security concerns. A powerful solution against cyber security threats is to obfuscate the classified documents through encryption [149]. On the other hand, although encryption algorithms are widely used to protect the confidential information, some desirable functionalities might be unrealisable for the encrypted data. In response to this issue, there was a surge of research interest in secure signal processing in the encrypted domain [7, 8, 18]. As an emerging topic in this research field, privacy-aware watermarking has attracted a lot of attention in recent years. For the reason that many watermarking algorithms are proprietary properties, there are restrictions for commercial purposes and the availability to the general public is rather limited. An efficient solution is to request an authorised cloud service provider to encode the watermark into the given digital media, not only due to the high computational capacity of the cloud, but also the availability of the intellectual property right or license to carry out the algorithms. On the other hand, since the cloud server is regarded as a semi-honest entity that may collect some information from the digital contents, data privacy should be taken into account. Therefore, privacy-aware watermarking is intended to the problem of entrusting the task of watermarking to a cloud server without compromising data privacy. This research trend is also known as watermarking in the encrypted domain.

In general, there are three parties involved in a watermarking protocol: a sender who encodes the watermark into the host media, a recipient who decodes the watermark from the marked media, and an adversary who has a malicious intent against the protocol. The malicious party is often modelled as a noisy channel between the encoder and the decoder. The watermarking strategy can be either fragile or robust against the channel noise depending on the applications. Conventionally, robust watermarking is applicable to copyright protection [55, 58, 60,

64], whereas fragile watermarking is advantageous to data authentication [68, 73, 75, 77]. Nevertheless, fragile watermarking, as a temper detection technique, contradict itself by inevitably inflicting distortions upon the content of media. Although the distortions are generally imperceptible, the effect could be arguable in particular circumstances featuring a strict integrity requirement, especially when the host media is used for military reconnaissance or medical diagnosis. Considering these potential applications, the notion of reversible watermarking was introduced in order to permit the reproduction of the original content once the authenticity of media is verified [104, 118, 119, 125, 126, 210–212].

Consider a watermarking protocol as illustrated in Fig. 4.1, where the sender Alice wants to communicate a digital content to the receiver Bob over an insecure channel. In order to resist malicious tampering, Alice has to embed a signature, which is sensitive against manipulations, into the content prior to dispatching it to Bob. Supposing that Alice, as a general individual, has no permission or resources to employ the watermarking algorithm, the task of watermarking has therefore to be entrusted to a licensed cloud server, *e.g.* Charlie. As a general presumption, Charlie is an honest-but-curious party who would not deviate from the protocol specifications, though on the other hand has the interest in learning the privacy information of the content. Hence, due to privacy concerns, Alice encrypts the content prior to commissioning it to Charlie for watermarking. In summary, to utilise the complementary advantages of watermarking and cryptography, it is of great significance to develop a secure reversible watermarking system that enables the watermarking function to be operated by a licensed third party in the encrypted domain in order to preserve privacy.

The challenge of reversible watermarking in the encrypted domain is a rather difficult one. Considering that the message is concealed by encryption, we are not able to observe, analyse, and exploit data redundancy. The early development of schemes was primarily compatible with symmetric-key cryptosystems [177, 181, 187, 188, 190, 192, 194, 195, 213–215]. These schemes are confronted with a practical issue that a secret key must be pre-shared between the sender and the recipient. In order to communicate the secret key, a secure channel resistant to eavesdropping or

a secure key exchange protocol must be established. Any of these solutions comes with extra costs.

In contrast to symmetric-key algorithms, asymmetric-key algorithms, also known as public-key cryptosystems, relax the restriction by using a publicly distributed key for encryption and a secret key known only to the recipient for decryption. Over the last few years, the watermarking research has extended to manage ciphers generated by public-key algorithms. The first class of strategy encrypts the data in such a manner that the input to the encryption function has a value range that is far smaller than it should be. Consider that each individual pixel is an input to be enciphered. The value range of pixels is between 0 and 255 for 8-bit greyscale images, while the message space of a given cryptosystem is, for example, between 0 and N, where N is the product of two large primes. Therefore, the input space defined by the watermarking shceme is much smaller than the actual message space defined by the cryptosystem. In implementation, a number of zeros are concatenated to the end of the input message to assure the length of input conform with message space. As a consequence, zero paddings can be viewed as additional redundancy for accommodating the payload in the encrypted domain [200–202]. It is nevertheless deficient for the following reasons. First, redundant bits are appended and thus the system is not space-efficient. Second, it is considered problematic since the amount of padded redundancy is at least equivalent to that of the payload in most cases. One may even simply append the payload information after the host signal in the plaintext domain, rather than performing complex computation to trade the pre-appended redundancy for the payload in the ciphertext doamin. Third, it is insecure since the watermark can be easily filtered out by decryption. To overcome these deficiencies, we suggest to encrypt a bit-stream of sufficient length each time, instead of a single sample with zero-padding.

The second class of strategy, though following the space-saving principle, requires a specific type of pre-processing to be applied prior to encryption. It utilises either truncation [205] or self-embedding [206] to shrink the range of sample values and then encrypts a sufficient number of samples each time. As a result, although the input space seems to be equal to the message space defined by the cryptosystem,

the actual input space is much smaller. For instance, initially the sample values range from 0 to 255 and yet subsequent to data pre-processing the possible values are reduced to a subrange. The reserved space can then be exploited to accommodate the payload in the encrypted domain. This class of strategy, also known as reserving room before encryption, was first developed for watermarking schemes to work in conjunction with symmetric-key cryptosystems [197, 198, 216]. The practicality of this strategy is however limited because a specific type of pre-processing must be applied to the data. To remove this restriction, we suggest to encrypt data without any specific pre-processing.

In this chapter, we address the problem of entrusting the task of watermarking to a cloud service provider without compromising information privacy. Considering the security threats of symmetric-key cryptosystems, we study reversible watermarking schemes compatible with public-key cryptosystems. The recent development of schemes requires the host data either to be enciphered in a redundant fashion or to be pre-processed prior to encryption. In response to this, we propose a novel research paradigm to overcome the aforementioned shortcomings. Two schemes are proposed to cope with ciphers generated by two types of homomorphic cryptosystems, respectively. The first scheme is constructed based upon multiplicative homomorphism and provides high content fidelity as well as high flexibility in watermarking capacity. The second scheme is devised based upon additive homomorphism and achieves an optimal reversibility. In addition to this, both online and offline content-adaptive predictors are introduced to assist watermark decoding. The former type of predictor is based upon variational method and utilises an iterative algorithm to approximate the signal, while the second type of predictor is based upon statistical inference and pre-trains a probability table for signal estimation. An accurate predictor can greatly improve the rates of correct watermark extraction and host signal recovery. Experimental results shows that the proposed predictors achieve the state-of-the-art performance. In summary, several positive contributions and improvements are made, which are briefly outlined as follows:

- Modern public-key cryptosystems are adopted to avoid the security risks and implementation costs of key exchange.

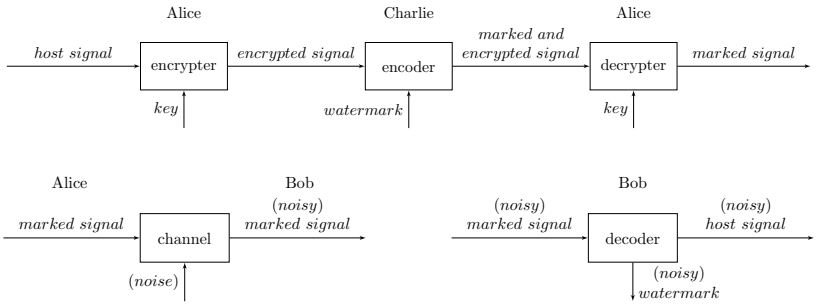


Figure 4.1: The proposed privacy-preserving reversible information hiding protocol. A transmitter Alice wants to protect a signal through embedding a watermark, which can be fulfilled by a cloud service provider Charlie. A marked signal is then transmitted over an insecure channel to a recipient Bob who is able to verify the authenticity of the received signal through analysing the extracted watermark and reproduced signal.

- Reversible watermarking schemes compatible with different types of partially homomorphic cryptosystems are studied.
- Storage efficiency is considered by encrypting a sufficiently long sequence of bits, instead of a single sample with redundant zero padding, each time.
- Specific type of data pre-processing prior to encryption is not required, which enhances the practicality and universality.
- Both online and offline content-adaptive predictors are developed with flexibility for various operational requirements.

The remainder of this chapter is organised as follows. Section 4.2 discusses how to apply public-key cryptosystems to encrypt digital images. Section 4.3.1 introduces the proposed watermarking scheme for multiplicative homomorphic cryptosystem. Section 4.3.2 further derives a scheme for additive homomorphic cryptosystem. Section 4.4 studies content-adaptive predictors based upon variational method and statistical inference. Section 4.5 evaluates the scheme performance and makes comparisons with state-of-the-art algorithms. Section 4.6 concludes our work and outlines the directions for future research.

4.2 Image Encryption

As aforementioned, when encrypting a digital image, it is not advisable to consider each pixel as an individual message. Generally, the value range of pixels is far smaller than the message space, as the latter is associated with large prime numbers. If a pixel is treated as a independent message to be encrypted, then a long sequence of zero bits will be padded after the bits of the pixel. Encrypting every pixel independently provokes serious *ciphertext expansion*. Apart from this, if the watermark is embedded into the padded bits created for accommodating the encryption algorithm, it will normally be filtered out by decryption. In some applications, this feature would not be desirable. There are several possible approaches to encrypt digital images without expanding the image files, for example, converting the image into a bit stream and encrypting a segment of sufficient length each time. In order to preserve the visual significance, we propose to encipher each bit-plane separately. Let $\mathbf{I}$ denote an 8-bit digital image such that

$$\mathbf{I} = (\mathbf{b}_1||\mathbf{b}_2||\dots||\mathbf{b}_8), \tag{4.1}$$

where '||' is the concatenation operator and $\mathbf{b}_1$, $\mathbf{b}_2$, ..., and $\mathbf{b}_8$ are eight different bit-planes. For each bit-plane

$$\mathbf{b}_i = (b_{i,1}||b_{i,2}||\dots), \tag{4.2}$$

we sample a sufficiently long bit-stream each time to form a message, which is then encrypted into a cipher. Supposing that the message space of a given cryptosystem is $\mathbb{Z}/N\mathbb{Z}$, a message is therefore a decimal number of $\log_2 N$ bits, or of $\lfloor \log_2 N \rfloor$ bits more precisely, considering that the former is not necessarily an integer.

The watermarking process is performed sequentially upon the selected messages until all the payload is embedded, while the unselected ones remain intact. Due to the fidelity constraint, we only change the messages converted from insignificant bit-planes. Intuitively, it seems that the optimal choice should be to select messages of the least significant bit-plane. However, by considering the decoding process, it is necessary that the selected messages are formed by some well-predictable bits. We

shall see the reason behind this later. To pave the way for the following presentation, let us assume that the messages for watermarking have been determined. The map to record the locations of selected messages serves as the watermarking key. In the following section, we will discuss separately how to embed watermarks into encrypted data generated by different types of cryptosystems.

4.3 Schemes Using Privacy Homomorphisms

In this section, we present a privacy-preserving reversible watermarking scheme for RSA-like cryptosystem and a scheme for Paillier-like cryptosystem.

4.3.1 RSA-Based Scheme

Let us recapitulate the problem statement. Let Alice denote a sender, Bob a recipient, and Charlie a cloud server. Suppose that Alice wants to deliver to Bob a digital file in which a watermark is embedded for authentication purposes. Due to various constraints (*e.g.* proprietary issues), Alice has to entrust the task of watermarking to Charlie by providing the encrypted file and the watermark. The encryption key is publicly known, while the decryption key is only known to Bob. We assume that the watermark payload is a sequence of compressed and encrypted digits such that only the intended recipient Bob is able to decode.

Consider the objective of embedding watermarks into ciphers generated by a multiplicative homomorphic cryptosystem such as RSA. To begin with, the host data is divided into binary sequences of the length in accordance with the message space of the given cryptosystem. Then, each sequence is transformed into an integer called a symbol. After encryption, a sufficient number of enciphered symbols are selected to carry the watermark payload. We suppose that the selected symbols are all composed of bits in the l-th bit-plane. To meet the fidelity requirement, the change of bits in the l-th bit-plane must not cause perceivable degradation on the visual quality of the image. For conciseness, we describe only how a single host symbol is processed, as shown in Fig. 4.2.

Let p and q be two large primes and $N = p \cdot q$. The alphabet of watermark

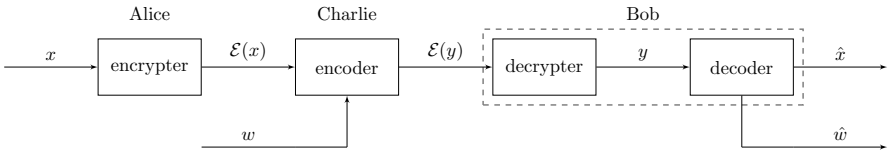


Figure 4.2: Watermarking procedures for a selected symbol. Let x be a host symbol and w be a watermark symbol. Alice encrypts x into $\mathcal{E}(x)$. Charlie encodes $\mathcal{E}(x)$ with w into $\mathcal{E}(y)$. Bob decrypts $\mathcal{E}(y)$ into y and then decodes it into $\hat{x}$ and $\hat{w}$.

symbols is defined as the set of k distinct positive integers up to N that are coprime to N. Accordingly, we are able to embed $\log_2 k$ bits of watermark information per host symbol. The number of positive integers up to N that are relatively prime to N can be calculated by Euler's phi function $\phi(N) = (p-1)(q-1)$ and thus $k \leq \phi(N)$. This alphabet, denoted by $\mathcal{W} = \{w_i\}_{i=0}^{k-1}$, is pre-shared between Alice and Bob and serves as a codebook so that different payloads can be represented by different watermark symbols. It is worth noting that revealing this pre-shared alphabet to an attacker does not compromise the security since the alphabet can be viewed as merely a pre-defined mapping between binary watermark digits to a special set of integers. Let a host symbol be $x \in \mathbb{Z}/N\mathbb{Z}$ and a watermark symbol be $w \in \mathcal{W}$. To begin with, Alice encrypts x into $\mathcal{E}(x)$ and uploads it along with $\mathcal{E}(w)$ to Charlie. The watermarking process is then operated by

$$\mathcal{E}(y) \equiv \mathcal{E}(x) \cdot \mathcal{E}(w) \equiv \mathcal{E}(w \cdot x) \pmod{N}. \tag{4.3}$$

After that, the marked and encrypted symbol $\mathcal{E}(y)$ is downloaded and sent to Bob and is then decrypted into

$$y \equiv \mathcal{D}(\mathcal{E}(y)) \equiv w \cdot x \pmod{N}. \tag{4.4}$$

Note that we strictly let $w_0 = 1$ in practice in order to minimise the average distortion. That is, when $w = w_0$, the marked symbol will be kept intact. The distortion only occurs when $w = w_i$, $\forall i \neq 0$, is embedded. We know that there exists a unique modular multiplicative inverse of an arbitrary integer a modulo N if and only if

$\gcd(a, N) = 1$. Since $\mathcal{W}$ is a subset of all integers that are coprime to N, we can construct the set $\overline{\mathcal{W}} = \{w_i^{-1}\}_{i=0}^{k-1}$ that comprises of the corresponding unique inverses. We generate k possible candidates for x in such a way that each is given by

$$x_i \equiv y \cdot w_i^{-1} \pmod{N}. \tag{4.5}$$

The above congruence can be rewritten as

$$x_i \equiv x \cdot w \cdot w_i^{-1} \pmod{N}. \tag{4.6}$$

Thus, we see that $x_i = x$ if and only if $w_i^{-1} = w^{-1}$. In other words, if we are able to distinguish x from k possible candidates $\{x_i\}_{i=0}^{k-1}$, then we can determine w jointly. Recall that a selected symbol is composed of bits that can be estimated with a certain degree of accuracy from some other correlated bits. Let us denote an estimated symbol for x by $\tilde{x}$. Therefore, the original x is determined by

$$\hat{x} = \arg\min_{x_i} \Delta(x_i, \tilde{x}), \tag{4.7}$$

where Δ is a general distortion metric (*e.g.* Hamming distance). The reversibility, namely the ability to recover the host media, primarily depends on two factors. One is the number of candidates and the other is the estimation accuracy. The reason is straightforward that given only a few possibilities and a highly credible clue, there is a high probability that the answer is correctly deduced. The number of candidates is governed by the capacity parameter k, whereas the estimation accuracy is related to the fidelity parameter l. Hence, we summarise that

$$\begin{aligned} \text{Capacity} &\propto k, \\ \text{Fidelity} &\propto l^{-1}, \\ \text{Reversibility} &\propto l \cdot k^{-1}. \end{aligned} \tag{4.8}$$

Example. *Let us demonstrate the scheme with an example as follows. Assume that*

$p = 11$, $q = 19$, *and* $N = 209$. *Let a payload bit be encoded by*

$$w = \begin{cases} w_0, & \textit{if the bit is } 0, \\ w_1, & \textit{if the bit is } 1, \end{cases}$$

where w_0 *and* w_1 *can be any two numbers coprime to* 209. *Note that* w_0 *is strictly set to one in practice and we let* $w_1 = 3$ *in this example. Suppose that Alice has a host symbol* $x = 100$ *and wants to embed a bit one. As a result, the watermark symbol is given by* $w = w_1$. *By encryption, Alice outputs* $\mathcal{E}(x) = \mathcal{E}(100)$. *After watermarking, Charlie produces* $\mathcal{E}(y) \equiv \mathcal{E}(3) \cdot \mathcal{E}(100) \bmod 209$. *By decryption, Bob obtains* $y = 91$, *which is computed by*

$$91 \equiv 3 \cdot 100 \bmod 209.$$

The modular multiplicative inverses of w_0 *and* w_1 *are* $w_0^{-1} = 1$ *and* $w_1^{-1} = 70$, *respectively. Hence, two candidates for* x *are* $x_0 = 91$ *and* $x_1 = 100$, *which are computed by*

$$91 \equiv 91 \cdot 1 \bmod 209,$$
$$100 \equiv 91 \cdot 70 \bmod 209,$$

respectively. Suppose that the approximation of x *is* $\tilde{x} = 102$. *We distinguish the original value of* x *from a given set of candidates by choosing the candidate closest to* $\tilde{x}$ *in terms of the Hamming distance, and determine the value of* w *correspondingly. The distances are calculated by*

$$\Delta_0 = \Delta(91, 102) = 5,$$
$$\Delta_1 = \Delta(100, 102) = 1,$$

and therefore the decoding results are $\hat{x} = 100$ *and* $\hat{w} = 3$.

It can be observed that the wrong candidate is, theoretically, a random integer. Thus, if N is sufficiently large and the number of candidates to distinguish from is relatively small, then it is unlikely that randomly drawn integers would be closer to the approximation of the correct one than the correct one itself.

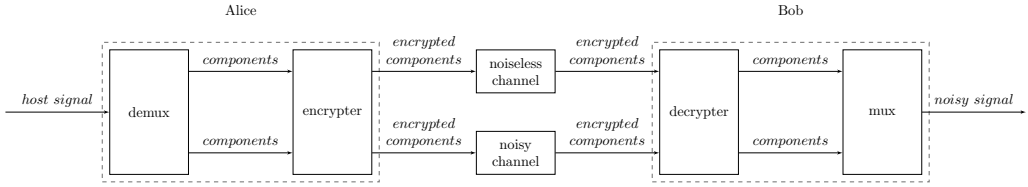


Figure 4.3: Watermarking as noise adding. Alice demultiplexes and encrypts the host signal into encrypted components and then sends each component over either a noiseless or noisy channel. Bob decrypts and multiplexes the received components into a noisy signal.

4.3.2 Paillier-Based Scheme

We have seen that the reversibility of the previous scheme is related to the minimum distance between possible candidates and the correct one. This distance is random in the case when the multiplicative homomorphism is permitted. To achieve a better performance, it is desirable to create the distances that are always far enough. We found that it is possible to construct such a function under an additive homomorphism. Let us consider embedding watermarks into encrypted data produced by an additive homomorphic cryptosystem such as the Paillier cryptosystem. For this type of cryptosystem, there exists an optimal strategy to maximise the Hamming distance between candidates in a special case that only one bit is embedded per symbol and there are only two candidates to be distinguished from in the decoding process (*i.e.* $k = 2$). Certainly, there is always a trade-off. In this case, it is the non-trivial data expansion from the message space to the cipher space due to homomorphic encryption and a little compromise on the fidelity due to distance maximisation.

Before proceeding further, let us introduce another viewpoint of this research problem, as diagramed in Fig. 4.3. Let Alice be a sender and Bob be a recipient. There are two communication channels between the two parties: a noiseless channel and a noisy channel. Let us assume that the noiseless channel is much more expensive than the noisy one, which complies with our intuition. Suppose that a message can be decomposed into several pieces of components and each component is transmitted over one of the channels. Alice wants the communication cost to be as low as possible. Bob, on the other hand, requires that the core idea of the message must be clear

and comprehensible regardless of some errors induced by the channel noise. In other words, the noisy components must not lead to the misunderstanding upon the core idea of the message. Moreover, Bob hopes that the errors can be corrected from the context given by the other correct components. Suppose that each message component is concealed in a safe envelope during the transmission. Nevertheless, the noisy channel is still capable of adding noise to the components even without opening the envelope, imagining that raindrops pass through the envelope and wet the letter.

This research problem is analogous to that of reversible watermarking in the encrypted domain in the following senses. First, the noisy channel is analogous to the watermarking function. Second, the reciprocal of the communication cost corresponds to the watermarking capacity. Third, the requirement of preserving the core idea of the message is equivalent to the fidelity constraint of watermarking. Last, the context-based error correction is akin to the decoding process that jointly detects the payload and recovers the host media. The task of error correction will be easy if the noisy channel always modifies the components to those of the opposite meaning in the presence of noise. The reason is that given a certain context, it is much easier to sense an antonym of a word than a synonym of that word. Hence, our objective is to construct a noisy channel that always maps components to their farthest counterparts in the presence of noise, that is to say,

$$\mathcal{E}(y) = \begin{cases} \mathcal{E}(x), & \text{if no noise occurs,} \\ \mathcal{E}(\bar{x}), & \text{otherwise,} \end{cases} \tag{4.9}$$

where $\bar{x}$ is the farthest counterpart of x. Let x be a non-negative integer composed of bits from a certain bit-plane and $\bar{x}$ be the bitwise complement of x. Although the Paillier cryptosystem does not permit homomorphic bitwise complement operation upon the encrypted data, we are still able to obtain $\mathcal{E}(\bar{x})$ by

$$\mathcal{E}(\bar{x}) \equiv \mathcal{E}(2^t - 1) \cdot \mathcal{E}(x)^{-1} \equiv \mathcal{E}(2^t - 1 - x) \bmod N^2, \tag{4.10}$$

where $t = \lfloor \log_2 N \rfloor$ is the number of bits used to describe x. We will prove that

$\bar{x} = 2^t - 1 - x$ later. Hence, the encoding process is carried out by

$$\mathcal{E}(y) \equiv \begin{cases} \mathcal{E}(x) \bmod N^2, & \text{if } w = 0, \\ \mathcal{E}(2^t - 1) \cdot \mathcal{E}(x)^{-1} \bmod N^2, & \text{if } w = 1, \end{cases} \tag{4.11}$$

and the decoding process is performed by

$$(x, w) = \begin{cases} (y, 0), & \text{if } \Delta(y, \tilde{y}) < \Delta(\bar{y}, \tilde{y}), \\ (\bar{y}, 1), & \text{otherwise}, \end{cases} \tag{4.12}$$

where $\bar{y}$ is the bitwise compliment of y, $\tilde{y}$ is the approximation of y, and Δ is the measure of the Hamming distance.

Lemma. *Let x be a non-negative integer, $\bar{x}$ be the bitwise compliment of x, and $t = \lfloor \log_2 N \rfloor$ be the number of bits used to describe x. Then, $\bar{x} = 2^t - 1 - x$.*

Proof. *Let us prove that*

$$x + \bar{x} = 2^t - 1$$

such that

$$x \oplus \bar{x} = \underbrace{11\ldots1}_{t \text{ bits}}, \quad \forall x, \bar{x} \in \mathbb{Z}/2^t\mathbb{Z},$$

where $\oplus$ denotes the bitwise XOR operation. This is proved by the fact that integer addition is equivalent to bitwise XOR and $2^t - 1 = 2^0 + 2^1 + \cdots + 2^{t-1} = \underbrace{11\ldots1}_{t \text{ bits}}$.

4.4 Online and Offline Content-Adaptive Predictors

Let us demonstrate how to implement the estimation mechanisms in a practical sense. Suppose that the host data is a digital image. Recall that we have to estimate the symbols in order to activate the decoding process. A symbol is defined as an integer converted from some randomly drawn bits from the l-th bit-plane. Therefore, the objective is to predict those selected bits. A possible strategy is to predict the pixel itself. This strategy is advantageous in terms of the implementation cost since there are a lot of image reconstruction tools available already, such as the total

variation denoising algorithm [217]. Nonetheless, most content-adaptive predictors are 'online' processing algorithms and thus it is computationally demanding for the decoder to perform those algorithms. In response to this, we further devise an 'offline' processing mechanism that efficiently infers the l-th bit of the given pixel by a pre-learned lookup table. The approach is derived from the Bayesian inference. We remark that these online and offline algorithms can be viewed as a classic example of space-time tradeoff. We begin with the implementation of the total variation denoising algorithm and then continue with the formulation of the Bayesian inference.

4.4.1 Total Variation Denoising

Let us consider a marked image as a noisy signal. The result of noise removal can be perceived as our expectation of the original signal. Recall that when the l-th bit-plane is taken as the watermarking channel, the l-th bit of each modifiable pixel is either flipped or kept intact. Therefore, by comparing the marked image with its denoised counterpart, we are able to infer whether an observed pixel has been modified or not. For instance, suppose that a pixel u is used to carry a watermark bit. At the receiving end, we want to know whether $u = u_0$ or $u = u_1$, where u_0 and u_1 denote the pixel values by setting the l-th bit of u to 0 and 1, respectively. By applying a denoising technique, we obtain an approximation of u denoted by $\tilde{u}$. We contend that the value of u is the one that minimises the Euclidean distance between u and $\tilde{u}$. Let $\mathbf{u}$ be a sequence of pixels to carry a watermark bit and $\tilde{\mathbf{u}}$ be an approximation of it. In practice, we tend to use an appropriately long sequence of pixels to carry one single bit of information in order to minimise the error rate, although, on the other hand, compromising the embedding rate. To decide whether $\mathbf{u} = \mathbf{u}_0$ or $\mathbf{u} = \mathbf{u}_1$, we calculate

$$\begin{aligned} \Delta_0 &= \|\mathbf{u}_0 - \tilde{\mathbf{u}}\|_1 = \sum_i |u_{0,i} - \tilde{u}_i|, \\ \Delta_1 &= \|\mathbf{u}_1 - \tilde{\mathbf{u}}\|_1 = \sum_i |u_{1,i} - \tilde{u}_i|, \end{aligned} \tag{4.13}$$

and choose the vector that produces a smaller L_1 norm.

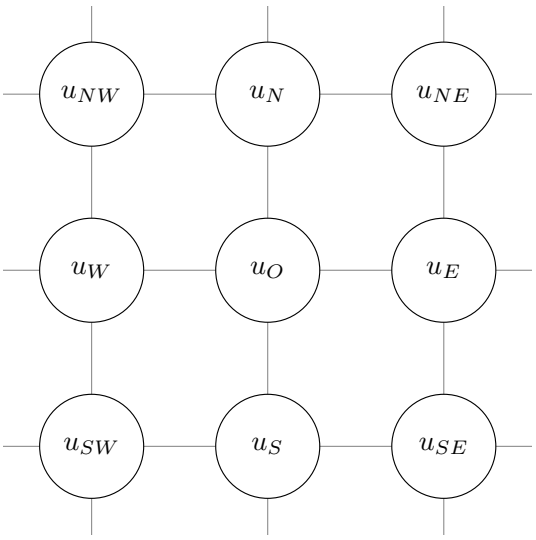


Figure 4.4: A given pixel u_O and its correlated neighbouring pixels u_N, u_S, u_W, u_E, u_{NW}, u_{NE}, u_{SW} and u_{SE}.

Total variation denoising is based on the principle that noisy signals have high total variation. According to this principle, the denoising problem is modelled as minimising the total variation of the reconstructed signal subject to it being close to the original observed signal. Let the intensity function $u^0(x, y)$ denote the pixel intensity of an observed noisy image and $u(x, y)$ the pixel values of the desired clean image for $x, y \in \Omega$. Hence,

$$u^0(x, y) = u(x, y) + n(x, y), \tag{4.14}$$

where $n(x, y)$ denotes the additive noise with zero mean and standard deviation σ. The objective is to remove n and reconstruct u from u_0. The variational model is to minimise

$$\iint_\Omega |\nabla u| \, dx \, dy, \tag{4.15}$$

subject to

$$\frac{1}{2} \iint_\Omega (u - u^0)^2 \, dx \, dy = \sigma^2. \tag{4.16}$$

By introducing a Lagrange multiplier λ, this problem can be converted into an

unconstrained optimisation problem, which is to minimise the functional

$$J[u] = \iint_{\Omega} |\nabla u| \, dx \, dy + \frac{\lambda}{2} \iint_{\Omega} (u - u^0)^2 \, dx \, dy. \tag{4.17}$$

Let us express the above equation in a compact form by

$$J = \iint_{\Omega} \mathcal{L}(\Omega, u, \nabla u) \, d\Omega. \tag{4.18}$$

To minimise J, we find the partial derivatives of $\mathcal{L}$, which are given by

$$\frac{\partial \mathcal{L}}{\partial u} = \frac{\partial}{\partial u} \frac{\lambda}{2} (u - u^0)^2 = \lambda (u - u^0), \tag{4.19}$$

and

$$\frac{\partial \mathcal{L}}{\partial \nabla u} = \frac{\partial \mathcal{L}}{\partial u_x} + \frac{\partial \mathcal{L}}{\partial u_y} = \frac{\nabla u}{|\nabla u|}, \tag{4.20}$$

where

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial u_x} &= \frac{\partial}{\partial u_x} |\nabla u| = \frac{\partial}{\partial u_x} \sqrt{u_x^2 + u_y^2} \\ &= \frac{\partial}{\partial u_x} \psi^{\frac{1}{2}} = \psi^{-\frac{1}{2}} u_x = \frac{u_x}{|\nabla u|}, \end{aligned} \tag{4.21}$$

and similarly

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial u_y} &= \frac{\partial}{\partial u_y} |\nabla u| = \frac{\partial}{\partial u_y} \sqrt{u_x^2 + u_y^2} \\ &= \frac{\partial}{\partial u_y} \psi^{\frac{1}{2}} = \psi^{-\frac{1}{2}} u_y = \frac{u_y}{|\nabla u|}. \end{aligned} \tag{4.22}$$

Note that we let $\psi = u_x^2 + u_y^2$. By substituting these into the Euler-Lagrange equation

$$\frac{\partial \mathcal{L}}{\partial u} - \frac{d}{d\Omega} \left(\frac{\partial \mathcal{L}}{\partial \nabla u} \right) = 0, \tag{4.23}$$

we obtain

$$\lambda (u - u^0) - \nabla \cdot \frac{\nabla u}{|\nabla u|} = 0. \tag{4.24}$$

Therefore, the steepest descent equation for J is given by

$$\frac{\partial u}{\partial t} = \lambda(u - u^0) - \nabla \cdot \frac{\nabla u}{|\nabla u|}, \tag{4.25}$$

and thus at the time t we update

$$\begin{aligned} u^{t+1} &= u^t + \partial u^t \\ &= u^t + \partial t(\lambda^t(u^t - u^0) - \nabla \cdot \frac{\nabla u^t}{|\nabla u^t|}), \end{aligned} \tag{4.26}$$

where

$$\nabla \cdot \frac{\nabla u}{|\nabla u|} = \frac{\partial}{\partial x}\frac{u_x}{\sqrt{(u_x)^2 + u_y^2}} + \frac{\partial}{\partial y}\frac{u_y}{\sqrt{u_x^2 + u_y^2}}. \tag{4.27}$$

At a given pixel u_O, let u_N, u_S, u_W and u_E denote its four neighbouring pixels at north, south, west and east directions respectively, as illustrated in Fig. 4.4. To simplify the notations, we denote the four neighbouring pixels altogether by u_P, where $P \in \Lambda = \{N, S, W, E\}$. In the following, we present the numerical method for updating u_O via gradient descent. The algorithm is performed iteratively until it converges to a stable state or the default maximum number of iterations is reached. For each iteration, the numerical approximation of u_O is computed by

$$u_O^{t+1} = u_O^t + \Delta t \cdot (\lambda^t(u_O^t - u_O^0) - \frac{1}{h}\frac{\nabla u_O^t}{|\nabla u_O^t|}), \tag{4.28}$$

where Δt and h are set to 1 in our implementation. For convenience, we set λ^t to a fixed small number instead of updating it dynamically ($\lambda^t = 0.001$). We discretise

$$\frac{\nabla u_O^t}{|\nabla u_O^t|} \simeq \sum_{P \in \Lambda} \frac{u_P - u_O^t}{\sqrt{(u_P - u_O^t)^2 + \xi_P^2 + \epsilon}}, \tag{4.29}$$

where

$$\begin{aligned}
\xi_N &= \frac{(u_W + u_{NW}) - (u_E + u_{NE})}{4}, \\
\xi_S &= \frac{(u_W + u_{SW}) - (u_E + u_{SE})}{4}, \\
\xi_W &= \frac{(u_N + u_{NW}) - (u_S + u_{SW})}{4}, \\
\xi_E &= \frac{(u_N + u_{NE}) - (u_S + u_{SE})}{4},
\end{aligned} \tag{4.30}$$

and ϵ is a very small number to avoid a zero divisor in practice.

4.4.2 Bayesian Inference

A statistical approach to estimate the l-th bit of a pixel given a certain context is to collect a large number of samples in the same context and see which case, either that the bit is 0 or 1, is observed more frequently. This method will, however, encounter the so-called curse of dimensionality when the context is in a high dimensional space. That is, the amount of data required to support the sampling grows exponentially with the dimensionality of context. As aforementioned, a pixel is correlated to its eight immediate neighbours, which implies that the l-th bit of a given pixel is correlated to other 7 bits of that pixel and 8 bits of each neighbouring pixel. In total, there are 71 correlated bits, which represent 2^{71} different contexts. In this case, we require an enormous amount of data so that there are sufficient samples for each context.

In order to reduce the dimensionality of context, we model that the l-th bit of a given pixel is correlated to its immediate neighbouring bits on the current and the two adjacent bit-planes, as illustrated in Fig. 4.5. For the least significant bit-plane, we consider only the bits on the current and the second least significant bit-planes. For the most significant bit-plane, we consider only the bits on the current and the second most significant bit-planes. There are 8 correlated bits on the current bit plane, 9 on each of the two adjacent bit-planes. For a bit on the second to the seventh bit-plane, it has 26 correlated bits and thus 2^{26} possible contexts. For a bit on the first or last bit-plane, there are 17 correlated bits and accordingly 2^{17} contexts. The number of contexts is significantly reduced and yet the estimation is

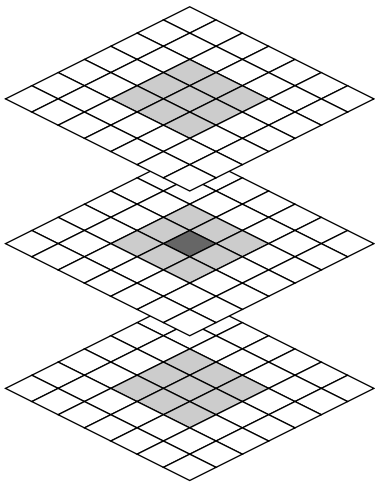


Figure 4.5: A given bit (coloured in dark grey) and its correlated neighbouring bits (coloured in light grey).

still remarkably accurate as demonstrated experimentally later. Let us denote the bit to be estimated by b and the collection of correlated bits, or the context, by Θ. According to the Bayes' theorem, we compute

$$\begin{aligned} P(b=0|\Theta) &\propto P(\Theta|b=0)P(b=0), \\ P(b=1|\Theta) &\propto P(\Theta|b=1)P(b=1), \end{aligned} \tag{4.31}$$

where $P(b)$ is the prior probability of the hypothesis of b, $P(\Theta|b)$ is the likelihood of observing the evidence Θ given the hypothesis of b, and $P(b|\Theta)$ is the posterior probability of the hypothesis of b given the observed evidence Θ. Therefore, the inference about the value of b is made by

$$\tilde{b} = \arg\max_{b\in\{0,1\}} P(b|\Theta). \tag{4.32}$$

As a result, we learn the Bayesian probability table from a large number of image samples and predict the bits by the lookup table, as illustrated in Table 4.1.

4.5 Experiments

In the following experiments, we test the schemes on 8-bit greyscale images of size 512×512, as shown in Fig. 4.6. We use the RSA and the Paillier cryptosystems as

Table 4.1: Bayesian probability table, in which each column represents a possible context and each row represents a possible bit value.

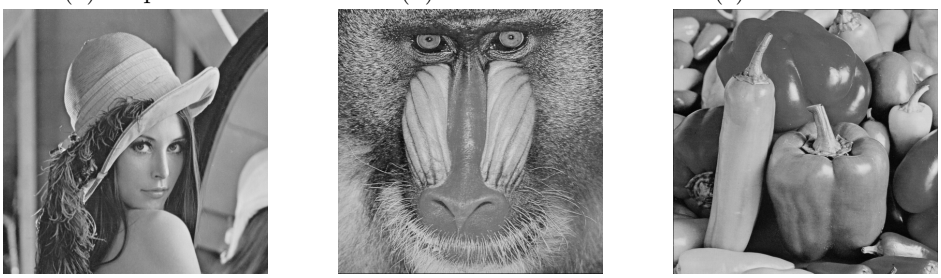
	Θ_1	Θ_2	$\cdots$
$b = 0$	$P(b = 0\|\Theta_1)$	$P(b = 0\|\Theta_2)$	$\cdots$
$b = 1$	$P(b = 1\|\Theta_1)$	$P(b = 1\|\Theta_2)$	$\cdots$

examples of multiplicative and additive homomorphic cryptosystems, respectively. Two prime numbers for the cryptosystems are $p = 7907$ and $q = 7919$ so that the bit-length of message is $\lfloor \log_2 pq \rfloor = 25$. Accordingly, for each bit-plane of an image, we convert every segment of 25 bits into a decimal number in order to fit the encryption function, where the marginal bits are negligible. The number of iterations for the total variation algorithm is set to 2000 since empirically it produces stable results. The Bayesian probability table is learned from a thousand image samples in the BOSSBase [218]. We begin with the analysis of three-way trade-off between capacity, fidelity, and reversibility. Then, we make comparisons between the proposed schemes and the state-of-the-art. The capacity is represented by the number of bits embedded in an image, whereas the fidelity and reversibility are measured by the peak signal-to-noise ratio (PSNR). Let $u_{i,j}$ and $\hat{u}_{i,j}$ denote a pixel and its noisy counterpart respectively, where i and j specify the pixel coordinates. The fidelity is quantified by the PSNR between the original and the marked images, and in a similar manner, the reversibility is evaluated by the PSNR between the original and the recovered images. For further analysis, we also measure the reversibility in terms of the number of bit errors in the following discussions.

Capacity-Fidelity Trade-Off

The trade-off between capacity and fidelity is visualised in Fig. 4.7a, which shows that the RSA-based scheme achieves higher fidelity than the Paillier-based scheme under the same capacity (2000 bits). The reason is straightforward: when the noise occurs, the former only introduces a random distortion while the latter inflicts a maximal distortion. Note that the fidelity is not content-dependent according to our scheme design. The impact of the selection of different bit-planes as the watermarking channel is shown in Table 4.2. It is intuitive that the fidelity of marked images

(a) Airplane (b) Barbara (c) Cameraman



(d) Lena (e) Mandrill (f) Peppers

Figure 4.6: Greyscale test images of size 512×512 with 256 tonal options.

decreases drastically when a more significant bit-plane is engaged to carry watermark information.

Capacity-Reversibility Trade-Off

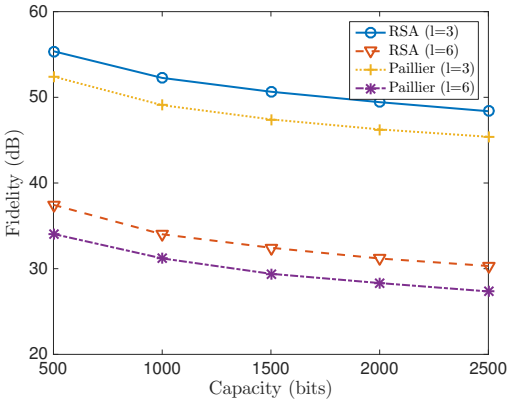
The trade-off between capacity and reversibility is illustrated in Fig. 4.7b. To assess the capacity-reversibility trade-off, we control the capacity by selecting different bit-planes for watermarking in such a way that the fidelity remains at 48 dB, and then observe the change of reversibility. Note that here we express the reversibility in terms of the number of bit errors in order to amplify the trend. If the reversibility is quantified by the PSNR, the proportionality may be indeterminate since a higher decoding error rate does not necessarily mean a lower visual quality. For instance, several errors occur on an insignificant bit-plane may still result in a higher visual quality than a single error on a relatively significant bit-plane. It is observed that the reversibility decreases as the capacity increases. It is due to the fact that we embed less amount of information into the more significant bit-plane and more amount into the less significant bit-plane in order to keep the fidelity at a fixed level.

Fidelity-Reversibility Trade-Off

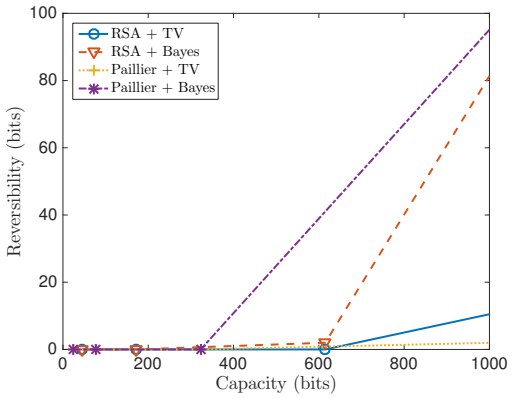
The trade-off between capacity and reversibility is illustrated in Fig. 4.7c. In a similar way, we evaluate the fidelity-reversibility trade-off by setting the capacity to 2000 bits, controlling the fidelity with the selection of encodable bit-plane, and observing the change of reversibility. It is evident that the Paillier-based scheme is advantageous in terms of the reversibility compared to the RSA-based scheme. This is because that the Paillier-based scheme inflicts a stronger noise to the host media, which is more distinguishable and removable than a faint noise imposed by the RSA-based scheme. On top of that, it is shown that the predictor based on total variation achieves higher reversibility than the predictor based on the Bayesian inference. Moreover, it can be observed that the reversibility is inversely proportional to the fidelity.

More comprehensive experimental results are demonstrated in Table 4.3, where a variety of images are tested with every possible combination of encoding and decoding mechanisms. It is witnessed that the error-free performance is achievable in practice. It can be observed that the error rate generally approaches zero when embedding payloads into a comparatively more significant bit-plane. From the previous analyses, we conclude that

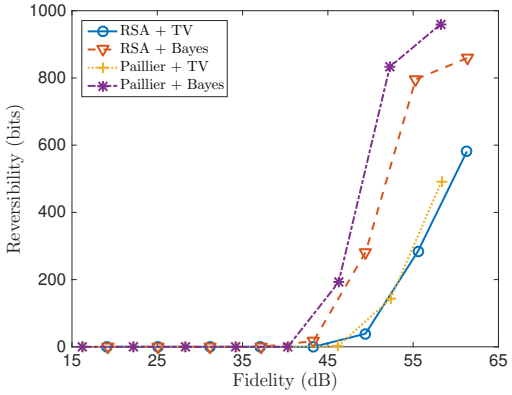
- Capacity is inversely proportional to fidelity.
- Capacity is inversely proportional to reversibility.
- Fidelity is inversely proportional to reversibility.
- The RSA-based scheme preserves higher fidelity than the Paillier-based scheme.
- The Paillier-based scheme achieves higher reversibility than the RSA-based scheme.
- The predictor based on total variation is more content-adaptive and results in higher reversibility than that based on Bayesian inference.
- The predictor based on Bayesian inference consumes less computational power in decoding than that based on total variation.



(a) Capacity-fidelity tradeoff.



(b) Capacity-reversibility tradeoff.



(c) Fidelity-reversibility tradeoff.

Figure 4.7: A three way trade-off between capacity, fidelity, and reversibility measured on the image Lena.

In the following, we compare the scheme performance with the state-of-the-art. We begin by classifying the existing reversible watermarking schemes compatible with public-key cryptosystems [200–202, 205, 206], as presented in Table 4.4. As far as we are aware, our work is one of the pioneering research on the schemes compatible with public-key homomorphic cryptosystems under the condition that neither additional bit padding nor specific pre-processing is undertaken. Although schemes addressing this strictly defined problem are hardly found, we remark that this design principle was followed in the earlier literature of schemes based on traditional symmetric-key cryptosystems. In order to make paralleled and meaningful comparisons, the proposed schemes are compared with those under the same, or at least similar, research constraint [180, 182, 219]. We test the fidelity and reversibility between

Table 4.2: Fidelity evaluation of RSA-based and Paillier-based schemes by using different bit-planes as the watermarking channel. It shows the PSNR (in dB) of the marked image Lena.

	RSA	Paillier
$l = 1$	**67.59**	**64.25**
$l = 2$	61.19	58.41
$l = 3$	55.46	52.43
$l = 4$	49.46	46.28
$l = 5$	43.27	39.86
$l = 6$	37.58	34.22
$l = 7$	31.18	28.09
$l = 8$	25.31	21.96

(a) Capacity: 500 bits

	RSA	Paillier
$l = 1$	**64.32**	**61.28**
$l = 2$	58.32	55.10
$l = 3$	52.17	49.23
$l = 4$	46.29	43.39
$l = 5$	40.32	37.20
$l = 6$	34.11	31.25
$l = 7$	28.41	25.28
$l = 8$	22.37	19.12

(b) Capacity: 1000 bits

	RSA	Paillier
$l = 1$	**62.53**	**59.70**
$l = 2$	56.55	53.42
$l = 3$	50.61	47.62
$l = 4$	44.40	41.47
$l = 5$	38.48	35.33
$l = 6$	32.56	29.49
$l = 7$	26.57	23.53
$l = 8$	20.29	17.53

(c) Capacity: 1500 bits

	RSA	Paillier
$l = 1$	**61.29**	**58.23**
$l = 2$	55.57	52.26
$l = 3$	49.41	46.26
$l = 4$	43.32	40.31
$l = 5$	37.08	34.18
$l = 6$	31.18	28.32
$l = 7$	25.09	22.18
$l = 8$	19.05	16.30

(d) Capacity: 2000 bits

different schemes by embedding randomly generated 2000 bits into various host images. It is observed from Table 4.5 that the visual qualities of various host images are similar for any listed encoding algorithm and thus it is evident that the fidelity is not content-dependent. Yet, different schemes and configurations result in different fidelity. It is shown that the RSA-based and Paillier-based schemes preserves comparatively high fidelity when embedding payloads into the third bit-plane. From Table 4.6, we can further observe that the proposed schemes achieve relatively high reversibility, especially when using the total variation technique for decoding. When the sixth bit-plane is used as the embedding channel, the proposed schemes show the error-free performance on most of the test images. Even when embedding payloads into the third bit-plane, the reversibility is higher than the average in most cases. The experimental results show that the proposed schemes achieve a remarkable

balance between fidelity and reversibility under the given capacity constraint. On top of this, the proposed schemes share the advantages of the modern public-key cryptosystems.

Table 4.3: Reversibility evaluation of different combinations of encoding and decoding components. It shows the PSNR (in dB) and the number of bit errors of the recovered images in which 2000 random bits are embedded.

RSA+TV

	Airplane		Barbara		Cameraman		Lena		Mandrill		Peppers	
	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM
$l = 1$	65.38	403	63.10	667	71.61	102	63.75	581	62.36	795	62.82	715
$l = 2$	64.38	135	59.20	417	81.24	3	60.90	285	56.53	760	58.43	500
$l = 3$	69.10	13	59.18	112	∞	0	64.06	39	52.44	489	56.41	205
$l = 4$	∞	0	60.90	20	∞	0	∞	0	50.22	216	56.41	10
$l = 5$	∞	0	63.60	3	∞	0	∞	0	54.21	23	∞	0
$l = 6$	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0
$l = 7$	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0
$l = 8$	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0

RSA+Bayes

	Airplane		Barbara		Cameraman		Lena		Mandrill		Peppers	
	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM
$l = 1$	62.85	706	61.82	886	67.51	249	62.02	859	61.90	882	61.68	915
$l = 2$	59.24	404	56.09	830	74.06	16	56.32	796	55.68	917	55.83	882
$l = 3$	62.04	60	52.93	434	∞	0	54.89	282	49.93	876	51.23	645
$l = 4$	∞	0	55.76	61	∞	0	61.60	17	45.54	599	54.33	87
$l = 5$	∞	0	61.60	4	∞	0	∞	0	45.42	160	∞	0
$l = 6$	∞	0	∞	0	∞	0	∞	0	49.88	15	∞	0
$l = 7$	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0
$l = 8$	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0

Paillier + TV

	Airplane		Barbara		Cameraman		Lena		Mandrill		Peppers	
	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM
$l = 1$	64.30	253	60.78	569	74.53	24	61.41	492	59.37	787	60.09	667
$l = 2$	68.00	27	58.62	234	∞	0	60.76	143	54.06	669	56.46	385
$l = 3$	∞	0	63.28	20	∞	0	73.28	2	50.87	348	58.16	65
$l = 4$	∞	0	∞	0	∞	0	∞	0	50.68	91	∞	0
$l = 5$	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0
$l = 6$	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0
$l = 7$	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0
$l = 8$	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0

Paillier + Bayes

	Airplane		Barbara		Cameraman		Lena		Mandrill		Peppers	
	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM
$l = 1$	59.60	746	58.42	979	65.64	186	58.51	960	58.42	981	58.31	1004
$l = 2$	57.11	331	52.69	916	∞	0	53.10	833	52.32	999	52.41	977
$l = 3$	65.15	13	51.45	305	∞	0	53.46	192	46.60	932	48.32	626
$l = 4$	∞	0	59.48	12	∞	0	∞	0	42.61	584	55.36	31
$l = 5$	∞	0	∞	0	∞	0	∞	0	45.33	78	∞	0
$l = 6$	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0
$l = 7$	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0
$l = 8$	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0

Table 4.4: Classification of reversible watermarking schemes compatible with public-key cryptosystems.

	Proposed	Wu *et al.* [200]	Wu *et al.* [201]	Li and Li [202]	Zhang *et al.* [205]	Xiang and Luo [206]
Bit-padding	No	Yes	Yes	Yes	No	No
Pre-processing	No	No	No	No	Yes	Yes

Table 4.5: Fidelity comparison with the state-of-the-art. It shows the PSNR (in dB) of the marked images in which 2000 random bits are embedded.

	Airplane	Barbara	Cameraman	Lena	Mandrill	Peppers
RSA ($l = 3$)	**49.27**	**49.32**	**49.47**	**49.43**	**49.13**	**49.19**
RSA ($l = 6$)	31.13	31.08	31.31	31.18	31.32	31.19
Paillier ($l = 3$)	46.18	46.36	46.10	46.25	46.26	46.23
Paillier ($l = 6$)	28.14	28.37	28.17	28.32	28.26	28.16
Zhang [180]	41.11	41.04	40.98	41.06	41.04	41.0
Wu and Sun ($l = 3$) [219]	46.30	46.16	46.33	46.29	46.16	46.27
Wu and Sun ($l = 6$) [219]	28.25	28.39	28.25	28.43	28.38	28.22
Liao and Shu [182]	41.12	41.04	41.05	41.04	41.03	41.04

Table 4.6: Reversibility comparison with the state-of-the-art. It shows the PSNR (in dB) and the number of bit errors of the recovered images in which 2000 random bits are embedded.

	Airplane		Barbara		Cameraman		Lena		Mandrill		Peppers	
	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM	PSNR	NUM
RSA + TV ($l = 3$)	69.10	13	59.18	112	∞	0	64.06	39	52.44	489	56.41	205
RSA + TV ($l = 6$)	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0
RSA + Bayes ($l = 3$)	62.04	60	52.93	434	∞	0	54.89	282	49.93	876	51.23	645
RSA + Bayes ($l = 6$)	∞	0	∞	0	∞	0	∞	0	49.88	15	∞	0
Paillier + TV ($l = 3$)	∞	0	63.28	20	∞	0	73.28	2	50.87	348	58.16	65
Paillier + TV ($l = 6$)	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0
Paillier + Bayes ($l = 3$)	65.15	13	51.45	305	∞	0	53.46	192	46.60	932	48.32	626
Paillier + Bayes ($l = 6$)	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0
Zhang [180]	55.09	71	48.93	169	54.38	85	57.10	28	46.16	309	54.57	47
Wu and Sun ($l = 3$) [219]	∞	0	58.58	59	∞	0	69.30	5	50.03	423	57.66	73
Wu and Sun ($l = 6$) [219]	∞	0	∞	0	∞	0	∞	0	∞	0	∞	0
Liao and Shu [182]	60.73	13	57.57	23	67.56	3	62.46	7	52.54	70	58.13	20

4.6 Summary

This chapter considers the problem of entrusting the watermarking operation to a cloud service provider without undermining data privacy. Constructing reversible watermarking schemes compatible with public-key cryptosystems is of non-trivial challenge since there is virtually no data redundancy to be exploited in the encrypted domain. The recent development of various schemes required the host media either to be enciphered in a redundant fashion or to be pre-processed prior to encryption. To address these limits, we propose a novel paradigm and derive different schemes compatible with different types of public-key homomorphic cryptosystems. The host image is encrypted in such a fashion that a sufficiently long sequence of bits, instead of a pixel, is regarded as input to the encryptor. It significantly reduces the size of the encrypted data and improves the space efficiency. In addition to this, we suggest to encrypt different bit-planes separately in order to control the fidelity factor. Any specific data pre-processing is not required in order to promote the practicality and universality. For the RSA-like cryptosystems that permit multiplicative homomorphic computation, the proposed scheme is flexible in terms of embedding capacity. As for the Paillier-like cryptosystems that allow additive operation as well as partial multiplicative operation, we propose a scheme which is optimal with respect to the reversibility. Furthermore, we develop content-adaptive predictors based on variational method and statistical inference for assisting watermark decoding. Both online and offline prediction algorithms are provided in order to suit different operational requirements. We remark that the decoding errors are inevitable from a theoretical point of view and yet in practice the error rate approaches zero at a low expense of the fidelity and capacity. Experimental results show that the proposed schemes achieve remarkable balance between fidelity and reversibility under the given capacity constraint. From our perspective, it is of great significance to develop privacy-aware watermarking methodology suitable for a variety of modern cryptosystems. On top of this, more practical applications, such as privacy protection in the Internet of Things, deserve further investigations.

Chapter 5

Privacy-Preserving Secret Sharing

Internet of Things (IoT) is an emerging technology that utilises cloud connected devices to collect data for analysis. Healthcare industry is one of the most promising fields that have adopted IoT solutions since its early stage. The development of wearable technology, wireless body area network and cloud computing has established a new way for medical practitioners to acquire health data from patients. It greatly benefits health monitoring, epidemiological studies, and pharmaceutical research [220–222].

A common IoT-aware architecture for healthcare applications is illustrated in Fig. 5.1, which consists of a gateway device, a cloud server and several sensor nodes. Each sensor node can be viewed as a wearable equipment used for monitoring the health status of an individual, such as heart rate, blood pressure, brain wave, glucose level, *etc.* Under the given framework, the sensor nodes send the medical data to a local gateway device via wireless communication such as Wi-Fi or Bluetooth, whereas the gateway device aggregates the data and store it in the cloud server for further analysis.

However, there are risks of information leakage during data transmission and storage. For example, an adversary may attempt to eavesdrop the wireless communication, attack the gateway device or even access to the cloud server. Therefore, it

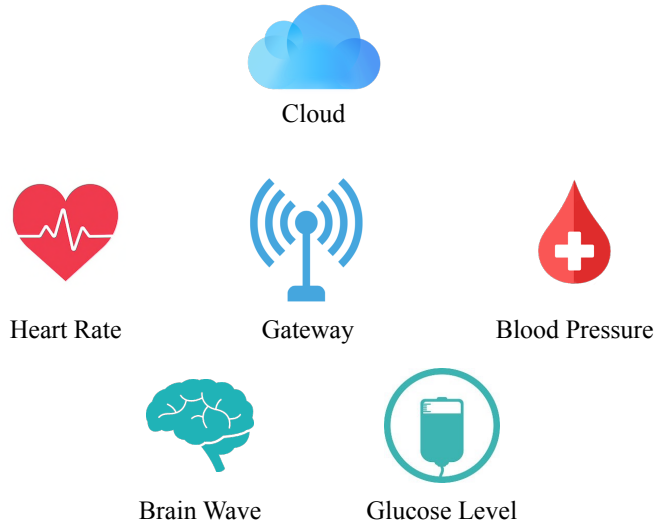


Figure 5.1: An IoT-based healthcare architecture. The health data acquired from the sensor nodes (*e.g.* heart rate, blood pressure, brain wave, and glucose level) is aggregated at the gateway and store in the cloud.

is advisable to encrypt the data at each sensor node immediately after it is produced and incorporate secret sharing schemes to realise access control. In more details, each sensor node transmits the encrypted data to the gateway device by which data is integrated and encoded into shares of information. Due to security concerns, these shares are stored in separate cloud servers and the data retrieval must conform with the access policy. To realise this system, we present a novel research upon secret sharing in the encrypted domain.

In this chapter, we study secret sharing mechanisms towards resolving privacy and security issues in IoT-based healthcare applications. Nevertheless, the applications of the proposed schemes are not limited to this particular IoT context. Instead, the proposed schemes are rather general and can be served as potential solutions for other similar problems. We show how multiple sources are possible to share their data amongst a group of participants without revealing their own data to one another as well as the dealer. Only an authorised subset of participants is able to reconstruct the data. We analyse the pros and cons of several possible solutions and develop practical schemes: a simple (2, 2)-threshold scheme, an extended (n, n)-threshold scheme, and a generalised (t, n)-threshold scheme. The developed schemes follow

Sharmir's construction in which a collusion of fewer than t participants has no better chance of guessing the secret than a non-participant who has no privileged information at all.

The remainder of this chapter is organised as follows. Section 5.1 gives a brief introduction of secret sharing. Section 5.2 presents a variety of privacy-preserving secret sharing schemes from the most limited construction to the most general construction. The chapter is concluded in Section 5.3 with directions for future research.

5.1 Secret Sharing

In this section, we discuss the basic notions and terminology of secret sharing and also briefly introduce its history and applications. The focus of this section is Shamir's secret sharing scheme, which forms the cornerstone of our scheme constructions.

Secret sharing is a study in cryptography and plays an essential role in visual cryptography. It can also be perceived as a special form of information hiding. In addition to this, It is widely used in applications of *secure multi-party computation* such as sealed-bid auction [223–226] and secret ballot [227–232].

Secret sharing was originated independently by Shamir [233] and Blakley [234] in 1979. In general, a secret sharing scheme describes how a dealer can split a secret message into pieces of information and distribute them amongst a group of participants. Each piece of information is referred to as a *share* (as Sharmir's terminology) or a *shadow* (as Blakley's terminology). The secret will be revealed only in the presence of a sufficient number of shares or when an authorised subgroup of participants work together. It is also rigorously defined that any individual share or unauthorised combination of shares reveals absolutely no information about the secret. Consider a naïve example of secret sharing in which a secret message, say, 'password' is split into the following shares: 'pa------', '--ss----', '----wo--', and '------rd'. It is manifestly insecure in the sense that every share leaks part of the secret.

The most typical scheme of secret sharing is a (t, n)-threshold scheme, which involves n participants and any group of t or more participants will be able to

reconstruct the secret. In Shamir's construction, the secret is set as the constant term of a polynomial of degree $t-1$ and the secret shares are any n different points on the polynomial. Hence, any t points will be sufficient to define the polynomial and thus its constant term, namely the secret digit. In Blakley's construction, consider the secret as a point in a three-dimensional space and the shadows as hyperplanes whose common intersection is the secret point. As a consequence, any three planes will suffice to identify the point. As each successive shadow is exposed, however, the range of possible values of the secret narrows. Shamir's secret sharing scheme is algebraic in nature in contrast to Blakley's geometric solution.

Shamir's secret sharing scheme forms the cornerstones for the later constructions of the proposed scheme. Hence, let us discuss Shamir's secret sharing in more detail. Suppose that a dealer wants to share a secret to n participants in such a way that only more than t participants pool their shares together will the secret be reconstructed. Let the secret be denoted by s and we generate $t-1$ random numbers denoted by r_1, r_2, ..., and r_{t-1}. Then, we form a polynomial

$$f(x) \equiv s + r_1 x + r_2 x^2 + \cdots + r_{t-1} x^{t-1} \pmod{P}, \tag{5.1}$$

where P is a randomly chosen prime number. Let us draw any n points from the polynomial, for example, $(1, f(1))$, $(2, f(2))$, ..., and $(n, f(n))$, and distribute them to n participants respectively as shares. It is observed that there are t unknown variables in the polynomial and thus with t different points one is able to solve for the variables including the secret (*i.e.* the constant term). In other words, the reconstruction process is to simply use Lagrange interpolation to solve a set of t simultaneous equations.

Since the introduction of secret sharing, numerous extended problems have appeared. The study towards a general access structure was considered by Ito, Saito, and Nishizeki [235] and had become a principal study since then [236–239]. To manage various malicious behaviour by dishonest parties, the notion of verifiable secret sharing was introduced by Chor *et al.* [240] and had been studied extensively thereafter [241–246]. Another closely related branch is visual cryptography originated

by Naor and Shamir [247] for the secrecy of visual information, including greyscale, colour, and halftone images [248–250].

5.2 Privacy-Preserving Secret Sharing

In this section, we present the proposed privacy-preserving secret sharing schemes that address the problem of sharing the secrets generated from multiple sources amongst a group of n participants through the computations in the encrypted domain. We analyse the pros and cons of several possible solutions and develop practical schemes: a simple (2, 2)-threshold scheme, an extended (n, n)-threshold scheme, and a generalised (t, n)-threshold scheme. The developed schemes follow Sharmir's construction in which a collusion of fewer than t participants has no better chance of guessing the secret than a non-participant who has no privileged information at all. In the remainder of this chapter, we assume all the homomorphic properties applied are those of the Paillier cryptosystem unless otherwise specified [164]. Nevertheless, the applicable homomorphic cryptosystems are included but by no means limited to this particular cryptosystem.

5.2.1 Naïve Solutions

In the following, let us consider two naïve solutions to our research problem and analyse their pros and cons. Among a variety of privacy protection mechanisms, encryption has a high level of reliability and universality. Naturally, the secrets are encrypted once they have been produced from the sources. The problem is therefore reduced to the sharing of encrypted data. Consider a key server who has a pair of public and private keys. The public key is used for encryption, whereas the private key is employed for decryption. The first solution is to create shares of the private key by arranging the key as the constant term in Eq. (5.1). The encrypted files, instead of being encoded as shares, are stored in a database. At the time when the number of collaborative participants are as many as required, the private key will be reconstructed and then the files in the database can be deciphered. On the one hand, this solution is simple and the computational load of the sharing procedure is

light. On the other hand, however, to access the secret files, one must perform one reconstruction algorithm for the key plus one decryption algorithm for the files. In addition to this, this scheme requires different pairs of public and private keys for different sets of secret files (*e.g.* different patients' health records); otherwise, once the participants reconstruct the private key, they will be able to decipher all the files stored in the database. Furthermore, storing all the important files in a central database may be vulnerable to a number of cyber attacks. Thus, it is reasonable to share the files to authorised participants to reduce the risk of cyber threats.

The second solution is that suppose there are t encrypted secrets denoted by $\mathcal{E}(s_0)$, $\mathcal{E}(s_1)$, ..., and $\mathcal{E}(s_{t-1})$. We form a polynomial by arranging t encrypted secrets as t coefficients in Eq. (5.1). More generally, we can assume that there are k encrypted secrets, where $k \leq t$, and choose $t - k$ random numbers as the rest of the coefficients to complete the polynomial. Either way, we can draw n points as the shares for individual participants. In the presence of t shares or more, the encrypted data will be reconstructed. With the decryption key, the data will eventually be revealed. In practice, this scheme has a non-trivial issue of key distribution amongst the participants. It may be addressed by one of the following approaches. First, use a secure channel to transmit the key to individual participants. Second, let the pair of encryption and decryption keys be generated by a key agreement protocol (*e.g.* Diffie–Hellman key exchange protocol [155]) amongst the group of participants, instead of being generated by the key server. Third, encrypt the key with each participant's public key and send it to the corresponding one as an instance of asymmetrical cryptography (*e.g.* elliptic curve cryptosystems [156]). Aside from the issue of key distribution, this scheme still requires extra efforts of participants, namely, one reconstruction step for the encrypted data plus one decryption step for the original data. It may be troublesome in particular situations. For instance, when there is a surgical emergency, the time delay for accessing health records becomes problematic. Hence, we conclude that these naïve solutions, though feasible, are deficient in several aspects, which motivate us towards finer constructions.

5.2.2 (2, 2)-threshold scheme

To begin with, we introduce a (2, 2)-threshold multi-secret sharing scheme to split a batch of two secrets into two shares via a semi-honest (or honest-but-curious) cloud service provider. Only in the presence of two shares, the batch of two secrets can be restored. Let us describe how this scheme can solve the problem of privacy-preserving secret sharing. Let s_1 and s_2 be two secrets generated from two separate sources, respectively. To preserve the privacy of secrets, s_1 and s_2 are encrypted immediately after being produced. The encrypted secrets $\mathcal{E}(s_2)$ and $\mathcal{E}(s_2)$ are uploaded to the dealer for sharing. Let x_1 and x_2 be any integers that satisfy

$$\begin{aligned} \gcd(x_1 + x_2, N) &= 1, \\ \gcd(x_1 - x_2, N) &= 1. \end{aligned} \tag{5.2}$$

Note that 'gcd' stands for greatest common divisor. It is not difficult to find proper x_1 and x_2 because N is the product of two large primes. Since

$$x_1{}^2 - x_2{}^2 \equiv (x_1 + x_2) \cdot (x_1 - x_2) \pmod{N}, \tag{5.3}$$

we derive

$$\gcd(x_1{}^2 - x_2{}^2, N) = 1. \tag{5.4}$$

This also implies

$$\gcd(x_1{}^2 - x_2{}^2, N) = 1. \tag{5.5}$$

Then, two shares are created as

$$\begin{aligned} \mathcal{E}(y_1) &\equiv \mathcal{E}(s_1)^{x_1} \cdot \mathcal{E}(s_2)^{x_2} \pmod{N^2}, \\ \mathcal{E}(y_2) &\equiv \mathcal{E}(s_1)^{x_2} \cdot \mathcal{E}(s_2)^{x_1} \pmod{N^2}. \end{aligned} \tag{5.6}$$

Following the homomorphic properties, we rewrite

$$\begin{aligned} \mathcal{E}(y_1) &\equiv \mathcal{E}(s_1 x_1 + s_2 x_2) \pmod{N^2}, \\ \mathcal{E}(y_2) &\equiv \mathcal{E}(s_1 x_2 + s_2 x_1) \pmod{N^2}. \end{aligned} \tag{5.7}$$

The dealer distributes x_1 and x_2 to two participants and sends $\mathcal{E}(y_1)$ and $\mathcal{E}(y_2)$ to the key server for decryption. The decrypted results are

$$\begin{aligned} y_1 &\equiv s_1 x_1 + s_2 x_2 \pmod{N}, \\ y_2 &\equiv s_1 x_2 + s_2 x_1 \pmod{N}. \end{aligned} \tag{5.8}$$

Then, y_1 and y_2 are also dispensed to the participants. When the participants pool their shares (x_1, y_1) and (x_2, y_2) together, they compute

$$x_1 y_1 - x_2 y_2 \equiv ({x_1}^2 - {x_2}^2) s_1 \pmod{N}. \tag{5.9}$$

Note that

$$\begin{aligned} x_1 y_1 &\equiv ({x_1}^2 s_1 + x_1 x_2 s_2) \pmod{N} \\ x_2 y_2 &\equiv ({x_2}^2 s_1 + x_1 x_2 s_2) \pmod{N}. \end{aligned} \tag{5.10}$$

Since $\gcd({x_1}^2 - {x_2}^2, n) = 1$, we know there exists one and only one modular multiplicative inverse such that

$$({x_1}^2 - {x_2}^2) \cdot ({x_1}^2 - {x_2}^2)^{-1} \equiv 1 \pmod{N}. \tag{5.11}$$

The value of $({x_1}^2 - {x_2}^2)^{-1}$ can be solved by the extended Euclidean algorithm. Eventually, the secret s_1 is unveiled by

$$s_1 \equiv (x_1 y_1 - x_2 y_2) \cdot ({x_1}^2 - {x_2}^2)^{-1} \pmod{N}. \tag{5.12}$$

In the same manner, the secret s_2 is decoded as

$$s_2 \equiv (x_2 y_1 - x_1 y_2) \cdot ({x_2}^2 - {x_1}^2)^{-1} \pmod{N}. \tag{5.13}$$

It is worth noting that even though y_1 and y_2 have been disclosed to the key server during the process, s_1 and s_2 are still kept secret since the key server has no knowledge about x_1 and x_2. The secret reconstruction process does not involve the decryption operation and thus is time-efficient.

5.2.3 (n, n)-threshold scheme

Let us extend the previous (2, 2)-threshold scheme to a (n, n)-threshold scheme. For conciseness, we omit modulus symbols in the following description where there is no ambiguity. Let n secrets generated from sources be denoted by s_1, s_2, ..., and s_n. After encryption, the encrypted results, written as $\mathcal{E}(s_1)$, $\mathcal{E}(s_2)$, ..., and $\mathcal{E}(s_n)$, are transmitted to the dealer for sharing. The dealer chooses n random numbers x_1, x_2, ..., and x_n such that a matrix

$$\mathbf{X} = \begin{pmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ x_2 & x_3 & x_4 & \cdots & x_1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_n & x_1 & x_2 & \cdots & x_{n-1} \end{pmatrix} \tag{5.14}$$

has a modular multiplicative inverse $\mathbf{X}^{-1}$ in $\mathbb{Z}/N\mathbb{Z}$. Alternatively, $\mathbf{X}$ must satisfy $\gcd(\det(\mathbf{X}), N) = 1$ and $\det(\mathbf{X}) \neq 0$. Note that 'det' stands for determinant. Let the dealer compute

$$\begin{aligned} \mathcal{E}(y_1) &= \mathcal{E}(s_1)^{x_1}\mathcal{E}(s_2)^{x_2}\cdots\mathcal{E}(s_n)^{x_n}, \\ \mathcal{E}(y_2) &= \mathcal{E}(s_1)^{x_2}\mathcal{E}(s_2)^{x_3}\cdots\mathcal{E}(s_n)^{x_1}, \\ &\vdots \\ \mathcal{E}(y_n) &= \mathcal{E}(s_1)^{x_n}\mathcal{E}(s_2)^{x_1}\cdots\mathcal{E}(s_n)^{x_{n-1}}. \end{aligned} \tag{5.15}$$

According to the homomorphic properties, we derive

$$\begin{aligned} \mathcal{E}(y_1) &= \mathcal{E}(s_1x_1 + s_2x_2 + \cdots + s_nx_n), \\ \mathcal{E}(y_2) &= \mathcal{E}(s_1x_2 + s_2x_3 + \cdots + s_nx_1), \\ &\vdots \\ \mathcal{E}(y_n) &= \mathcal{E}(s_1x_n + s_2x_1 + \cdots + s_nx_{n-1}). \end{aligned} \tag{5.16}$$

The sharing process can be fulfilled by cloud computing to relieve the dealer of computational burdens without revealing the private information about the secrets. The dealer dispenses x_1, x_2, ..., and x_n to n participants respectively and passes

$\mathcal{E}(y_1)$, $\mathcal{E}(y_2)$, ..., and $\mathcal{E}(y_n)$ to the key server for decryption. The decrypted results, written as

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \mathbf{X} \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{pmatrix}, \tag{5.17}$$

are allocated to individual participants as well. When all the participants pool their shares together, they retrieve the secrets by

$$\begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{pmatrix} = \mathbf{X}^{-1} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}. \tag{5.18}$$

Example. *Let us demonstrate that the previous (2, 2)-threshold scheme is actually a special case of the (n, n)-threshold scheme. In the case where there are two secrets s_1 and s_2 to be encoded, the dealer randomly chooses x_1 and x_2 to form a matrix*

$$\begin{pmatrix} x_1 & x_2 \\ x_2 & x_1 \end{pmatrix}.$$

Then, we compute

$$\mathcal{E}(y_1) = \mathcal{E}(s_1)^{x_1}\mathcal{E}(s_2)^{x_2} = \mathcal{E}(s_1x_1 + s_2x_2),$$
$$\mathcal{E}(y_2) = \mathcal{E}(s_1)^{x_1}\mathcal{E}(s_2)^{x_2} = \mathcal{E}(s_1x_2 + s_2x_1),$$

which are equivalent to the results in Eq. (5.6) *and Eq.* (5.7)*. By decryption, we obtain*

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} x_1 & x_2 \\ x_2 & x_1 \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \end{pmatrix},$$

which are equal to the results in Eq. (5.8)*. Eventually, we retrieve the secrets by*

$$\begin{pmatrix} s_1 \\ s_2 \end{pmatrix} = \begin{pmatrix} x_1 & x_2 \\ x_2 & x_1 \end{pmatrix}^{-1} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix},$$

which are identical to the results in Eq. (5.12) *and Eq.* (5.13)*.*

As an extension of the (2, 2)-threshold scheme, this scheme has the same security strength. It is theoretically secure in the sense that any subset of participants has absolutely no knowledge about the secrets unless all the shares are in presence. The secret reconstruction procedure does not involve decryption. Thus, it is time-efficient and can be established without the means of key distribution.

5.2.4 (t, n)-threshold scheme

In light of the previous (n,n)-threshold scheme, we further derive a generalised (t, n)-threshold scheme. Before we proceed further, let us discuss some possible (t, n)-threshold schemes and analyse their pros and cons. Let $\{s_i\}_{i=1}^{t}$ denote t secrets generated from separate sources and P be a large prime. With Shamir's algorithm, the dealer constructs a polynomial

$$f(x) \equiv \sum_{i=1}^{t} s_i x^{i-1} \pmod{P}, \tag{5.19}$$

and draws n points $(x_1, f(x_1))$, $(x_2, f(x_2))$, $\ldots$, $(x_n, f(x_n))$ as shares for n participants. In our defined scenario, the secrets are encrypted into $\{\mathcal{E}(s_i)\}_{i=1}^{t}$ immediately after being produced. Let k denote the decryption key. The first possible scheme is to split k into n shares by drawing n points from the following polynomial:

$$f_1(x) \equiv k + \sum_{j=1}^{t-1} r_j x^j \pmod{P}, \tag{5.20}$$

where $\{r_j\}_{j=1}^{t-1}$ are $t-1$ randomly chosen integers. The encrypted data has to be stored in a database so that when t or more participants reconstruct the key collaboratively, they can retrieve and decrypt the data. Nonetheless, the database

may be vulnerable to numerous cyber attacks. The second possible scheme is to create shares according to the following polynomial:

$$f_2(x) \equiv \sum_{i=1}^{t} \mathcal{E}(s_i)x^{i-1} \pmod{P}. \tag{5.21}$$

When t or more participants co-operate, they can reconstruct the encrypted data. In order to decrypt the data, the scheme must engage a key distribution protocol to share the key amongst the participants. To compensate for the shortcomings, one may think of combining the previous two solutions and build a polynomial in the following form:

$$f_3(x) \equiv k + \sum_{j=1}^{t-1} \mathcal{E}(s_j)x^{j} \pmod{P}. \tag{5.22}$$

In this way, the authorised subset of participants is able to reconstruct and decrypt the data from the shares. With the knowledge of the key, however, the dealer is able to decipher the data and thus the privacy is threatened. Regrettably, as previous strategies all have obvious limitations, we need to find another way to do so.

For a moment, let us forget about the problem of sharing ciphertexts and only consider sharing the plaintexts since extending the idea to the sharing of the ciphertexts is easy once the following concepts are understood. Let $\mathbf{s}_{t,1}$ denote a vector of t secrets, $\mathbf{y}_{n,1}$ denote a vector of n shares, and $\mathbf{X}_{n,t}$ denote an $n \times t$ matrix. We define an encoding function

$$\mathbf{y}_{n,1} = \mathbf{X}_{n,t} \cdot \mathbf{s}_{t,1} \tag{5.23}$$

and a decoding function

$$\mathbf{s}_{t,1} = \mathbf{X}_{t,t}^{-1} \cdot \mathbf{y}_{t,1}, \tag{5.24}$$

where $\mathbf{y}_{t,1} \subset \mathbf{y}_{n,1}$, and $\mathbf{X}_{t,t} \subset \mathbf{X}_{n,t}$. In the case of (n, n)-threshold secret sharing, the above encoding and decoding functions are equivalent to Eq. (5.17) and Eq. (5.18), respectively. In the previous special case, we only require that $\mathbf{X}_{n,n}$ has a modular multiplicative inverse. In the current generalised case, however, we require that any $t \times t$ sub-matrix of $\mathbf{X}_{n,t}$ has a modular multiplicative inverse. In fact, when

$t = n$, the current requirement reduces to the previous one since the one and only sub-matrix of $\mathbf{X}_{n,t}$ is $\mathbf{X}_{n,t}$ itself. Our question is hence 'is it possible to construct a valid matrix $\mathbf{X}_{n,t}$ such that any square matrix $\mathbf{X}_{t,t}$ consisting of t rows of $\mathbf{X}_{n,t}$ has a multiplicative inverse?'.

A matrix $\mathbf{A}$ is invertible if and only if its determinant is non-zero. When t and n are small numbers, we could use trial and error to construct a valid $\mathbf{X}_{n,t}$ such that $\det(\mathbf{X}_{t,t}) \neq 0$ for any $\mathbf{X}_{t,t}$. This approach is, however, not practical since collisions become difficult to be handled as the ratio between n and t, implying the number of possible combinations, grows large. To obtain a valid matrix in a systematic way, one of the possible solutions is to construct a Vandermonde matrix.

Definition (Vandermonde matrix)**.** *An $n \times t$ Vandermonde matrix has a form*

$$\mathbf{A}_{n,t} = \begin{pmatrix} {\alpha_1}^0 & {\alpha_1}^1 & {\alpha_1}^2 & \cdots & {\alpha_1}^{t-1} \\ {\alpha_2}^0 & {\alpha_2}^1 & {\alpha_2}^2 & \cdots & {\alpha_2}^{t-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ {\alpha_n}^0 & {\alpha_n}^1 & {\alpha_n}^2 & \cdots & {\alpha_n}^{t-1} \end{pmatrix}.$$

For a $t \times t$ square Vandermonde matrix, the determinant is given by

$$\det(\mathbf{A}_{t,t}) = \prod_{1 \leq i < j \leq t} (\alpha_j - \alpha_i).$$

Example. *Let us compute* $\det(\mathbf{A})$*, where*

$$\mathbf{A} = \begin{pmatrix} 1 & \alpha_1 & {\alpha_1}^2 \\ 1 & \alpha_2 & {\alpha_2}^2 \\ 1 & \alpha_3 & {\alpha_3}^2 \end{pmatrix}.$$

The determinant of $\mathbf{A}$ is given by

$$
\begin{aligned}
&\det \begin{pmatrix} 1 & \alpha_1 & {\alpha_1}^2 \\ 1 & \alpha_2 & {\alpha_2}^2 \\ 1 & \alpha_3 & {\alpha_3}^2 \end{pmatrix} \\
&= \det \begin{pmatrix} 1 & \alpha_1 & {\alpha_1}^2 \\ 0 & \alpha_2 - \alpha_1 & {\alpha_2}^2 - {\alpha_1}^2 \\ 0 & \alpha_3 - \alpha_1 & {\alpha_3}^2 - {\alpha_1}^2 \end{pmatrix} \\
&= \det \begin{pmatrix} \alpha_2 - \alpha_1 & {\alpha_2}^2 - {\alpha_1}^2 \\ \alpha_3 - \alpha_1 & {\alpha_3}^2 - {\alpha_1}^2 \end{pmatrix} \\
&= (\alpha_2 - \alpha_1)(\alpha_3 - \alpha_1) \det \begin{pmatrix} 1 & \alpha_2 - \alpha_1 \\ 1 & \alpha_3 - \alpha_1 \end{pmatrix} \\
&= (\alpha_2 - \alpha_1)(\alpha_3 - \alpha_1) \det \begin{pmatrix} 1 & \alpha_2 - \alpha_1 \\ 0 & \alpha_3 - \alpha_2 \end{pmatrix} \\
&= (\alpha_2 - \alpha_1)(\alpha_3 - \alpha_1)(\alpha_3 - \alpha_2)
\end{aligned}
$$

Corollary (Invertible Vandermonde matrix)**.** *A square Vandermonde matrix is invertible if and only if all α_i are distinct. When the condition suffices, the matrix has a nonzero determinant.*

Given the above preliminaries, we can start with the detailed construction of sharing ciphertexts. Consider a Vandermonde matrix written as

$$
\mathbf{X}_{n,t} = \begin{pmatrix} {x_1}^0 & {x_1}^1 & {x_1}^2 & \cdots & {x_1}^{t-1} \\ {x_2}^0 & {x_2}^1 & {x_2}^2 & \cdots & {x_2}^{t-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ {x_n}^0 & {x_n}^1 & {x_n}^2 & \cdots & {x_n}^{t-1} \end{pmatrix}, \tag{5.25}
$$

where $\{x_i\}_{i=1}^{n}$ are all distinct. The shares are created as

$$\begin{aligned}
\mathcal{E}(y_1) &= \mathcal{E}(s_1)^{{x_1}^0}\mathcal{E}(s_2)^{{x_1}^1}\cdots\mathcal{E}(s_t)^{{x_1}^{t-1}},\\
\mathcal{E}(y_2) &= \mathcal{E}(s_1)^{{x_2}^0}\mathcal{E}(s_2)^{{x_2}^1}\cdots\mathcal{E}(s_t)^{{x_2}^{t-1}},\\
&\vdots\\
\mathcal{E}(y_n) &= \mathcal{E}(s_1)^{{x_n}^0}\mathcal{E}(s_2)^{{x_n}^1}\cdots\mathcal{E}(s_t)^{{x_n}^{t-1}}.
\end{aligned} \tag{5.26}$$

Due to privacy homomorphisms, the above results are equivalent to

$$\begin{aligned}
\mathcal{E}(y_1) &= \mathcal{E}(s_1{x_1}^0 + s_2{x_1}^1 + \cdots + s_t{x_1}^{t-1}),\\
\mathcal{E}(y_2) &= \mathcal{E}(s_1{x_2}^0 + s_2{x_2}^1 + \cdots + s_t{x_2}^{t-1}),\\
&\vdots\\
\mathcal{E}(y_n) &= \mathcal{E}(s_1{x_n}^0 + s_2{x_n}^1 + \cdots + s_t{x_n}^{t-1}).
\end{aligned} \tag{5.27}$$

After decryption, the results become

$$\begin{aligned}
y_1 &= s_1{x_1}^0 + s_2{x_1}^1 + \cdots + s_t{x_1}^{t-1},\\
y_2 &= s_1{x_2}^0 + s_2{x_2}^1 + \cdots + s_t{x_2}^{t-1},\\
&\vdots\\
y_n &= s_1{x_n}^0 + s_2{x_n}^1 + \cdots + s_t{x_n}^{t-1},
\end{aligned} \tag{5.28}$$

or alternatively, as expressed in Eq. (5.23). Each participant will receive a share (x_i, y_i), where $i \in \{1, 2, \ldots, n\}$. Suppose that a subset of participants has gathered a collection of shares, say, (x_j, y_j), where $j \in \{1, 2, \ldots, t\}$. Hence, the participants form a matrix

$$\mathbf{X}_{t,t} = \begin{pmatrix}
{x_1}^0 & {x_1}^1 & {x_1}^2 & \cdots & {x_1}^{t-1}\\
{x_2}^0 & {x_2}^1 & {x_2}^2 & \cdots & {x_2}^{t-1}\\
\vdots & \vdots & \vdots & \ddots & \vdots\\
{x_t}^0 & {x_t}^1 & {x_t}^2 & \cdots & {x_t}^{t-1}
\end{pmatrix}, \tag{5.29}$$

and reconstruct the secrets with Eq. (5.24). Note that $\mathbf{X}_{t,t}$ is a square Vandermonde matrix, thus invertible. The reader may have observed that when $\mathbf{X}_{n,t}$ is a Vander-

monde matrix, Eq. (5.23) and Eq. (5.24) are the encoding and decoding functions of Shamir's scheme *per se*. Let $f(x)$ denote Shamir's encoding function, while $g(x)$ denotes ours. The connection between two functions can be expressed as

$$g(x) = \prod_{i=1}^{t} \mathcal{E}(s_i)^{x^{i-1}} = \mathcal{E}(\sum_{i=1}^{t} s_i x^{i-1}) = \mathcal{E}(f(x)). \tag{5.30}$$

Except for the processing domain (either the plaintext or ciphertext domain), a noticeable difference between two schemes is the decoding process for which Shamir uses the Lagrange interpolation and we utilise a matrix multiplication. We remark that there are many studies on fast algorithms for matrix inversion [251] and multiplication [252–254].

5.3 Summary

In this chapter, we address a novel research problem of secret sharing in the encrypted domain for IoT-based healthcare applications. We study the problem of sharing encrypted data, acquired from different sensor nodes, among a set of cloud servers. In conclusion, the proposed schemes are theoretically secure in the following senses. First, since the secret data is concealed by a secure encryption algorithm immediately after its creation, the dealer as well as other sources cannot access the secret data. Second, the key server only has partial shares and thus is also unable to retrieve the secret data. Third, conforming with the access policy, a subset of fewer than a certain number of participants does not suffice to decode the secrets either. In addition to this, the data is not required to be stored in a common database so that the scheme is not vulnerable to cyber threats against the database. Furthermore, since data retrieval does not involve computationally expensive decryption operations, the scheme is advantageous in time-sensitive circumstances.

On the other hand, however, the proposed methods may have some limitations since they are based on one of the most basic and fundamental secret sharing scheme, namely Shamir's scheme. That is to say, although the proposed schemes are theoretically secure, they might not be sufficient strong solutions when putting into

a more complex environment (e.g. dishonest parties involved) and considering an extended access structures. We would like to emphasise that the proposed methods are intended to serve as prototypes of privacy-preserving secret sharing. In the near future we intend to extend this work into a more general access structure based on the assumption that there are dishonest parties involved. Another line of further investigation is the application of this work in visual cryptography.

Chapter 6

Conclusion

The objectives of this thesis are to explore novel privacy-preserving information hiding techniques and their applications. Although conventional information hiding techniques have been shown effective and widely used for multimedia content protection, there are privacy and security concerns when applying these techniques in cloud computing environments. In spite of the advantages and benefits of cloud computing, the society has undergone a number of catastrophic incidents regarding data breaches and privacy invasions. In order to address privacy threats against conventional information hiding, we develop various privacy-preserving information hiding techniques that deals with encrypted multimedia.

In this thesis, a variety of privacy-preserving information hiding techniques as well as their real-world applications in the contexts of cyber security and cloud computing have been presented. The experimental results have been used to verify their effectiveness and to show the improvements over the state-of-the-art with regards to various performance factors. A novel research of privacy-preserving secret sharing was also introduced in order to realise a secure access control for IoT-based healthcare systems.

6.1 Thesis Summary

The objectives of this thesis were to explore information hiding and secret sharing techniques for various privacy-aware applications. We demonstrated the applicability

of proposed approaches to all sorts of widely-used symmetric-key and asymmetric-key cryptosystems, including stream ciphers, RSA cryptosystem, Paillier cryptosystem, and other cryptosystems sharing the similar homomorphic attributes.

Background

In Chapter 2, a brief introduction of related disciplines and a survey of representative prior art were provided. The problem of concealing information under carrier media has a long and successful history. Information hiding can be broadly divided into two branches of research according to different purposes and characteristics. Steganography aims to conceal the very fact that a secret communication takes place. By contrast, watermarking serves as a protection mechanism against a variety of illegitimate attacks concerning the carrier media, including but not limited to copyright infringement, data forgery, illegal distribution, and malicious tampering.

In spite the fact that imperceptible distortions inflicted by the act of information hiding could be admissible and tolerable in common usage, there are situations in which such alterations would be strictly restricted, for example, in military reconnaissance or medical diagnosis. It becomes a more critical concern by taking account of the artificial intelligence aided automated systems, such as autonomous vehicle systems and autonomous diagnostic systems, which could be unfavourably affected by subtle perturbations if not being properly trained.

Modern cryptography serves as the basis of privacy-preserving signal processing systems. In general, symmetric-key algorithms are of higher computational efficiency, while asymmetric-key algorithms are of higher level of security due to the revocation of key exchange. The prior art of privacy-preserving information hiding can be categorised by the class of cryptosystems on which it is based. The schemes can also be characterised by whether some compulsory preprocessing steps takes place prior to encryption. A scheme is referred to as reserving room before encryption (RRBE) if preprocessing is obligatory and vacating room after encryption (VRAE) if otherwise. Although it is arguable that there are both pros and cons of each paradigm, we drew attention to the paradigm that requires no specifically arranged preprocessing by identifying the flexibility and generality as a high priority.

Information Hiding Based on Symmetric Cryptography

In Chapter 3, two privacy-preserving reversible information hiding schemes based upon stream ciphers were presented. The proposed schemes can be used to prevent data exfiltration in such a way that a permission code is embedded into an encrypted image and can be detected by a network administrator when transmissions occur. The permission code can be detected and read in the encrypted domain, which preserves the content privacy throughout storage, retrieval, and transfer.

The first scheme exploits the arithmetic of quadratic residues to encode messages into an encrypted image and a projection-based predictive model to recover the original image copy. A quadratic residue has four square roots and this property can be utilised to encode payloads in a dynamic fashion. If we regard an element of carrier image as a square root, we can represent the intended message by altering the element to one of the square roots which share the same quadratic residue. Recovering the original copy of image is equivalent to distinguishing the original element from the given set of square roots. This aim can be fulfilled by a specifically devised predictive model that produces an estimation of the original element. We devised an efficient weighted predictive model that computes the weight parameters based upon Hilbert projection theorem.

The second scheme utilises lexicographic permutations for message embedding and a gradient-based estimation mechanism for assisting the carrier signal recovery. We started with a scheme that embeds payloads by changing the ordering of a sequence of carrier elements to a particular permutation. It was observed that in some extreme cases where elements are similar to one another, despite permuting the elements, the perturbed sequence may still be similar to the original sequence. This would cause ambiguity when attempting to distinguish the original sequence from the wrong candidates. Hence, we introduced an invertible transform to further randomise the perturbed elements in order to minimise the ambiguity during the recovery process. In addition to this, we devised a content-adaptive predictor which exploits edge gradient to estimate pixel values, and used this predictor to assist the recovery of original carrier image.

Experimental results showed that the proposed schemes outperformed the

state-of-the-art in terms of capacity, fidelity and reversibility in most cases observed.

Information Hiding Based on Asymmetric Cryptography

In Chapter 4, two privacy-preserving reversible information hiding schemes based respectively upon the RSA and Paillier cryptosystems were proposed. The proposed schemes permit a party to outsource the task of watermarking to a cloud service provider without compromising information privacy. The framework is nearly identical to the conventional framework of watermarking with the only difference that the watermark embedding process is conducted in the encrypted domain. The early development of cloud-based schemes were primarily based upon traditional symmetric-key ciphers and thus the security might be endangered during key exchange.

Although schemes compatible with asymmetric-key cryptosystems have undergone considerable development in recent, there were still some practical weaknesses to be tackled. In particular, the prior art either handles the encryption scheme in an inefficient manner that expands the size of data tremendously, or requires specially designed preprocessing for carrier signals and thus sacrifices the universality and applicability to act on unprocessed carrier signals. To address these issues, we proposed a novel research paradigm and presented schemes compatible with different partially homomorphic cryptosystems. By taking the practicality and availability into consideration, it would be preferable to design a scheme based upon partially homomorphic cryptosystems, instead of fully homomorphic cryptosystems. The proposed schemes are developed for multiplicative homomorphism such as the RSA cryptosystem and additive homomorphism such as the Paillier cryptosystem. The watermark embedding function is viewed as noise adding in the encrypted domain, whereas the joint watermark extraction and carrier recovery function is perceived as a special denoising process that distinguishes the original host symbol from a set of possible candidates. For the RSA-like cryptosystems, the proposed scheme can adjust the embedding capacity readily and flexibly by managing the number of possible candidates. However, watermark extraction and image recovery are not optimal since the scheme produces random distance between each candidate instead of the farthest distance. In contrast to this, for the Paillier-like cryptosystems, the scheme

produces the farthest distance between two candidates and therefore is optimal with respect to the reversibility. Yet, the number of candidate is limited to two and thus the fixed capacity.

We devised an online predictive model and an offline predictive model to cope with the denoising problem. The online predictor is based upon total variation denoising, which is operated on a terminal machine iteratively until the carrier image is restored to a steady state. The offline predictor is based upon Bayesian inference, which utilises a pre-constructed probability table to guide the restoration of contaminated parts of the image. A three-way trade-off between the capacity, fidelity and reversibility was analysed and the experimental results showed that the proposed schemes achieved the state-the-art performance.

Privacy-Preserving Secret Sharing

In Chapter 5, three privacy-preserving secret sharing schemes were developed with different levels of generality. The schemes permit secrets generated from different source to be shared amongst a group of terminal users. The secrets is encrypted before being transmitted to a central hub for the distribution process that follows. The retrieval of secrets must conform to the pre-defined access policy. Hence, the problem can be viewed as secret sharing in the encrypted domain. This technique can be applied to privacy protection in Internet of things enabled healthcare systems. Consider a context that an individual's health records are collected from different wearable equipments and sensor nodes. In order to safeguard the privacy of patients and to impose access control, these records are encrypted and distributed to a group of medical practitioners and clinical analysts. The data is protected in such a way that it can only be accessed if a sufficient number of authorised individuals agree to retrieve it.

We began with naïve solutions which, despite the feasibility, encountered the issues of key distribution and time-consuming data retrieval process. The data retrieval process should be computationally efficient when taking account of surgical emergency. In response to these problems, we proposed a (2, 2)-threshold scheme and further derived an (n, n)-threshold scheme. Finally, we developed a generalised

(t, n)-threshold scheme by using the invertible Vandermonde matrix. The proposed (t, n)-threshold scheme provides a general access control that enables a group of t out of n authorised share holders to reconstruct the secrets. We showed that the proposed scheme can be viewed as an extension of Shamir's secret sharing scheme for encrypted data.

6.2 Future Work

Although this thesis has presented technical principles and practical applications of privacy-preserving information hiding, rigorous security analysis is also of significant importance and thus should be taken into further consideration. Privacy-preserving information hiding is a multidisciplinary research encompassing cryptography, steganography, watermarking, and other areas of knowledge to which the systems are applied. In spite of the fact that the proposed schemes are based upon theoretically secure cryptographic algorithms, it is possible that some novel types of security threats exist depending on different applied environments. Malicious attempts to remove or replace the embedded messages as well as other means to corrupt and compromise the systems entail more attention. In the meantime, although potential applications in the context of cyber security, cloud computing, and Internet of things were discussed, some novel applications would also be favourable. In view of the growing research trends in autonomous vehicle technology, it might be possible to apply privacy-preserving information hiding techniques to vehicle telematics. For instance, these techniques could be integrated into navigation systems to protect and authenticate vehicular communications. Aside from this, fake news and its dissemination on social media has drawn increasing attention over time. Fingerprinting is a special class of digital watermarking that is widely used for tracing the unauthorised distribution of digital objects. It may also be applied to the study of fake news spread patterns through embedding fingerprint codes into social media posts.

Previous studies of privacy-preserving reversible information hiding have led to a wide variety of methods. This thesis has categorised the prior art by the cryptosystems on which it is based and further divided it into the class of

reserving room before encryption (RRBE) and the class of vacating room after encryption (VRAE). However, it would not be surprising if a scheme is composed of characteristics beyond these definitions. Therefore, a more constructive and systematic review of the state-of-the-art would be especially beneficial. In addition to this, a more general model integrating different types of schemes deserves further investigation. Improvements over the latest predictive models are expected through the use of advanced neural networks and machine learning techniques. Regarding the research of privacy-preserving secret sharing, although the proposed scheme permits a general access threshold, more complex access structures can be further studied. It is equally significant to develop theoretical analysis of security and take into account application-oriented attacks.

In conclusion, privacy-preserving information hiding is a rather new area of knowledge that requires efforts from various research communities. It is hoped that novel and original ideas presented in this thesis can be of the building blocks for more future research devoted to this research area.

6.3 Concluding Remarks

In summary, the main contributions of this thesis are:

- two privacy-preserving reversible information hiding schemes based upon symmetric cryptography using respectively arithmetic of quadratic residues and lexicographic permutations.
- two privacy-preserving reversible information hiding schemes based upon asymmetric cryptography using respectively multiplicative and additive privacy homomorphisms.
- four predictive models for assisting the recovery of original signals using respectively projection theorem, image gradient, total variation denoising, and Bayesian inference.
- three privacy-preserving secret sharing algorithms in ascending level of generality.

- a variety of applications in the contexts of cloud computing, cyber security, Internet of things, *etc.*

Bibliography

[1] S. Issenberg, "How Obama's team used big data to rally voters," *MIT Technology Review*, Dec. 2012.

[2] C. Cadwalladr and E. Graham-Harrison, "Revealed: 50 million Facebook profiles harvested for Cambridge Analytica in major data breach," *The Guardian*, Mar. 2018.

[3] Digital, Culture, Media and Sport Committee, "Disinformation and 'fake news': Final report," *House of Commons*, Feb. 2019.

[4] L. M. Kaufman, "Data security in the world of cloud computing," *IEEE Security & Privacy*, vol. 7, no. 4, pp. 61–64, July 2009.

[5] H. Takabi, J. B. D. Joshi, and G. Ahn, "Security and privacy challenges in cloud computing environments," *IEEE Security & Privacy*, vol. 8, no. 6, pp. 24–31, Nov. 2010.

[6] D. Zissis and D. Lekkas, "Addressing cloud computing security issues," *Future Generation Computer Systems*, vol. 28, no. 3, pp. 583–592, Mar. 2012.

[7] R. L. Lagendijk, Z. Erkin, and M. Barni, "Encrypted signal processing for privacy protection: Conveying the utility of homomorphic encryption and multiparty computation," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 82–105, Jan. 2013.

[8] C. Aguilar-Melchor, S. Fau, C. Fontaine, G. Gogniat, and R. Sirdey, "Recent advances in homomorphic encryption: A possible future for signal processing in the encrypted domain," *IEEE Signal Processing Magazine*, vol. 30, no. 2, pp. 108–117, Mar. 2013.

[9] J. R. Troncoso-pastoriza and F. Perez-Gonzalez, "Secure signal processing in the cloud: Enabling technologies for privacy-preserving multimedia cloud processing," *IEEE Signal Processing Magazine*, vol. 30, no. 2, pp. 29–41, Mar. 2013.

[10] I. A. T. Hashem, I. Yaqoob, N. B. Anuar, S. Mokhtar, A. Gani, and S. U. Khan, "The rise of "big data" on cloud computing: Review and open research issues," *Information Systems*, vol. 47, no. Supplement C, pp. 98–115, Jan. 2015.

[11] R. Agrawal and R. Srikant, "Privacy-preserving data mining," in *Proceedings of the ACM International Conference on Management of Data (SIGMOD)*, Dallas, TX, USA, May 2000, pp. 439–450.

[12] M. Kantarcioglu and C. Clifton, "Privacy-preserving distributed mining of association rules on horizontally partitioned data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 9, pp. 1026–1037, Sept. 2004.

[13] Y.-C. Chang and M. Mitzenmacher, "Privacy preserving keyword searches on remote encrypted data," in *Proceedings of the International Conference on Applied Cryptography and Network Security (ACNS)*, New York, NY, USA, June 2005, pp. 442–455.

[14] Z. Erkin, A. Piva, S. Katzenbeisser, R. L. Lagendijk, J. Shokrollahi, G. Neven, and M. Barni, "Protection and retrieval of encrypted multimedia content: When cryptography meets signal processing," *EURASIP Journal on Information Security*, vol. 2007, pp. 17:1–17:20, Jan. 2007.

[15] A. B. Chan, , and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Anchorage, AK, USA, June 2008, pp. 1–7.

[16] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," *IEEE Transactions on Computers*, vol. 62, no. 2, pp. 362–375, Feb. 2013.

[17] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 1, pp. 222–233, Jan. 2014.

[18] M. Barni, G. Droandi, and R. Lazzeretti, "Privacy protection in biometric-based recognition systems: A marriage between cryptography and signal processing," *IEEE Signal Processing Magazine*, vol. 32, no. 5, pp. 66–76, Sept. 2015.

[19] Z. Xia, X. Wang, L. Zhang, Z. Qin, X. Sun, and K. Ren, "A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing," *IEEE*

Transactions on Information Forensics and Security, vol. 11, no. 11, pp. 2594–2608, Nov. 2016.

[20] X. Wei, C. Li, Z. Lei, D. Yi, and S. Z. Li, "Dynamic image-to-class warping for occluded face recognition," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 12, pp. 2035–2050, Dec. 2014.

[21] Y. Guan, C. Li, and F. Roli, "On reducing the effect of covariate factors in gait recognition: A classifier ensemble method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 7, pp. 1521–1528, July 2015.

[22] N. Jia, V. Sanchez, and C. Li, "On view-invariant gait recognition: a feature selection solution," *IET Biometrics*, vol. 7, no. 4, pp. 287–295, June 2018.

[23] C. Li, "Source camera identification using enhanced sensor pattern noise," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 2, pp. 280–287, June 2010.

[24] X. Lin and C. Li, "Preprocessing reference sensor pattern noise via spectrum equalization," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 1, pp. 126–140, Jan. 2016.

[25] X. Lin and C. Li, "Enhancing sensor pattern noise via filtering distortion removal," *IEEE Signal Processing Letters*, vol. 23, no. 3, pp. 381–385, Mar. 2016.

[26] A. Khadidos, V. Sanchez, and C. Li, "Weighted level set evolution based on local edge features for medical image segmentation," *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 1979–1991, Apr. 2017.

[27] X. Guan, C. Li, and Y. Guan, "Matrix factorization with rating completion: An enhanced svd model for collaborative filtering recommender systems," *IEEE Access*, vol. 5, pp. 27 668–27 678, Nov. 2017.

[28] R. Leyva, V. Sanchez, and C. Li, "Video anomaly detection with compact feature sets for online performance," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3463–3478, July 2017.

[29] X. Lin and C. Li, "Large-scale image clustering based on camera fingerprints," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 793–808, Apr. 2017.

[30] N. Kerris and T. Muller, "Update to celebrity photo investigation," *Apple Media Advisory*, Sept. 2014.

[31] T. Bianchi and A. Piva, "Secure watermarking for multimedia content protection: A review of its benefits and open issues," *IEEE Signal Processing Magazine*, vol. 30, no. 2, pp. 87–96, Mar. 2013.

[32] B. Pfitzmann, "Information hiding terminology," in *Proceedings of the International Workshop on Information Hiding (IH)*, Cambridge, UK, May 1996, pp. 347–350.

[33] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding–A survey," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1062–1078, July 1999.

[34] P. Moulin and J. A. O'Sullivan, "Information-theoretic analysis of information hiding," *IEEE Transactions on Information Theory*, vol. 49, no. 3, pp. 563–593, Mar. 2003.

[35] S. Katzenbeisser and F. Petitcolas, *Information Hiding*. Norwood, MA, USA: Artech House, 2016.

[36] I. J. Cox, M. L. Miller, J. A. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, 2nd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2008.

[37] J. Fridrich, *Steganography in Digital Media: Principles, Algorithms, and Applications*. Cambridge, UK: Cambridge University Press, 2010.

[38] J. Trithemius, *The Steganographia of Johannes Trithemius*, A. McLean, Ed. Edinburgh, Scotland: Magnum Opus Hermetic Sourceworks, 1982.

[39] Herodotus, *The Histories*. London, England: J. M. Dent & Sons, 1992.

[40] J. E. Hoover, "The enemy's masterpiece of espionage," *Reader's Digest*, vol. 48, pp. 49–53, May 1946.

[41] G. J. Simmons, "The prisoners' problem and the subliminal channel," in *Advances in Cryptology*, D. Chaum, Ed. Boston, MA, USA: Springer US, 1984, pp. 51–67.

[42] N. F. Johnson and S. Jajodia, "Exploring steganography: Seeing the unseen," *Computer*, vol. 31, no. 2, pp. 26–34, Feb. 1998.

[43] R. J. Anderson and F. A. P. Petitcolas, "On the limits of steganography," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 4, pp. 474–481, May 1998.

[44] N. Provos and P. Honeyman, "Hide and seek: an introduction to steganography," *IEEE Security & Privacy*, vol. 99, no. 3, pp. 32–44, May 2003.

[45] A. D. Ker, "Steganalysis of lsb matching in grayscale images," *IEEE Signal Processing Letters*, vol. 12, no. 6, pp. 441–444, June 2005.

[46] J. Fridrich, M. Goljan, P. Lisonek, and D. Soukal, "Writing on wet paper," *IEEE Transactions on Signal Processing*, vol. 53, no. 10, pp. 3923–3935, Oct. 2005.

[47] J. Mielikainen, "Lsb matching revisited," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 285–287, May 2006.

[48] X. Zhang and S. Wang, "Efficient steganographic embedding by exploiting modification direction," *IEEE Communications Letters*, vol. 10, no. 11, pp. 781–783, November 2006.

[49] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868–882, June 2012.

[50] C.-Y. Chang and S. Clark, "Practical linguistic steganography using contextual synonym substitution and a novel vertex coding method," *Computational Linguistics*, vol. 40, no. 2, pp. 403–448, June 2014.

[51] R. G. van Schyndel, A. Z. Tirkel, and C. F. Osborne, "A digital watermark," in *Proceedings of the International Conference on Image Processing (ICIP)*, vol. 2, Austin, TX, USA, Nov. 1994, pp. 86–90.

[52] F. Hartung and M. Kutter, "Multimedia watermarking techniques," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1079–1107, July 1999.

[53] C. I. Podilchuk and E. J. Delp, "Digital watermarking: algorithms and applications," *IEEE Signal Processing Magazine*, vol. 18, no. 4, pp. 33–46, July 2001.

[54] P. Moulin and R. Koetter, "Data-hiding codes," *Proceedings of the IEEE*, vol. 93, no. 12, pp. 2083–2126, Dec. 2005.

[55] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1673–1687, Dec. 1997.

[56] D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data," *IEEE Transactions on Information Theory*, vol. 44, no. 5, pp. 1897–1905, Sept. 1998.

[57] R. B. Wolfgang, C. I. Podilchuk, and E. J. Delp, "Perceptual watermarks for digital images and video," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1108–1126, July 1999.

[58] I. J. Cox, M. L. Miller, and A. L. McKellips, "Watermarking as communications with side information," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1127–1141, July 1999.

[59] N. M. and, "A buyer-seller watermarking protocol," *IEEE Transactions on Image Processing*, vol. 10, no. 4, pp. 643–649, Apr. 2001.

[60] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Transactions on Information Theory*, vol. 47, no. 4, pp. 1423–1443, May 2001.

[61] M. Barni, F. Bartolini, and A. Piva, "Improved wavelet-based watermarking through pixel-wise masking," *IEEE Transactions on Image Processing*, vol. 10, no. 5, pp. 783–791, May 2001.

[62] P. Bas, J. . Chassery, and B. Macq, "Geometrically invariant watermarking using feature points," *IEEE Transactions on Image Processing*, vol. 11, no. 9, pp. 1014–1028, Sept. 2002.

[63] H. S. Malvar and D. A. F. Florencio, "Improved spread spectrum: A new modulation technique for robust watermarking," *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 898–905, Apr. 2003.

[64] C.-C. Chang, P. Tsai, and C.-C. Lin, "SVD-based digital image watermarking scheme," *Pattern Recognition Letters*, vol. 26, no. 10, pp. 1577–1586, July 2005.

[65] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on copyright marking systems," in *Proceedings of the International Workshop on Information Hiding (IH)*, D. Aucsmith, Ed., Portland, OR, USA, Nov. 1998, pp. 218–238.

[66] M. M. Yeung and F. Mintzer, "An invisible watermarking technique for image verification," in *Proceedings of the International Conference on Image Processing (ICIP)*, vol. 2, Santa Barbara, CA, USA, Oct. 1997, pp. 680–683.

[67] J. Fridrich, "Image watermarking for tamper detection," in *Proceedings of the International Conference on Image Processing (ICIP)*, vol. 2, Chicago, IL, USA, Oct. 1998, pp. 404–408.

[68] D. Kundur and D. Hatzinakos, "Digital watermarking for telltale tamper proofing and authentication," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1167–1180, July 1999.

[69] M. Holliman and N. Memon, "Counterfeiting attacks on oblivious block-wise independent invisible watermarking schemes," *IEEE Transactions on Image Processing*, vol. 9, no. 3, pp. 432–441, Mar. 2000.

[70] E. T. Lin, C. I. Podilchuk, and E. J. Delp, "Detection of image alterations using semifragile watermarks," in *Proceedings of SPIE*, vol. 3971, San Jose, CA, USA, May 2000, pp. 152–163.

[71] C.-Y. Lin and S.-F. Chang, "Semifragile watermarking for authenticating jpeg visual content," in *Proceedings of SPIE*, vol. 3971, San Jose, CA, USA, May 2000, pp. 140–151.

[72] H. Si and C.-T. Li, "Fragile watermarking scheme based on the block-wise dependence in the wavelet domain," in *Proceedings of the Workshop on Multimedia and Security (MM&Sec)*, Magdeburg, Germany, Sept. 2004, pp. 214–219.

[73] C.-T. Li, "Digital fragile watermarking scheme for authentication of JPEG images," *IEE Proceedings - Vision, Image Signal Processing*, vol. 151, no. 6, pp. 460–466, Dec. 2004.

[74] J. Fridrich, "Security of fragile authentication watermarks with localization," in *Proceedings of SPIE*, vol. 4675, San Jose, CA, USA, Apr. 2002, pp. 691–700.

[75] M. U. Celik, G. Sharma, E. Saber, and A. M. Tekalp, "Hierarchical watermarking for secure image authentication with localization," *IEEE Transactions on Image Processing*, vol. 11, no. 6, pp. 585–595, June 2002.

[76] A. H. Ouda and M. R. El-Sakka, "Localization and security enhancement of block-based image authentication," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, vol. 1, Genova, Italy, Sept. 2005, pp. 673–676.

[77] C.-C. Chang, Y.-S. Hu, and T.-C. Lu, "A watermarking-based image ownership and tampering authentication scheme," *Pattern Recognition Letters*, vol. 27, no. 5, pp. 439–446, Apr. 2006.

[78] H. Liu and M. Steinebach, "Digital watermarking for image authentication with localization," in *Proceedings of the International Conference on Image Processing (ICIP)*, Atlanta, GA, USA, Oct. 2006, pp. 1973–1976.

[79] C.-C. Chang, K.-N. Chen, C.-F. Lee, and L.-J. Liu, "A secure fragile watermarking scheme based on chaos-and-hamming code," *Journal of Systems and Software*, vol. 84, no. 9, pp. 1462–1470, Sept. 2011.

[80] J. Lee and C. S. Won, "Authentication and correction of digital watermarking images," *Electronics Letters*, vol. 35, no. 11, pp. 886–887, May 1999.

[81] J. Fridrich and M. Goljan, "Images with self-correcting capabilities," in *Proceedings of the International Conference on Image Processing (ICIP)*, vol. 3, Kobe, Japan, Oct. 1999, pp. 792–796.

[82] X. Zhu, A. T. Ho, and P. Marziliano, "A new semi-fragile image watermarking with robust tampering restoration using irregular sampling," *Signal Processing: Image Communication*, vol. 22, no. 5, pp. 515–528, June 2007.

[83] T.-Y. Lee and S. D. Lin, "Dual watermark for image tamper detection and recovery," *Pattern Recognition*, vol. 41, no. 11, pp. 3497–3506, Nov. 2008.

[84] X. Zhang and S. Wang, "Fragile watermarking with error-free restoration capability," *IEEE Transactions on Multimedia*, vol. 10, no. 8, pp. 1490–1499, Dec. 2008.

[85] C.-W. Yang and J.-J. Shen, "Recover the tampered image based on vq indexing," *Signal Processing*, vol. 90, no. 1, pp. 331–343, Jan. 2010.

[86] X. Zhang, S. Wang, Z. Qian, and G. Feng, "Reference sharing mechanism for watermark self-embedding," *IEEE Transactions on Image Processing*, vol. 20, no. 2, pp. 485–495, Feb. 2011.

[87] X. Zhang, Z. Qian, Y. Ren, and G. Feng, "Watermarking with flexible self-recovery quality based on compressive sensing and compositive reconstruction," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 4, pp. 1223–1232, Dec. 2011.

[88] C. Qin, C.-C. Chang, and P.-Y. Chen, "Self-embedding fragile watermarking with restoration capability based on adaptive bit allocation mechanism," *Signal Processing*, vol. 92, no. 4, pp. 1137–1150, Apr. 2012.

[89] P. Korus and A. Dziech, "Efficient method for content reconstruction with self-embedding," *IEEE Transactions on Image Processing*, vol. 22, no. 3, pp. 1134–1147, Mar. 2013.

[90] P. Korus and A. Dziech, "Adaptive self-embedding scheme with controlled reconstruction performance," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 2, pp. 169–181, Feb. 2014.

[91] P. Korus, J. Białas, and A. Dziech, "Towards practical self-embedding for jpeg-compressed digital images," *IEEE Transactions on Multimedia*, vol. 17, no. 2, pp. 157–170, Feb. 2015.

[92] S. Sarreshtedari and M. A. Akhaee, "A source-channel coding approach to digital image protection and self-recovery," *IEEE Transactions on Image Processing*, vol. 24, no. 7, pp. 2266–2277, July 2015.

[93] M. Hamid and C. Wang, "Adaptive image self-recovery based on feature extraction in the dct domain," *IEEE Access*, vol. 6, pp. 67 156–67 165, Nov. 2018.

[94] A. M. Eskicioglu and P. S. Fisher, "Image quality measures and their performance," *IEEE Transactions on Communications*, vol. 43, no. 12, pp. 2959–2965, Dec. 1995.

[95] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Processing Letters*, vol. 9, no. 3, pp. 81–84, Mar. 2002.

[96] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[97] A. W. Rix, A. Bourret, and M. P. Hollier, "Models of human perception," *BT Technology Journal*, vol. 17, no. 1, pp. 24–34, Jan. 1999.

[98] B. B. Zhu, M. D. Swanson, and A. H. Tewfik, "When seeing isn't believing: Multimedia authentication technologies," *IEEE Signal Processing Magazine*, vol. 21, no. 2, pp. 40–49, Mar. 2004.

[99] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978.

[100] G. L. Friedman, "The trustworthy digital camera: Restoring credibility to the photographic image," *IEEE Transactions on Consumer Electronics*, vol. 39, no. 4, pp. 905–910, Nov. 1993.

[101] I. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proceedings of the International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, May 2015, pp. 1–11.

[102] Y.-Q. Shi, X. Li, X. Zhang, H.-T. Wu, and B. Ma, "Reversible data hiding: Advances in the past two decades," *IEEE Access*, vol. 4, pp. 3210–3237, May 2016.

[103] J. M. Barton, "Method and apparatus for embedding authentication information within digital data," U.S. Patent 5 646 997, July 8, 1997.

[104] J. Fridrich, M. Goljan, and R. Du, "Invertible authentication," in *Proceedings of SPIE*, vol. 4314, San Jose, CA, USA, Jan. 2001, pp. 197–208.

[105] M. Goljan, J. J. Fridrich, and R. Du, "Distortion-free data embedding for images," in *Proceedings of the International Workshop on Information Hiding (IH)*, Pittsburgh, PA, USA, Apr. 2001, pp. 27–41.

[106] J. Fridrich, M. Goljan, and R. Du, "Lossless data embedding: New paradigm in digital watermarking," *EURASIP Journal on Advances in Signal Processing*, vol. 2002, no. 2, pp. 185–196, Feb. 2002.

[107] G. Xuan, J. Zhu, J. Chen, Y.-Q. Shi, Z. Ni, and W. Su, "Distortionless data hiding based on integer wavelet transform," *Electronics Letters*, vol. 38, no. 25, pp. 1646–1648, Dec. 2002.

[108] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Lossless generalized-lsb data embedding," *IEEE Transactions on Image Processing*, vol. 14, no. 2, pp. 253–266, Feb. 2005.

[109] M. U. Celik, G. Sharma, and A. M. Tekalp, "Lossless watermarking for image authentication: a new framework and an implementation," *IEEE Transactions on Image Processing*, vol. 15, no. 4, pp. 1042–1049, Apr. 2006.

[110] T. Kalker and F. M. J. Willems, "Capacity bounds and constructions for reversible data-hiding," in *Proceedings of the International Conference on Digital Signal Processing (DSP)*, Santorini, Greece, July 2002, pp. 71–76.

[111] W. Zhang, B. Chen, and N. Yu, "Improving various reversible data hiding schemes via optimal codes for binary covers," *IEEE Transactions on Image Processing*, vol. 21, no. 6, pp. 2991–3003, June 2012.

[112] S. Lin and W. Chung, "The scalar scheme for reversible information-embedding in gray-scale signals: Capacity evaluation and code constructions," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 4, pp. 1155–1167, Aug. 2012.

[113] X. Zhang, "Reversible data hiding with optimal value transfer," *IEEE Transactions on Multimedia*, vol. 15, no. 2, pp. 316–325, Feb. 2013.

[114] X. Hu, W. Zhang, X. Hu, N. Yu, X. Zhao, and F. Li, "Fast estimation of optimal marked-signal distribution for reversible data hiding," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 5, pp. 779–788, May 2013.

[115] W. Zhang, X. Hu, X. Li, and N. Yu, "Recursive histogram modification: Establishing equivalency between reversible data hiding and lossless data compression," *IEEE Transactions on Image Processing*, vol. 22, no. 7, pp. 2775–2785, July 2013.

[116] W. Zhang, X. Hu, X. Li, and Y. Nenghai, "Optimal transition probability of reversible data hiding for general distortion metrics and its applications," *IEEE Transactions on Image Processing*, vol. 24, no. 1, pp. 294–304, Jan. 2015.

[117] F. Balado, "Optimum reversible data hiding and permutation coding," in *Proceedings of the IEEE International Workshop on Information Forensics and Security (WIFS)*, Rome, Italy, Nov. 2015, pp. 1–4.

[118] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890–896, Aug. 2003.

[119] A. M. Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE Transactions on Image Processing*, vol. 13, no. 8, pp. 1147–1156, Aug. 2004.

[120] D. Coltuc and J. Chassery, "Very fast watermarking by reversible contrast mapping," *IEEE Signal Processing Letters*, vol. 14, no. 4, pp. 255–258, Apr. 2007.

[121] S. Lee, C. D. Yoo, and T. Kalker, "Reversible image watermarking based on integer-to-integer wavelet transform," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 321–330, Sept. 2007.

[122] H. J. Kim, V. Sachnev, Y. Q. Shi, J. Nam, and H. Choo, "A novel difference expansion transform for reversible data embedding," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 456–465, Sept. 2008.

[123] X. Wang, X. Li, B. Yang, and Z. Guo, "Efficient generalized integer transform for reversible watermarking," *IEEE Signal Processing Letters*, vol. 17, no. 6, pp. 567–570, June 2010.

[124] L. Kamstra and H. J. A. M. Heijmans, "Reversible data embedding into images using wavelet techniques and sorting," *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2082–2090, Dec. 2005.

[125] Z. Ni, Y.-Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 3, pp. 354–362, Mar. 2006.

[126] D. M. Thodi and J. J. Rodriguez, "Expansion embedding techniques for reversible watermarking," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 721–730, Mar. 2007.

[127] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y. Q. Shi, "Reversible watermarking algorithm using sorting and prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 7, pp. 989–999, July 2009.

[128] L. Luo, Z. Chen, M. Chen, X. Zeng, and Z. Xiong, "Reversible image watermarking using interpolation technique," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 1, pp. 187–193, Mar. 2010.

[129] C. Qin, C. Chang, Y. Huang, and L. Liao, "An inpainting-assisted reversible steganographic scheme using a histogram shifting mechanism," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 7, pp. 1109–1118, July 2013.

[130] M. Fallahpour, "Reversible image data hiding based on gradient adjusted prediction," *IEICE Electronics Express*, vol. 5, no. 20, pp. 870–876, Oct. 2008.

[131] D. Coltuc, "Improved embedding for prediction-based reversible watermarking," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 873–882, Sept. 2011.

[132] B. Ou, X. Li, Y. Zhao, R. Ni, and Y. Shi, "Pairwise prediction-error expansion for efficient reversible data hiding," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 5010–5021, Dec. 2013.

[133] X. Li, B. Yang, and T. Zeng, "Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection," *IEEE Transactions on Image Processing*, vol. 20, no. 12, pp. 3524–3533, Dec 2011.

[134] G. Coatrieux, W. Pan, N. Cuppens-Boulahia, F. Cuppens, and C. Roux, "Reversible watermarking based on invariant image classification and dynamic histogram shifting," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 1, pp. 111–120, Jan 2013.

[135] W. Hong, "Adaptive reversible data hiding method based on error energy control and histogram shifting," *Optics Communications*, vol. 285, no. 2, pp. 101–108, Jan. 2012.

[136] Q. Pei, X. Wang, Y. Li, and H. Li, "Adaptive reversible watermarking with improved embedding capacity," *Journal of Systems and Software*, vol. 86, no. 11, pp. 2841–2848, Nov. 2013.

[137] W. Hong, T.-S. Chen, and J. Chen, "Reversible data hiding using delaunay triangulation and selective embedment," *Information Sciences*, vol. 308, pp. 140–154, July 2015.

[138] M. Fallahpour and M. H. Sedaaghi, "High capacity lossless data hiding based on histogram modification," *IEICE Electronics Express*, vol. 4, no. 7, pp. 205–210, Apr. 2007.

[139] G. Xuan, Y. Q. Shi, P. Chai, X. Cui, Z. Ni, and X. Tong, "Optimum histogram pair based image lossless data embedding," in *Proceedings of the International Workshop on Digital Watermarking (IWDW)*, Guangzhou, China, Dec. 2007, pp. 264–278.

[140] X. Li, B. Li, B. Yang, and T. Zeng, "General framework to histogram-shifting-based reversible data hiding," *IEEE Transactions on Image Processing*, vol. 22, no. 6, pp. 2181–2191, June 2013.

[141] J. Wang, J. Ni, X. Zhang, and Y. Shi, "Rate and distortion optimization for reversible data hiding using multiple histogram shifting," *IEEE Transactions on Cybernetics*, vol. 47, no. 2, pp. 315–326, Feb. 2017.

[142] X. Li, W. Zhang, X. Gui, and B. Yang, "A novel reversible data hiding scheme based on two-dimensional difference-histogram modification," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 7, pp. 1091–1100, July 2013.

[143] S. Wang, C. Li, and W. Kuo, "Reversible data hiding based on two-dimensional prediction errors," *IET Image Processing*, vol. 7, no. 9, pp. 805–816, Dec. 2013.

[144] I. Caciula and D. Coltuc, "Improved control for low bit-rate reversible watermarking," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May 2014, pp. 7425–7429.

[145] X. Li, W. Zhang, X. Gui, and B. Yang, "Efficient reversible data hiding based on multiple histograms modification," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 9, pp. 2016–2027, Sep. 2015.

[146] C. P. Bauer, *Secret History: The Story of Cryptology*. Boca Raton, FL, USA: CRC Press, 2013.

[147] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*, 2nd ed. Boca Raton, FL, USA: CRC Press, 2014.

[148] A. Kerckhoffs, "La cryptographie militaire," *Journal des Sciences Militaires*, vol. IX, pp. 5–38, Jan. 1883.

[149] C. E. Shannon, "Communication theory of secrecy systems," *The Bell System Technical Journal*, vol. 28, no. 4, pp. 656–715, Oct. 1949.

[150] R. J. Anderson, *Security Engineering*, 2nd ed. Indianapolis, IN, USA: Wiley Publishing, Inc., 2008.

[151] G. Vernam, "Secret signaling system," U.S. Patent 1 310 719, July 22, 1919.

[152] S. Fluhrer, I. Mantin, and A. Shamir, "Weaknesses in the key scheduling algorithm of rc4," in *Proceedings of the International Workshop on Selected Areas in Cryptography (SAC)*, Toronto, Ontario, Canada, Aug. 2001, pp. 1–24.

[153] E. Biham and A. Shamir, "Differential cryptanalysis of des-like cryptosystems," *Journal of Cryptology*, vol. 4, no. 1, pp. 3–72, Jan. 1991.

[154] J. Daemen and V. Rijmen, *The Design of Rijndael: AES - The Advanced Encryption Standard*. Springer, 2002.

[155] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, Nov. 1976.

[156] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203–209, Jan. 1987.

[157] R. L. Rivest, L. Adleman, and M. L. Dertouzos, "On data banks and privacy homomorphisms," in *Foundations of Secure Computation*, R. A. DeMillo *et al.*, Eds. Orlando, FL, USA: Academic Press, 1978, pp. 169–180.

[158] M. O. Rabin, "Digitalized signatures and public-key functions as intractable as factorization," MIT Laboratory for Computer Science, Cambridge, MA, USA, Tech. Rep. MIT/LCS/TR-212, Jan. 1979.

[159] S. Goldwasser and S. Micali, "Probabilistic encryption," *Journal of Computer and System Sciences*, vol. 28, no. 2, pp. 270–299, Apr. 1984.

[160] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469–472, Jul. 1985.

[161] J. Benaloh, "Dense probabilistic encryption," in *Proceedings of the Workshop on Selected Areas in Cryptography (SAC)*, Kingston, Canada, May 1994, pp. 120–128.

[162] T. Okamoto and S. Uchiyama, "A new public-key cryptosystem as secure as factoring," in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, Espoo, Finland, May 1998, pp. 308–318.

[163] D. Naccache and J. Stern, "A new public key cryptosystem based on higher residues," in *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, San Francisco, CA, USA, Nov. 1998, pp. 59–66.

[164] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, Prague, Czech Republic, May 1999, pp. 223–238.

[165] I. Damgård and M. Jurik, "A generalisation, a simplication and some applications of Paillier's probabilistic public-key system," in *Proceedings of the International Workshop on Practice and Theory in Public Key Cryptography (PKC)*, Feb. 2001, pp. 119–136.

[166] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the Annual ACM Symposium on Theory of Computing (STOC)*, Bethesda, MD, USA, May 2009, pp. 169–178.

[167] M. van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, "Fully homomorphic encryption over the integers," in *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, French Riviera, France, May 2010, pp. 24–43.

[168] N. P. Smart and F. Vercauteren, "Fully homomorphic encryption with relatively small key and ciphertext sizes," in *Proceedings of the International Conference on Practice and Theory in Public Key Cryptography (PKC)*, Paris, France, May 2010, pp. 420–443.

[169] D. Stehlé and R. Steinfeld, "Faster fully homomorphic encryption," in *Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT)*, Singapore, Dec. 2010, pp. 377–394.

[170] C. Gentry and S. Halevi, "Implementing Gentry's fully-homomorphic encryption scheme," in *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, Tallinn, Estonia, May 2011, pp. 129–148.

[171] Z. Brakerski and V. Vaikuntanathan, "Efficient fully homomorphic encryption from (standard) LWE," in *Proceedings of the IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, Palm Springs, CA, USA, Oct. 2011, pp. 97–106.

[172] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "(Leveled) fully homomorphic encryption without bootstrapping," in *Proceedings of the Innovations in Theoretical Computer Science Conference (ITCS)*, Cambridge, MA, USA, Jan. 2012, pp. 309–325.

[173] J. H. Cheon, J.-S. Coron, J. Kim, M. S. Lee, T. Lepoint, M. Tibouchi, and A. Yun, "Batch fully homomorphic encryption over the integers," in *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, Athens, Greece, May 2013, pp. 315–335.

[174] C. Gentry, A. Sahai, and B. Waters, "Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based," in *Proceedings of the Annual Cryptology Conference (CRYPTO)*, R. Canetti and J. A. Garay, Eds., Santa Barbara, CA, USA, Aug. 2013, pp. 75–92.

[175] J.-S. Coron, T. Lepoint, and M. Tibouchi, "Scale-invariant fully homomorphic encryption over the integers," in *Proceedings of the International Conference on Practice and Theory in Public-Key Cryptography (PKC)*, Buenos Aires, Argentina, Mar. 2014, pp. 311–328.

[176] N. P. Smart and F. Vercauteren, "Fully homomorphic SIMD operations," *Designs, Codes and Cryptography*, vol. 71, no. 1, pp. 57–81, Apr. 2014.

[177] W. Puech, M. Chaumont, and O. Strauss, "A reversible data hiding method for encrypted images," in *Proceedings of SPIE*, vol. 6819, San Jose, CA, USA, Mar. 2008, p. 68191E.

[178] P. Puteaux and W. Puech, "Reversible data hiding in encrypted images based on adaptive local entropy analysis," in *Proceedings of the International Conference on Image Processing Theory, Tools and Applications (IPTA)*, Montreal, Canada, Nov. 2017, pp. 1–6.

[179] P. Puteaux and W. Puech, "An efficient MSB prediction-based method for high-capacity reversible data hiding in encrypted images," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 7, pp. 1670–1681, July 2018.

[180] X. Zhang, "Reversible data hiding in encrypted image," *IEEE Signal Processing Letters*, vol. 18, no. 4, pp. 255–258, Apr. 2011.

[181] W. Hong, T. S. Chen, and H. Y. Wu, "An improved reversible data hiding in encrypted images using side match," *IEEE Signal Processing Letters*, vol. 19, no. 4, pp. 199–202, Apr. 2012.

[182] X. Liao and C. Shu, "Reversible data hiding in encrypted images based on absolute mean difference of multiple neighboring pixels," *Journal of Visual Communication and Image Representation*, vol. 28, pp. 21–27, Apr. 2015.

[183] J. Zhou, W. Sun, L. Dong, X. Liu, O. C. Au, and Y. Y. Tang, "Secure reversible image data hiding over encrypted domain via key modulation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 3, pp. 441–452, Mar. 2016.

[184] C. Qin and X. Zhang, "Effective reversible data hiding in encrypted image with privacy protection for image content," *Journal of Visual Communication and Image Representation*, vol. 31, pp. 154–164, Aug. 2015.

[185] Z. Qian, S. Dai, F. Jiang, and X. Zhang, "Improved joint reversible data hiding in encrypted images," *Journal of Visual Communication and Image Representation*, vol. 40, pp. 732–738, Oct. 2016.

[186] R. Bhardwaj and A. Aggarwal, "An improved block based joint reversible data hiding in encrypted images by symmetric cryptosystem," *Pattern Recognition Letters*, Feb. 2018.

[187] X. Zhang, "Separable reversible data hiding in encrypted image," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 826–832, Apr. 2012.

[188] M. Johnson, P. Ishwar, V. Prabhakaran, D. Schonberg, and K. Ramchandran, "On compressing encrypted data," *IEEE Transactions on Signal Processing*, vol. 52, no. 10, pp. 2992–3006, Oct. 2004.

[189] R. Lazzeretti and M. Barni, "Lossless compression of encrypted grey-level and color images," in *Proceedings of the European Signal Processing Conference (EUSIPCO)*, Lausanne, Switzerland, Aug. 2008, pp. 1–5.

[190] D. Schonberg, S. C. Draper, C. Yeo, and K. Ramchandran, "Toward compression of encrypted images and video sequences," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 4, pp. 749–762, Dec. 2008.

[191] W. Liu, W. Zeng, L. Dong, and Q. Yao, "Efficient compression of encrypted grayscale images," *IEEE Transactions on Image Processing*, vol. 19, no. 4, pp. 1097–1102, Apr. 2010.

[192] D. Klinc, C. Hazay, A. Jagmohan, H. Krawczyk, and T. Rabin, "On compression of data encrypted with block ciphers," *IEEE Transactions on Information Theory*, vol. 58, no. 11, pp. 6989–7001, Nov. 2012.

[193] J. Zhou, O. C. Au, G. Zhai, Y. Y. Tang, and X. Liu, "Scalable compression of stream cipher encrypted images through context-adaptive sampling," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 11, pp. 1857–1868, Nov. 2014.

[194] Z. Qian and X. Zhang, "Reversible data hiding in encrypted images with distributed source encoding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 4, pp. 636–646, Apr. 2016.

[195] Z. Qian, X. Zhang, and G. Feng, "Reversible data hiding in encrypted images based on progressive recovery," *IEEE Signal Processing Letters*, vol. 23, no. 11, pp. 1672–1676, Nov. 2016.

[196] Z. Qian, H. Zhou, X. Zhang, and W. Zhang, "Separable reversible data hiding in encrypted JPEG bitstreams," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 6, pp. 1055–1067, Nov. 2018.

[197] K. Ma, W. Zhang, X. Zhao, N. Yu, and F. Li, "Reversible data hiding in encrypted images by reserving room before encryption," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 3, pp. 553–562, Mar. 2013.

[198] X. Cao, L. Du, X. Wei, D. Meng, and X. Guo, "High capacity reversible data hiding in encrypted images by patch-level sparse representation," *IEEE Transactions on Cybernetics*, vol. 46, no. 5, pp. 1132–1143, May 2016.

[199] K. Chen and C.-C. Chang, "High-capacity reversible data hiding in encrypted images based on extended run-length coding and block-based msb plane rearrangement," *Journal of Visual Communication and Image Representation*, vol. 58, pp. 334–344, Jan. 2019.

[200] H.-T. Wu, Y.-M. Cheung, and J. Huang, "Reversible data hiding in Paillier cryptosystem," *Journal of Visual Communication and Image Representation*, vol. 40, pt. B, pp. 765–771, Oct. 2016.

[201] X. Wu, B. Chen, and J. Weng, "Reversible data hiding for encrypted signals by homomorphic encryption and signal energy transfer," *Journal of Visual Communication and Image Representation*, vol. 41, pp. 58–64, Nov. 2016.

[202] M. Li and Y. Li, "Histogram shifting in encrypted images with public key cryptosystem for reversible data hiding," *Signal Processing*, vol. 130, pp. 190–196, Jan. 2017.

[203] B. Chen, X. Wu, and Y.-S. Wei, "Reversible data hiding in encrypted images with private-key homomorphism and public-key homomorphism," *Journal of Visual Communication and Image Representation*, vol. 57, pp. 272–282, Nov. 2018.

[204] C.-W. Shiu, Y.-C. Chen, and W. Hong, "Encrypted image-based reversible data hiding with public key cryptography from difference expansion," *Signal Processing: Image Communication*, vol. 39, pp. 226–233, Nov. 2015.

[205] X. Zhang, J. Long, Z. Wang, and H. Cheng, "Lossless and reversible data hiding in encrypted images with public-key cryptography," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 9, pp. 1622–1631, Sept. 2016.

[206] S. Xiang and X. Luo, "Reversible data hiding in homomorphic encrypted domain by mirroring ciphertext group," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 11, pp. 3099–3110, Nov. 2018.

[207] C.-C. Chang and S.-M. Tsu, "Arithmetic operations on encrypted data," *International Journal of Computer Mathematics*, vol. 56, no. 1-2, pp. 1–10, Apr. 1995.

[208] G. H. Hardy and E. M. Wright, *An Introduction to the Theory of Numbers*, 6th ed., R. Heath-Brown *et al.*, Eds. Oxford, UK: Oxford University Press, 2008.

[209] I. C. Dragoi, H. G. Coanda, and D. Coltuc, "Improved reversible data hiding in encrypted images based on reserving room after encryption and pixel prediction," in *Proceedings of the European Signal Processing Conference (EUSIPCO)*, Kos, Greece, Aug. 2017, pp. 2186–2190.

[210] C. W. Honsinger, P. W. Jones, M. Rabbani, and J. C. Stoffel, "Lossless recovery of an original image containing embedded data," U.S. Patent 6 278 791, Aug. 21, 2001.

[211] Y. Yang, X. Sun, H. Yang, C.-T. Li, and R. Xiao, "A contrast-sensitive reversible visible image watermarking technique," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 5, pp. 656–667, May 2009.

[212] W.-L. Tai, C.-M. Yeh, and C.-C. Chang, "Reversible data hiding based on histogram modification of pixel differences," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 6, pp. 906–910, June 2009.

[213] Z. Qian, X. Zhang, Y. Ren, and G. Feng, "Block cipher based separable reversible data hiding in encrypted images," *Multimedia Tools and Applications*, vol. 75, no. 21, pp. 13 749–13 763, Nov. 2016.

[214] X. Zhang, Z. Qian, G. Feng, and Y. Ren, "Efficient reversible data hiding in encrypted images," *Journal of Visual Communication and Image Representation*, vol. 25, no. 2, pp. 322–328, Feb. 2014.

[215] F. Huang, J. Huang, and Y.-Q. Shi, "New framework for reversible data hiding in encrypted domain," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 12, pp. 2777–2789, Dec. 2016.

[216] W. Zhang, K. Ma, and N. Yu, "Reversibility improved data hiding in encrypted images," *Signal Processing*, vol. 94, pp. 118–127, Jan. 2014.

[217] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1, pp. 259–268, Nov. 1992.

[218] P. Bas, T. Filler, and T. Pevný, "Break our steganographic system: The ins and outs of organizing BOSS," in *Proceedings of the International Workshop on Information Hiding (IH)*, Prague, Czech Republic, May 2011, pp. 59–70.

[219] X. Wu and W. Sun, "High-capacity reversible data hiding in encrypted images by prediction error," *Signal Processing*, vol. 104, pp. 387–400, Nov. 2014.

[220] T. Wu, F. Wu, J. Redouté, and M. R. Yuce, "An autonomous wireless body area network implementation towards iot connected healthcare applications," *IEEE Access*, vol. 5, pp. 11 413–11 422, June 2017.

[221] F. Sebbak and F. Benhammadi, "Majority-consensus fusion approach for elderly IoT-based healthcare applications," *Annals of Telecommunications*, vol. 72, no. 3, pp. 157–171, Apr. 2017.

[222] U. Satija, B. Ramkumar, and M. Sabarimalai Manikandan, "Real-time signal quality-aware ECG telemetry system for iot-based health care monitoring," *IEEE Internet of Things Journal*, vol. 4, no. 3, pp. 815–823, June 2017.

[223] H. Kikuchi, S. Hotta, K. Abe, and S. Nakanishi, "Distributed auction servers resolving winner and winning bid without revealing privacy of bids," in *Proceedings of the International Conference on Parallel and Distributed Systems (ICPADS)*, Iwate, Japan, July 2000, pp. 307–312.

[224] M. Abe and K. Suzuki, "Receipt-free sealed-bid auction," in *Proceedings of the International Conference on Information Security (ISC)*, Sao Paulo, Brazil, Sept. 2002, pp. 191–199.

[225] K. Peng, C. Boyd, E. Dawson, and K. Viswanathan, "Robust, privacy protecting and publicly verifiable sealed-bid auction," in *Proceedings of the International Conference on Information and Communications Security (ICICS)*, Singapore, Dec. 2002, pp. 147–159.

[226] K. Suzuki and M. Yokoo, "Secure combinatorial auctions by dynamic programming with polynomial secret sharing," in *Proceedings of the International Conference on Financial Cryptography (FC)*, Southampton, Bermuda, Mar. 2003, pp. 44–56.

[227] J. D. C. Benaloh, "Verifiable secret-ballot elections," Ph.D. dissertation, Yale University, New Haven, CT, USA, 1987.

[228] R. Cramer, M. Franklin, B. Schoenmakers, and M. Yung, "Multi-authority secret-ballot elections with linear work," in *Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT)*, Saragossa, Spain, May 1996, pp. 72–83.

[229] B. Schoenmakers, "A simple publicly verifiable secret sharing scheme and its application to electronic voting," in *Proceedings of the Annual International Cryptology Conference (CRYPTO)*, Santa Barbara, CA, USA, Aug. 1999, pp. 148–164.

[230] M. Hirt and K. Sako, "Efficient receipt-free voting based on homomorphic encryption," in *Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT)*, Bruges, Belgium, May 2000, pp. 539–556.

[231] C. A. Neff, "A verifiable secret shuffle and its application to e-voting," in *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, Philadelphia, PA, USA, Nov. 2001, pp. 116–125.

[232] S. Iftene, "General secret sharing based on the Chinese remainder theorem with applications in e-voting," *Electronic Notes in Theoretical Computer Science*, vol. 186, pp. 67–84, July 2007.

[233] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979.

[234] G. R. Blakley, "Safeguarding cryptographic keys," in *Proceedings of the AFIPS National Computer Conference (NCC)*, New York, NY, USA, June 1979, pp. 313–317.

[235] M. Ito, A. Saito, and T. Nishizeki, "Secret sharing scheme realizing general access structure," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, Tokyo, Japan, Nov. 1987, pp. 99–102.

[236] J. Benaloh and J. Leichter, "Generalized secret sharing and monotone functions," in *Proceedings of the Conference on the Theory and Application of Cryptography (CRYPTO)*, Santa Barbara, CA, USA, Aug. 1988, pp. 27–35.

[237] E. F. Brickell, "Some ideal secret sharing schemes," in *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques (EUROCRYPT)*, Houthalen, Belgium, Apr. 1989, pp. 468–475.

[238] E. F. Brickell and D. M. Davenport, "On the classification of ideal secret sharing schemes," *Journal of Cryptology*, vol. 4, no. 2, pp. 123–134, Jan. 1991.

[239] A. Beimel and B. Chor, "Universally ideal secret-sharing schemes," *IEEE Transactions on Information Theory*, vol. 40, no. 3, pp. 786–794, May 1994.

[240] B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch, "Verifiable secret sharing and achieving simultaneity in the presence of faults," in *Proceedings of the Annual Symposium on Foundations of Computer Science (SFCS)*, Portland, OR, USA, Oct. 1985, pp. 383–395.

[241] P. Feldman, "A practical scheme for non-interactive verifiable secret sharing," in *Proceedings of the Annual Symposium on Foundations of Computer Science (SFCS)*, Los Angeles, CA, USA, Oct. 1987, pp. 427–438.

[242] T. Rabin and M. Ben-Or, "Verifiable secret sharing and multiparty protocols with honest majority," in *Proceedings of the Annual ACM Symposium on Theory of Computing (STOC)*, Seattle, WA, USA, May 1989, pp. 73–85.

[243] M. Tompa and H. Woll, "How to share a secret with cheaters," *Journal of Cryptology*, vol. 1, no. 3, pp. 133–138, Oct. 1989.

[244] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Proceedings of the Annual International Cryptology Conference (CRYPTO)*, Santa Barbara, CA, USA, Aug. 1991, pp. 129–140.

[245] M. Stadler, "Publicly verifiable secret sharing," in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, Saragossa, Spain, May 1996, pp. 190–199.

[246] R. Cramer, I. Damgård, and U. Maurer, "General secure multi-party computation from any linear secret-sharing scheme," in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, Bruges, Belgium, May 2000, pp. 316–334.

[247] M. Naor and A. Shamir, "Visual cryptography," in *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques (EUROCRYPT)*, Perugia, Italy, May 1994, pp. 1–12.

[248] C. Blundo, A. D. Santis, and M. Naor, "Visual cryptography for grey level images," *Information Processing Letters*, vol. 75, no. 6, pp. 255–259, Nov. 2000.

[249] Y.-C. Hou, "Visual cryptography for color images," *Pattern Recognition*, vol. 36, no. 7, pp. 1619–1629, July 2003.

[250] Z. Zhou, G. R. Arce, and G. D. Crescenzo, "Halftone visual cryptography," *IEEE Transactions on Image Processing*, vol. 15, no. 8, pp. 2441–2453, July 2006.

[251] L. Csanky, "Fast parallel matrix inversion algorithms," in *Proceedings of the Annual Symposium on Foundations of Computer Science (SFCS)*, Berkeley, CA, USA, Oct. 1975, pp. 11–12.

[252] V. Strassen, "Gaussian elimination is not optimal," *Numerische Mathematik*, vol. 13, no. 4, pp. 354–356, Aug. 1969.

[253] D. Coppersmith and S. Winograd, "Matrix multiplication via arithmetic progressions," *Journal of Symbolic Computation*, vol. 9, no. 3, pp. 251–280, Mar. 1990.

[254] F. Le Gall, "Powers of tensors and fast matrix multiplication," in *Proceedings of the International Symposium on Symbolic and Algebraic Computation (ISSAC)*, Kobe, Japan, July 2014, pp. 296–303.