The University Of Sheffield.

Graph Spectral Domain Data Hiding

Hiba AL-Khafaji

Supervisor:

Dr. Charith Abhayaratne

Thesis submitted in candidature for graduating with a degree of doctor of philosophy from the University of Sheffield, Faculty of Engineering, Department of Electronic and Electrical Engineering, July 1, 2020.

Abstract

Recent years have witnessed an increase in applications such as social, transportation, and sensor networks. The authentication and protection of these networks' data have become a major concern. Since these data are spread at arbitrary positions, without following a Cartesian grid, the techniques of classical signal processing cannot be applied to these data. This thesis explores the recently advanced signal processing of graphs for spread spectrum data hiding to protect and authenticate data captured via these networks. In this research, we first explore the graph Fourier domain for data hiding. Our proposed method involves two models for reducing the embedding distortion in the host graph that results from hiding the secret data and for enhancing the robustness of the embedded data against attacks namely, noise addition and deletion of random nodes data. We consider two data hiding scenarios: non-blind and blind. The experimental results demonstrate that the proposed methods have reduced the distortion using MSE by an average of 94% and 80% for non-blind and blind algorithms, respectively. In addition, the robustness of the proposed method is enhanced using the Hamming Distance (HD) by an average of 93% and 99.8% for non-blind algorithm and by an average of 60% and 71% for blind algorithm after the additive noise and deleting nodes data, respectively. The second contribution focuses on proposing a new approach for reversible data hiding for unstructured data in the graph Fourier domain. The proposed methodology includes a model to reduce embedding distortion based on establishing the relationship between the value of the embedded bits and the MSE of the modified graph; our methodology includes another model to maximise the robustness of the embedded bits against the additive noise. The experimental results demonstrate that the proposed method outperforms the previous methods by an average of 87% and 92% in terms of the embedding distortion, and by an average of 54% and 86% in terms of the robustness against the additive noise, and by an average 97% and 99% in terms of reversibility of the original graph signal compared to the previous methods, respectively. The third contribution involves exploiting the Graph Wavelet Transform (GWT) properties for graph data hiding. We explore the graph wavelet transform for proposing data hiding methods, including irreversible and reversible data hiding, with new models that minimise distortion in the host

graph (resulting from hiding the secret bits) and enhance robustness against attacks. The experimental simulations show that the proposed GWT data hiding method outperforms the original data hiding methods (without using the proposed models) by an average of 99% and 99.4% for non-blind and blind data hiding, respectively in terms of embedding distortion. The robustness of the GWT data hiding algorithms are enhanced by an average of 77%, 71%, 60% and 99% for non-blind and blind algorithms after the additive noise and deleting nodes data, respectively. Similarly, the proposed GWT reversible data hiding method has achieved better performance compared to the previous methods by an average of 68%, 82%, 78%, 92%, 95% and 99% in terms of the embedding distortion, robustness against additive noise and reversibility of the original signal, respectively.

Contents

List of Figures

List of Tables

List of symbols

$\zeta()$	Embedding function
$\xi()$	Extraction function
$\oplus$	XOR operation
λ	Graph Eigenvalue
ℓ	Index in the spectral domain
α	Data hiding parameter
μ	Mean Square Error
η	Sub-band
γ	Decomposition level
σ^2	Noise variance
$\mathcal{L}$	The normalised graph Laplacian matrix
Δ_a	Modification value due to attack
$\Delta_{a_{max}}$	Maximum modifications value
$\Delta_{a_{min}}$	Minimum modifications value
$\Delta\mathbf{x}$	Error power in the input signal
$\Delta\mathbf{X}$	Error power in the graph Fourier coefficients
Δ_s	Modification value due to shifting process
δ	Average distortion due to embedding process
Δ_T	Total modification value
$\Delta\mathbf{Y}$	Error power in the graph wavelet coefficients
$\Re$	Natural number
$\mathbb{R}$	Real number
$\mathbf{A}$	Adjacency matrix
b	Embedded secret bit
b'	Extracted secret bit
C	Embedding capacity
c_1, c_2	Orthonormality correction factor
$\mathbf{D}$	Degree matrix

D_E	Total distortion due to embedding process
D_R	Total distortion due to shifting process
D_T	Total distortion
dB	Decibels
$\mathcal{E}$	Graph edges
$\mathcal{G}$	Original graph
$\mathcal{G}'$	Modified graph
G	Synthesis filter
HD	Hamming distance
H	Analysis filter
$h(Y_{Max})$	Peak point in GWT
$h(Y_{Min})$	Zero point in GWT
$h(X_{Max})$	Peak point in GFT
$h(X_{Min})$	Zero point in GFT
I	Identity matrix
J	Anti-diagonal matrix
i, j, i_1	Index
K	Length of the secret bits
$\mathbf{L}$	The combinatorial Graph Laplacian matrix
M	The number of modified coefficients
max	Maximum node value
m	Index
N	Number of graph nodes
n	Decomposition levels
R	Weighting parameter
S	Correlation Similarity
$\tilde{S}_d$	Downsampling matrix
$\tilde{S}_u$	Upsampling matrix
T	Threshold

t	Transpose operator
$\mathbf{U}$	Graph Eigenvector
$\mathbf{u}$	Individual graph Eigenvector
$\mathcal{V}$	Graph vertices
w	Secret bits
w'	Extracted secret bits
$\mathbf{x}$	Graph signal
q	Shifting bin value
$\mathbf{X}$	Graph Fourier coefficients
x_d	Downsampled signal in vertex domain
X_d	Downsampled signal in spectral domain
$\mathbf{X}_w$	Modified graph Fourier coefficients
$\mathbf{X}_s$	Sorted graph Fourier coefficients
$\mathbf{Y}$	Graph wavelet coefficients
$\mathbf{Y}_L$	Low-frequency wavelet coefficients
$\mathbf{Y}_H$	High-frequency wavelet coefficients
$\mathbf{Y}_w$	Modified graph wavelet coefficients
$\mathbf{Y}_s$	Sorted graph wavelet coefficients

List of abbreviations

3D	Three Dimensional
ATM	Automated Teller Machines
BER	Bit Error Rate
CS	Correlation Similarity
CDF	Cohen Daubechies Feauveau
DSA	Digital Signature Algorithm
DFT	Discrete Fourier Transform
DWT	Discrete Wavelet Transform
DCT	Discrete Cosine Transform
DE	Difference Expansion
GWT	Graph Wavelet Transform
GSP	Graph Signal Processing
GFT	Graph Fourier Transform
HD	Hamming Distance
HS	Histogram Shifting
IGFT	Inverse Graph Fourier Transform
IGWT	Inverse Graph Wavelet Transform
IRDH	IRreversible Data Hiding
LC	Lossless Compression
LoDs	Levels of Details
MSE	Mean Square Error
PSNR	Peak Signal to Noise Ratio
PE	Prediction Error
QIM	Quantisation Index Modulation
RDH	Reversible Data Hiding
RSA	Rivest Shamir Adleman
RMSE	Root Mean Square Error

Declaration

This thesis is the result of my own work, ideas, experiments and has not previously been submitted or accepted for any degree other than Doctor of Philosophy of the University of Sheffield. This thesis includes materials, which have been appeared in a published journal and conference papers.

Acknowledgements

I would like to take this opportunity to express my deep gratitude and respect to my great supervisor, Dr Charith Abhayaratne, for his patience, motivation, enthusiasm and immense knowledge. I am especially indebted to my supervisor for sharing his pearls of wisdom and experience during my doctoral studies. I could not have imagined having a better advisor and mentor for my PhD. study. I feel lucky to have him as my supervisor who helped me learn not only the technical aspects, but also inspired me to become an independent researcher and helped me realise the power of critical reasoning. It also demonstrated what a brilliant and hard-working scientist can accomplish. This journey would not have been possible without his support, thank him for encouraging me in my activities and inspiring me to follow my dreams. It was a great privilege and honour to work and study under his guidance. I am extremely grateful for what he has offered me.

Also, I am sincerely grateful to my parents to motivate and encourage me for this long journey, and my family: my husband and my children, Ali, Hasanain, and Zainab to support and help me.

Finally, special thanks to the Ministry of Higher Education and Scientific Research (Iraq) for providing a scholarship for my PhD study.

Chapter 1

Introduction

Due to the ease of illegal data copying, tampering, and distribution, data security has become a matter of significant concern. Encryption and data hiding are considered to be essential for secure communication; encryption serves to encipher a message to be unreadable to all except the receiver, who is able to decipher it, while data hiding aims to hide the existence of secret bits in host media in a secure, robust and imperceptible manner. Therefore, data hiding is generally considered to be the ideal technique for secure communication in many applications due to its ability of hiding the secret data in the host in an imperceptible way [5].

There are two categories of data hiding techniques based on their applications: irreversible data hiding (IRDH) and reversible data hiding (RDH). Irreversible data hiding approaches are once more divided into two types: steganography and digital watermarking. Steganography allows for covert communication by embedding messages in host media (in most of the cases except the case of cover-less steganography) [5]; digital watermarking enables to embed the watermark in digital host media for many applications such as digital rights management [5]. In both of these irreversible data hiding approaches, the embedding of secret bits distorts the host media, meaning the original host signal cannot be restored once the bits are removed. Since this distortion is unacceptable in many circumstances, such as medical and military applications, reversible data hiding is used on account of its ability to restore the original host data without errors following the extraction of the hidden bits.

1.1 Motivation

Recent years have seen a growth of using various sensors to sense and measure various data. As these sensors are located at arbitrary locations, without following a Cartesian grid, the data recorded using a network of sensors can be represented in a graph, with vertices (nodes) representing the locations of sensors. The connectivity between nodes can be defined by considering the relationship among sensors. This thesis concerns the protection and authentications of data captured via data networks. However, the ability to address the protection of irregular data structures is still limited. The most popular methods to protect and authenticate graph data are inserting more edges between nodes [6], adding extra vertices [7] and embedding subgraphs [8]. For clarifying the real world applications of the proposed work, we have considered two scenarios, social networks and sensor networks.

We have seen an increase in sensitive data which are captured in large graphs in recent years. These data can involve maps of autonomous systems in the Internet, social networks which represent a lot of friendships, or records of patent citations. The main challenge is to control access to these data. To be specific, it is often that the owners of the graph data need to share access to them for a specific set of entities without sharing these data with the public domain. For instance, large social networks such as Facebook or LinkedIn may require to share portions of its sensitive data with trusted academic colleagues, but want to prevent their leakage into the broader research community. One option is to build strong mechanisms to access control for preventing data leakage beyond authorized parties but the owners of the data cannot restrict physical access to the data, and have limited control once the data are shared with the trusted cooperators. The best option is to embed a sequence of secret bits to the graph data in an imperceptible way and difficult to remove for the purpose of protecting and authenticating the graph data. For example, in Intellectual Property Protection (IPP), a sequence of secret bits are embedded in the graph data which represent the author's signature. Another example, for limiting data piracy by music vendors such as Apple and Walmart, user's personal information is embedded into a music file at the time of purchase/download. The same concept is applied in graph data hiding based on identifying a copy of a graph with its authorized user. In the case of leaking the shared graph data, the owner of this data can extract the secret bits from

the leaked data and use it as proof to seek damages against the cooperator who is responsible for that. Another scenario is sensor networks. A group of sensors are located at arbitrary locations for sensing and measuring various data. Sensor networks are used in many applications such as military surveillance, environment monitoring and healthcare. For example, in military applications, various sensors are spread for battlefield surveillance, where intelligence and Surveillance are important sources of information required for the military operations. Information shared through military networks is very critical and must be confidential because leakage of such information can lead to security issues. So there is a need to authenticate the data of these sensors. Moreover, sensors installed in insecure environments make them reveal various security risks such as eavesdropping, signal distortions and spoofing. For such sensitive applications, security is considered one of the main concerns. Another application for sensor networks is environment monitoring to measure environmental conditions such as temperature, sound, pollution levels, humidity, wind, and so on. For instance, various sensors are used to measure and monitor the temperature in several cities. These sensors can be used to build temperature monitoring systems which can monitor the temperature in real time to prevent fire and other accidents. So maintaining confidentiality of these sensitive data (without tampering or modifying them) is an important aspect. Since these works are based on vertex domain, they are not robust to data processing, noise removal or geometrical attacks, such as, adding new nodes, edges or sub-graphs. Another important aspect is the computational complexity of the existing work. As these works are based on graph colouring in other words they need to change the nodes colours after embedding the secret bits, the computational complexity cost of these works is high. In addition, these methods are not secure, if the original graph is available the watermark can be detected by comparing the two graph topologies [8]. Finally, the embedding capacity is very small due to the embedding process is depended on the graph topology, not the correlation of its data.

On the other hand , data hiding using a spread spectrum has been proven to be a successful method for image and video protection, largely due to developments in signal transforms [9–22]. However, we cannot apply classical transforms, such as the discrete Fourier transform (DFT) [23] and the discrete wavelet transform (DWT) [24], to graph data where the vertices are located at arbitrary positions, as opposed to sampling on regular structures in images

and other signals. Spread spectrum data hiding is one of the most secure techniques of data hiding because the secret data are spread over many frequency bands so that the energy in one band is undetectable. This thesis proposes a spread spectrum data hiding method for graph data that takes advantage of recent developments in signal processing of graph on spectral decomposition of the Laplacian matrix of graph that captures the node connectivity [25, 26]. Another aspect is the graph colouring problem is a well-known hard problem, therefore using the graph spectral domain is considered an alternative method of data hiding with a low computational complexity cost. The computational complexity cost of the proposed methods using graph Fourier transform is $O(N^2)$ for full eigendecomposition. The computational complexity cost is reduced by using the polynomial function. For the proposed methods using graph wavelet transform, computational complexity cost is $O(p|\mathcal{E}|)$, where p is the degree of polynomial function.

The main research question of this work is to explore the graph spectral domain for unstructured data hiding. We consider two graph spectral domains: graph Fourier and graph wavelet for both irreversible and reversible data hiding. Our findings suggest that graph spectral domain data hiding successfully protects unstructured data; additionally, embedding in the spectral domain results in minimal embedding distortion and high robustness against attacks.

1.2 Key contributions

The main contributions of this thesis are as follows:

1. **The proposal of a new irreversible data hiding method for graph data in the graph Fourier domain.**

   We propose a new irreversible data hiding approach for unstructured graph data in the graph Fourier domain. **Chapter 3** explores advancements in graph signal processing (GSP) for spread-spectrum data hiding for unstructured data. The first contribution of this thesis is to hide the secret bits in the graph Fourier coefficients, which are selected based on two proposed models: the embedding distortion minimisation model and the robustness model. In order to minimise embedding distortion, we identify the relationship between the error distortion and the chosen graph Fourier coefficients to embed the secret bits. In order to improve the robustness of the embedded bits against attacks, we propose

a robustness model based on the relationship between the extraction of the embedded bits and the effect of the attacks, namely, noise addition and nodes data deletion. This work considers two data hiding scenarios: non-blind using a magnitude based multiplicative data hiding method and blind using prediction-based graph data hiding. The empirical results show that embedding distortion is minimised and robustness is enhanced by the proposed models.

2. **The proposal of a new reversible data hiding method for graph data in the graph Fourier domain.**

   We propose a new reversible data hiding algorithm using histogram shifting in **Chapter 4**. We take advantage of developments in graph signal processing to propose a reversible data hiding approach for data recorded on non-Cartesian grids. We embed the secret bits through a slight modification of the graph Fourier coefficients' magnitudes that enables the blind extractor to extract the secret bits and the original coefficients without distortion. In order to minimise the embedding distortion, we establish the relationship between the error distortion metric and the value of the embedded data. In order to improve robustness against attacks, we propose a robustness model that identifies the relationship between data extraction and the effect of noise addition. Our experimental evaluation verifies the successful recovery of the original graph signal and the secret bits without any errors with high embedding rates. Using a graph spectral domain, our empirical results demonstrate the superiority of the proposed method over the existing reversible data hiding methods in terms of embedding distortion, original signal reversibility and robustness against additive noise.

3. **The proposal of new data hiding methods for graph data in the graph wavelet domain.**

   Discrete wavelet transform is considered a typical option for data hiding in multimedia due to its ability to represent data in both time and frequency domain and with multi-resolution decomposition. We propose new irreversible and reversible data hiding methods using graph wavelet transform in **Chapter 5**. The proposed methodology includes new models: the embedding distortion minimisation model and the robustness model.

Our experimental results demonstrate that the proposed data hiding methods in the graph wavelet domain are superior to those in the graph Fourier domain.

1.3 Publications

Part of the work in this thesis has been published in the following conference papers.

1. H. Al-khafaji and C. Abhayaratne, "Graph Spectral Domain Watermarking for Unstructured Data from Sensor Networks," Proc. of International Conference on Digital Signal Processing (DSP), IEEE, 2017, pp. 1–5.

2. H. Al-khafaji and C. Abhayaratne, "Graph Spectral Domain Blind Watermarking," Proc. of International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019, pp. 2492–2496.

Also, the following papers are being prepared currently to publish as journal articles:

1. H. Al-khafaji and C. Abhayaratne, " Graph Spectral Domain Watermarking for Graph Data," to be published as journal article.

2. H. Al-khafaji and C. Abhayaratne, "Graph Spectral Domain Reversible Data Hiding for Graph Data," to be published as journal article.

3. H. Al-khafaji and C. Abhayaratne, "Graph Wavelet Domain Watermarking for Graph Data," to be published as journal article.

1.4 Outline

The rest of the thesis include the following five chapters:

Chapter 2 serves as an overview of data hiding. Section 2.1 offers insight into the various data hiding techniques and their applications. Previous works on irreversible and reversible data hiding are reviewed in Section 2.2 and Section 2.3, respectively. Section 2.4 illustrates the fundamental concepts of graph theory and graph spectral theory while Section 2.5 and Section 2.6 review the existing work on irreversible and reversible data hiding in the graph spectral domain, respectively. Finally, Section 2.7 serves to summarise the chapter.

Chapter 3 proposes a novel method for irreversible data hiding in the graph Fourier domain. Section 3.1 introduces the proposed method while Section 3.2 further details the proposed models, those being the embedding distortion minimisation model and the robustness model. Section 3.3 evaluates the performance of the proposed method. The chapter is then summarised in Section 3.4.

Chapter 4 proposes a new method for reversible data hiding in the graph Fourier domain. Section 4.1 introduces the proposed method while Section 4.2 discusses in more depth the proposed models, those being the embedding distortion minimisation model and the robustness model. Section 4.3 evaluates the performance of the proposed method and Section 4.4 serves to conclude the chapter.

Chapter 5 proposes new methods for both irreversible and reversible data hiding using graph wavelet transform. Section 5.1 introduces the proposed data hiding methods while Section 5.2 details the two new models, those being the embedding distortion minimisation model and the robustness model. Section 5.3 evaluates the performance of these proposed methods and, finally, Section 5.4 presents a summary of the chapter.

Chapter 6 concludes this thesis by summarising the research results and describing the future directions for data hiding in graph data.

Chapter 2

Background and related work

This chapter serves as an overview of data hiding techniques and their applications, graph theory, graph spectral theory and some relevant works on irreversible and reversible data hiding.

In this thesis, we explore the use of recent advancements in graph signal processing to protect data recorded on non-Cartesian grids, such as sensor data and 3D point clouds. This chapter includes seven sections. First, Section 2.1 provides preliminary information about data hiding, its various techniques and their applications. Section 2.2 and Section 2.3 review the existing work on irreversible and reversible data hiding, respectively. Section 2.4 illustrates the concepts of graph theory and graph spectral theory. Previous works on irreversible and reversible data hiding in the graph spectral domain are then reviewed in Section 2.5 and Section 2.6, respectively. Finally, Section 2.7 offers concluding remarks.

2.1 Data hiding techniques

Rapid development in digital communication technology has led to the ability to easily copy and manipulate digital media. Powerful software paired with modern devices, such as digital cameras, scanners and MP3 players, enables users to generate and tamper with data. The internet and wireless networks also simplify the process of changing and transmitting data. This create a motivation to find techniques that enhance the security of digital media against threats. The main techniques for secure communication are encryption and data hiding. Through encryption, secret data are enciphered using a secret key that makes the data unreadable to all

except the recipient, who knows the secret key. The encryption of the host media makes the media meaningless, so the third person (the attacker) becomes suspicious.

Alternatively, data hiding techniques are used to keep the existence of secret data undetectable, as the majority of them do not degrade the host media [5]. Therefore, data hiding is generally considered to be more confidential than encryption because most of the data hiding techniques hide the existence of secret data rather than just protecting their content [27]. The advantage of the majority of data hiding techniques is that the existence of a secret message is hidden by embedding its data in the host media in a way that is imperceptible, secure and robust against attacks [5, 27]. Data hiding has been utilised in several applications, such as ownership protection, authentication and access control. For instance, spread spectrum modulation is used in military communication to protect the signal from enemy interception. Other applications include healthcare and copyright monitoring [28].

Generally, data hiding techniques are categorised into two types: irreversible and reversible. Through irreversible data hiding, the secret bits are restored without error but the original host media are distorted. Through reversible data hiding, the secret bits and the original host media are both restored without error. The main types of irreversible data hiding techniques are digital watermarking and steganography. Majority of watermarking entails hiding a watermark in host media in an imperceptible and robust manner while steganography is essentially the science of communicating in a manner through which the existence of secret data is undetected. This makes steganography the preferred candidate in many applications, such as intelligence, law enforcement and counter-intelligence agencies.

The main differences are that watermarking must be more robust against various types of attacks and there is a relationship between the hidden data and host media in watermarking; in other words, the host media are more important for the receiver.

Data hiding is an ancient art; there are many stories that reveal it dates back to antiquity. The most famous story about steganography is about Herodotus (486–425 B.C), who hide a secret message by shaving the hair of his slave [29]. Another example is a Greek man who warned his king of an invasion by writing on a piece of wood after removing the wax from a writing tablet, then covering it with wax [29]. In 1870–1871, messages were sent by pigeons on microfilms during the Franco-Prussian War. In 1905, nostrils, ears and fingerprints were

used to hide microscopic images during the Russo-Japanese War. During World War I, various stages of photographic reduction were used to reduce messages to microdots. Additionally, by World War I, developments in chemistry allowed for more sophisticated collections of ink. However, this technology was abandoned following the invention of universal developers that enabled the identification of paper pieces that had been welted. Despite China inventing the art of paper-making over one thousand years earlier, watermarking appeared in Italy in about 1282. By the eighteenth century, paper watermarks were used as trademarks in Europe and America. Interest in watermarking digital media began in the mid-1990s with a focus on audio, images and video [1].

Nowadays, watermarking research has become quite mature due to its many applications in, for example, image processing, telecommunications, computer science, and remote sensing. The majority of existing data hiding work is based on watermarking for its high level of protection; its many applications are illustrated in Table 2.1.

Table 2.1: Applications of digital watermarking [1]

Application	**Description**
Broadcast monitoring	It can be implemented based on inserting a unique watermark in each clip of video in order to identify when and where each clip appears.
Copyright identification	The watermark bits are utilised as copyright data.
Content authentication	To authenticate of the original data and protect them against digital forgery.
Copy control	Watermarking can be used as a strong tool to prevent illegal copying of multimedia.
Packaging and tracking	Inserting watermark on packages in order to track and protect it against forged consumable items including pharmaceutical products.
Banking document authentication	To authenticate the documents of financial like authentication banking.
Temper detection	The watermark bits are embedded in the sensitive data, if the watermark data are distorted this mean the data cannot be trusted.
Fingerprinting	For each copy of data a fingerprint should be added.
Telemedicine	It is the science which is used to solve the health problems.

According to the embedding domain, secret data are hidden in two different domains: spatial and spectral (frequency) as shown in Figure 2.1. The frequency domain allows for better insight into enhancing watermark robustness and decreasing embedding distortion. Digital watermarking is classified by host: image, video, audio, text and graph. From a human perspective,

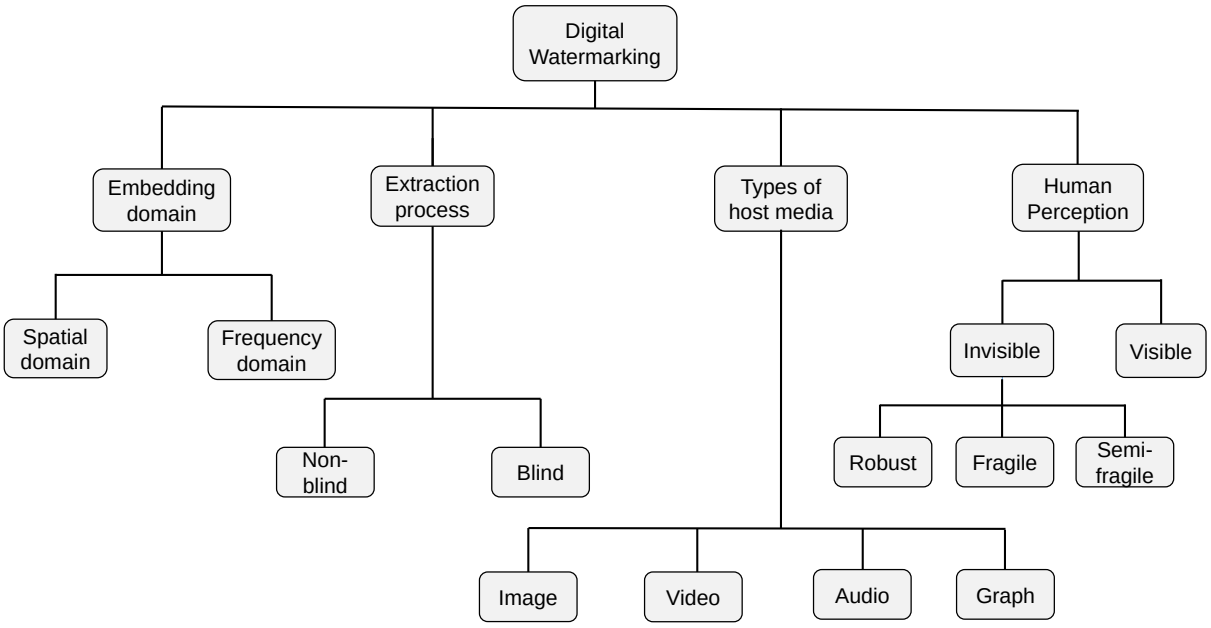


Figure 2.1: Types of digital watermarking [1]

however, the methods are classified as either invisible and visible watermarking. Invisible watermarking methods are divided into robust, fragile and semi-fragile [1]. The watermark bits survive intentional attack using the robust watermarking method; using the fragile method, the watermark bits are destroyed following any modification to or attack on the host media [30]; using the semi-fragile method, the watermark bits survive certain kinds of attacks but are destroyed following other kinds [31].

An attack is defined as any process trying to remove or modify secret data in host media. The attacks are classified into three groups: removal attacks, which aim to damage or remove the secret data, such as noise, histogram equalization, blur and sharpen attacks; geometry attacks, which aim to distort the secret data, such as rotation and translation; malicious attacks aim to remove the secret data by manipulating the modified host media; protocol attacks, which aim to add their own secret data to the data in question, such as invertible and copied attack [32].

Irreversible data hiding techniques, i.e., digital watermarking and steganography distort the host media when hiding secret bits, meaning that the host media cannot be restored following extraction [5]. This distortion is unacceptable in many applications, such as military intelligence, as the original host media is also important [33]. Therefore, reversible data hiding techniques are employed in these applications. Reversible data hiding is a technique that enables the

blind extractor to recover both original data and hidden data without error [33]. Reversible data hiding research dates back several decades but early works have problems. In [34], Mintzer et al. display a form of visible reversible data hiding for applications such as on-line content distribution. The image is marked with a reversible visible watermark before distribution or posting on the Internet, and the watermarked image content serves as a teaser that users may view or obtain for free. Then, the watermark can be removed to recreate the unmarked image by using a vaccine program that is available for an additional fee. The first invisible reversible watermark was suggested in [35] and involved adding the modulo operation to the existing additive approach. Similar work was proposed in [36] but this approach suffers from visual artefacts. However, another work [37] overcame the artefacts problem in the reversibility process.

2.1.1 Data hiding properties

The common characteristics of the data hiding systems are presented in Table 2.2. The importance of these characteristics depends on the requirements of the application [1].

Table 2.2: Characteristics of data hiding system

Property	**Description**
Robustness	The secret data should be detected after benign signal processing.
Imperceptibility	The secret data be imperceptible and without distorting the host media.
Security	It can be defined as the capability of secret bits to withstand to the malicious attacks.
Capacity	Size of secret data which can be embedded in the host media.

2.1.2 Secret data types

The secret data can be divided into two different categories: pseudo-random sequence and text/image/logo according to the application. The logos are classified into binary logo, gray scale logo and colour logo. We can categorised the pseudo-random sequence into two kinds: Natural number sequence and Binary sequence [1].

2.1.3 Data hiding

We can define data hiding as the process of hiding the secret bits into host media like video, image and audio or graph. Data hiding includes two main stages: embedding and extraction.

2.1.3.1 Embedding

In this stage, the secret data are embedded in the host media (in our work a graph) by using a suitable algorithm. According to the algorithm, the secret bits can be embedded into some or all host data (graph data) in the spatial domain (graph vertex domain) and the frequency coefficients (spectral coefficients in the graph spectral domain) as illustrated in the following equation:

$$\mathcal{G}^{'} = \zeta(\mathcal{G}, \mathbf{w}, \mathbb{k}), \tag{2.1}$$

where $\mathcal{G}$ is the original host (original graph), $\mathbf{w}$ are the secret bits, $\mathbb{k}$ is the embedding key , $\mathcal{G}^{'}$ is the modified host (modified graph) and $\zeta()$ is the embedding function, Figure 2.2 shows the embedding process. We can divide the embedding function in the frequency (spectral) domain

Figure 2.2: Embedding process

into sub-processes: a) Forward transform, b) Coefficients selection, c) Embedding algorithm and d) Inverse transform.

2.1.3.2 Extraction

The embedded bits are extracted from the modified host (modified graph). The extraction procedure includes two sub-processes: extracting the secret bits and the authentication of them. In the extraction process, the secret bits are extracted based on reversing the embedding process. In order to extract the secret data, some extraction algorithms require the original host (graph), this data hiding kind is called a non-blind data hiding. While, in the case of blind data hiding, the embedded bits can extract without requiring the original host (graph). The extraction

process can be defined as follows:

$$w^{'} = \xi(\mathcal{G}^{'}, \mathcal{G}, \Bbbk), \tag{2.2}$$

where $\mathcal{G}^{'}$ is the modified host (graph), $\mathcal{G}$ is the original host (graph), $\Bbbk$ is the extraction key, $\mathbf{w}^{'}$ is the extracted secret bits and ξ is the extraction function. The next step is the authentication process which compares the extracted secret bits with the original secret bits as illustrated in Figure 2.3. For reversible data hiding, the original host data (graph data) should be recovered without any error after the embedded data have been extracted. There are many reversible algorithms which are used to restore the original data without any distortion.

2.1.4 Fidelity Metrics

Many fidelity metrics can be utilised to measure the performance of data hiding methods. These metrics are mainly classified into two types [38]:

2.1.4.1 Embedding performance

The most common metrics which are used to measure the robustness and embedding distortion between the original media and modified media are:

2.1.4.1.1 Mean Square Error (MSE)

The Mean Square Error (MSE) is used to measure the embedding distortion between the original graph and modified graph. The low MSE value means low embedding distortion. The MSE

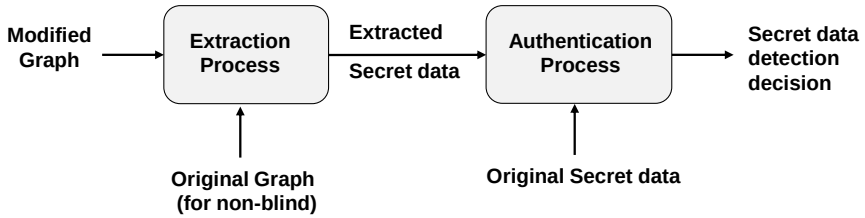


Figure 2.3: Extraction process

value is calculated as given:

$$MSE = \frac{1}{N}\sum_{i=0}^{N-1}(\mathbf{x}_w(i) - \mathbf{x}(i))^2, \tag{2.3}$$

where $\mathbf{x}_w$ is the modified graph, $\mathbf{x}$ is the original graph and N is the number of graph nodes.

2.1.4.1.2 Peak Signal to Noise Ratio (PSNR):

It is considered to be the most popular ways to compare the distortion between two signals such as images and graphs according to the Root Mean Square Error (RMSE) as follows:

$$PSNR = 20\, log_{10}(\frac{max}{RMSE})dB, \tag{2.4}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=0}^{N-1}(\mathbf{x}_w(i) - \mathbf{x}(i))^2}, \tag{2.5}$$

where max is represents the maximum node value of the graph. A high PSNR value means less distortion in the modified graph. In other words, when the PSNR value is high this means the similarity between the two graphs is high.

2.1.4.2 Extraction performance

The most popular metrics can be used to authenticate the extracted secret bits are:

2.1.4.2.1 Bit Error Rate (BER)

The most common way utilised to calculate the difference between the extracted and original secret data for evaluation the robustness is the Hamming Distance (HD) [1]. Hamming Distance is often referred as Bit Error Rate (BER) in communication systems. Hamming distance is widely used for a binary secret bits detection as we considered the pseudo-random binary sequence as the secret bits. When the Hamming Distance is less than a certain threshold, the extracted bits can survive after the attack [1]. The Hamming Distance (HD) can be calculated

as follows:

$$HD(\mathbf{w}, \mathbf{w}^{'}) = \frac{1}{K} \sum_{i=1}^{K} \mathbf{w}_i \oplus \mathbf{w}_i^{'}, \tag{2.6}$$

where $\mathbf{w}$ represents the original secret bits, $\mathbf{w}_{\prime}$ is the extracted secret bit, K is the length of the secret bits vector and $\oplus$ is the XOR operation.

2.1.4.2.2 Correlation Similarity (S)

Correlation Similarity (S) is another measurement, which is utilised to measure the similarity between the original secret bits and extracted secret bits. The secret bits can survive after the attack when the correlation similarity is greater than a certain threshold. This measurement can be computed based on the equation [1]:

$$S(\mathbf{w}, \mathbf{w}^{'}) = \frac{\mathbf{w} \cdot \mathbf{w}^{'}}{\sqrt{\mathbf{w}^{'} \cdot \mathbf{w}^{'}} \sqrt{\mathbf{w} \cdot \mathbf{w}}}. \tag{2.7}$$

2.2 Irreversible data hiding techniques

Different irreversible data hiding methods have been proposed over the years [39–42]. The data hiding methods are basically classified into two categories according to the embedding domain: spatial domain and frequency domain. The substitution technique is the most common type of data hiding in spatial domain. The Least Significant Bits (LSB) method is the most popular method of this technique. The secret bits are embedded by substituting insignificant parts of the host by secret data [43]. The receiver can extract the embedded bits from the host bits based on a secret key. Since the modifications in the host data are assumed to be minor, the embedder expects that these modifications will be unnoticeable. LSB substitution, however, despite its simplicity brings some drawbacks. Although it may survive transformations such as cropping, any addition of noise or lossy compression is likely to defeat the embedding process. Furthermore, once the algorithm is discovered, the embedded information could be easily modified by an intermediate party pseudo-random number generator to determine the pixels to be used for embedding based on a given key [44].

An advantage of the substitution techniques discussed above is that they can be easily ap-

plied to any host media [45] but a disadvantage of substitution techniques is that they are not robust to many types of attacks. The robustness and quality of the hiding process are improved if the properties of the host image could similarly be exploited. For instance, it is generally preferable to hide secret data in noisy regions and edges of images, rather than in smoother regions. The benefit is that the degradation in smoother regions of an image is more noticeable to the Human Visual System (HVS). Instead of the time domain, the transform domain is considered an effective domain for data hiding. Discrete Cosine Transform (DCT) is the most popular domain for image processing. The DCT allows an image to be broken up into different frequency bands, making it much easier to embed secret data into the middle frequency bands of an image. The middle frequency bands are chosen such that they do not affect the most visually important parts of the image (low frequencies) and to be robust against the compression and noise attacks [46]. Another domain is a Discrete Wavelet Transform (DWT) which is considered an optimal choose in the areas of image and video processing such as compression, noise reduction and data hiding [47]; this is attributed to its properties in time-frequency localisation, multi-resolution representation and superior Human Visual System (HVS) modelling [48]. The secret data can be hidden on one sub-band or several sub-bands. One of the many advantages of the wavelet transform is its adaptivity to the HVS as compared to the other transforms, which allow a big embedding capacity in regions that the HVS is known to be less sensitive to, such as the high resolution detail bands. In general, embedding the secret data in the frequency domain will increase the robustness of the secret data against many attacks, and at the same time they remain imperceptible to the human sensory system.

Another common technique of data hiding is Spread Spectrum (SS). It has been developed since the 1950th to avoid interception and anti-jamming communications. In this technique, the signal occupies a bandwidth in excess of the minimum necessary to send the information. The band spread is accomplished by a code which is independent of the data, and a synchronized reception with the code at the receiver is used for despreading and subsequent data recovery [49, 50]. In data hiding, two techniques of SS are used: direct sequence and frequency-hopping schemes. In direct-sequence schemes, the signal is spread by a constant called chip rate, modulated with a pseudo-random signal and added to the host. On the other hand, in frequency-hopping schemes, the frequency of the carrier signal is altered in a way

that it hops rapidly from one frequency to another. Even if parts of the signal are removed in several frequency bands, enough information remains in other bands that enable us to recover the secret data. Spread spectrum techniques are robust against many attacks and can be used in watermarking [51].

2.3 Reversible data hiding techniques

Reversible Data Hiding (RDH) is a branch of data hiding that allows for the accurate extraction of embedded data and the recovery of the original host signal without error. In general, RDH algorithms are classified into three categories. The first category of algorithms follow Lossless Compression embedding framework (LC). In these algorithms, a twin feature is computed for a pair of pixel and compressed. Secret data are embedded in the extra space left by lossless compression. Fridrich et al. [52] propose an approach based on modifying a bit plane of the image. This method had an occurrence of disturbing artifacts due to the varying number of bit planes as the capacity change from image to image. Fridrich's method work strictly for environments where, if a modified image is lossy processed resulting in a bit modification, the bit-plane containing the payload will disturb the entropy synchronization thus losing the hidden data permanently. Vleeschouwer et al. [37] propose a method that overcame Fridrich's method. The method based on using the circular interpretation of bijective transformations. A circle is mapped with the histograms for groups of pixels that was operated by the transform. The relative orientation among the histograms of two groups convey one bit of information. The reversibility process in this method does not experience artifacts and the wrapped pixels are not altered.

The second category of algorithms are based on difference expansion (DE). The differences between two pixels are expanded to get the least significant bits (LSBs) in order to be used for embedding secret data. The first method of difference expansion is proposed by Tian [53]. It is applied on a pair of pixels to develop a low distorted high-capacity reversible watermark. In this method, the image is divided into pixel pairs that do not cause an overflow or underflow. Then a single bit is embedded into the difference of the pixel pair. The payload includes a compressed location map mentioning the modified pairs. A spatial domain reversible data

hiding method is proposed in [54] based on the contrast mapping (RCM). This method has a high-embedding capacity without any secondary compression stage. The space occupied by the LSBs is used for data hiding. Difference expansion transforms are concentrated for improving RDH characteristics [55–57].

The last category of RDH algorithms are based on histogram shifting (HS). The histogram of an image, which is uneven, can be modified for embedding data by considering the bins of the histogram. Ni et al. [3] propose the first RDH algorithm based on slightly modifying the gray scale value of the pixel. This method has a high embedding capacity. A different approach for RDH is suggested in [58] based on histogram modification technique. It uses differences between pixels thus increasing hiding capacity. A binary tree is used to remove the requisite of communication between pairs of zero and peak points to the recipient. Jung et al. [59] propose an algorithm using a data embedding level. It is adjusted for every pixel depending on human visual system characteristics. For reducing the distortion, the embedding level is determined by the estimated values based on an edge and the slightly differential values of each pixel. Zhang et al. [60] generalised the method in [59] using a decompression algorithm. In this method, the secret bits are embedded using a coding scheme. It aims for predefined entropy to be reached using a compression algorithm by reaching the rate distortion bound for the generalised code. It succeeded, using these binary codes, three RDH schemes- one for spatial images, one for JPEG images, and a pattern substitution scheme for binary images that used binary feature sequences as covers are improved. Prediction error expansion (PEE) for RDH is proposed for multiple histograms [61]. Sequences of histograms are used based on multiple histograms modification (MHM) to devise a new embedding mechanism. A prediction error histogram (PEH) is generated based on complexity measurement for each pixel according to its context. The result is minimised embedding distortion, fixed modification manner, and independence of image content. [62] enhances the contrast of a host image thus improving the visual quality keeping PSNR high. This is accomplished by histogram equalization on the highest two bins in the histogram. With this method the original image is completely recoverable.

2.4 Graph spectral theory concepts

This section illustrates the main concepts of spectral graph theory. The graph theory was invented by Leonhard Euler to solve the Königsberg problem. The main aim of the Eulerian graph is to connect three lands with the mainland using seven bridges in a way that crosses each bridge once and only once [63]. Graph signal processing has received significant interest in the past decade due to its advantages in several applications [64]. Many applications involve data defined on complicated domains (non-Euclidean spaces). Examples include data defined on network-like structures, data defined on manifolds or irregularly shaped domains, and data consisting of point clouds, such as collections of feature vectors with associated labels.As many traditional methods for signal processing are designed for data defined on regular Euclidean spaces (data recorded on Cartesian grids) as images and videos, the development of methods that are able to accommodate complicated data domains is an important demand (more details in [25]).

As in classical signal processing, signals can stem from a variety of domains; unlike in classical signal processing, however, the underlying graphs can tell a fair amount about those signals through their structure (more details in [65]). Both a signal on a graph with N vertices and a classical signal with N samples can be viewed as vectors in $\mathbb{R}^N$. However, a major obstacle to the application of the classical signal processing techniques in the graph setting is that processing the graph signal in the same ways as a classical signal processing ignores key dependencies arising from the irregular data domain (more details in [26]), for these reasons, the classical transforms such as KLT and DCT cannot be applied in the graph signal processing.

Graph is a generic data structure that can represent complex relationships among data and can be used in many fields of engineering and science. It consists of nodes and edges, and each edge is usually assigned a weight determined by the similarity of the nodes, e.g., physical or feature space distance between nodes in the network. In graph signal processing, a sample is placed on each node of a graph. Graph signal processing can explicitly consider the structure of the signal, unlike traditional digital signal processing.

2.4.1 Graph spectral theory

With the development of computer technology, there have been considerable interests for analysing or processing irregular and high-dimensional data in many fields, including physical infrastructure network like sensor networks and neural networks. Graph signal processing has been developed to respond to these demands [26].

Graph signal processing is a relatively new field that has been extensively studied since around 2011. It has been an interesting topic in signal and information processing for both theoretical and practical reasons. From a theoretical viewpoint, it is related to signal processing, information theory [66] and computational harmonic analysis [25]. Moreover, from a practical viewpoint, it has been used on an extensive amount of data with irregular structures, e.g., sensor and brain networks [67, 68], traffic [69], learning [70–72], and images [73, 74].

The graph is considered a mathematical model and is used to represent structured data (e.g., regular and irregular data). We can define the graph as a set of nodes (vertices) and links (edges), which represent the connections between the vertices. A directed graph is a graph that is made up of a set of vertices connected by edges, where the edges have a direction associated with them. In an undirected graph, the same edge is used between any two nodes. We call a graph connected if there is a path between every pair of vertices. A disconnected graph is a graph with at least one vertex that is not connected to other vertices but that is otherwise connected.

Let suppose that $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{A}\}$, is an undirected graph without self loops (there is no an edge that connects a vertex to itself) and multiple edges between vertices (only one edge that connects any two vertices), where $\mathcal{V}$ is the set of N vertices as $\mathcal{V} = \{v_0, v_1, ..., v_{N-1}\}$, $\mathcal{E}$ is the set of edges and $\mathbf{A}$ is the adjacency matrix with edge weights. The entry $\mathbf{A}_{i,j}$ represents the weight of the edge, if there is an edge connecting vertices i and j; otherwise $\mathbf{A}_{i,j} = 0$. There are many ways of defining the edge weight (interpreting the relationship between vertices) such as Euclidean distance as shown [75]:

$$\mathbf{A}_{i,j} = \begin{cases} \varepsilon_{i,j}, & \text{if nodes i and j are connected,} \\ 0, & \text{otherwise .} \end{cases} \tag{2.8}$$

where $\varepsilon_{i,j}$ is the Euclidean distance between the nodes i and j. Another way to represent the

the relationship between vertices by considering value equal to one if there is an edge between the vertex i and the vertex j, and zero when there is no edge as follows [76]:

$$\mathbf{A}_{i,j} = \begin{cases} 1, & \text{if nodes i and j are connected,} \\ 0, & \text{otherwise .} \end{cases} \tag{2.9}$$

Another alternative way of representing the edge weight is via a threshold Gaussian kernel weighted function as given [26]:

$$\mathbf{A}_{i,j} = \begin{cases} exp(-\frac{[\varepsilon_{i,j}]^2}{2\theta^2}), & \text{if } \varepsilon_{i,j} \leq \kappa, \\ 0, & \text{otherwise .} \end{cases} \tag{2.10}$$

where θ and κ are some parameters and $\varepsilon_{i,j}$ is the Euclidean distance."

A graph signal is a real-valued scalar function $\mathbf{x} : \mathcal{V} \rightarrow \mathbb{R}$ defined on graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ such that $\mathbf{x}(v)$ is the sample value of function at vertex $v \in \mathcal{V}$. On a finite graph, the graph signal can be viewed as a sequence or a vector $\mathbf{x} = [\mathbf{x}(0), \mathbf{x}(1), ..., \mathbf{x}(N)]^t$, where the order of arrangement of the samples in the vector is arbitrary and neighbourhood information is provided separately by the adjacency matrix $\mathbf{A}$. Graph-signals can, for example, be a set of measured values by sensor network nodes [77] or traffic measurement samples on the edges of an Internet graph [69] or information about the actors in a social network. The combinatorial Laplacian matrix of graph, $\mathbf{L}$, is defined as follows:

$$\mathbf{L} = \mathbf{D} - \mathbf{A}, \tag{2.11}$$

where $\mathbf{D}$ is the diagonal matrix of vertex degrees, whose diagonal components are calculated as given:

$$\mathbf{D}_{(i,i)} = \sum_{j=0}^{N-1} \mathbf{A}_{(i,j)}, \qquad i = 0, 1, ..., N-1. \tag{2.12}$$

Since, $\mathbf{L}$, is a symmetric positive semi-definite matrix, from theorem of spectral projection, there exists a real unitary matrix, $\mathbf{U}$, that diagonalizes $\mathbf{L}$, such that $\mathbf{U}\mathbf{L}\mathbf{U}^t = \Lambda = diag\{\lambda_\ell\}$

is a non negative diagonal matrix , leading to an eigenvalue decomposition of $\mathbf{L}$ matrix as given:

$$\mathbf{L} = \mathbf{U}\Lambda\mathbf{U}^t = \sum_{\ell=0}^{N-1} \lambda_\ell \mathbf{u}_\ell \mathbf{u}_\ell^t, \tag{2.13}$$

where t is a transpose operator, $\mathbf{u}_\ell$, the column vectors of $\mathbf{U}$, are the set of orthonormal eigenvectors of $\mathbf{L}$ with corresponding eigenvalues, $0 = \lambda_0 < \lambda_1 \leq \lambda_2 ... \leq \lambda_{N-1} = \lambda_{max}$. [26]. The normalised Laplacian matrix $\mathcal{L}$ can be defined as:

$$\mathcal{L} = D^{-1/2}\, \boldsymbol{L}\, D^{-1/2}. \tag{2.14}$$

The difference between the combinatorial graph Laplacian and the normalised graph Laplacian is not clear in terms of which one represents the optimal version of the Laplacian matrix. However, the two versions of Laplacian matrices have a similar notion of frequency [26]. In general, the combinatorial Laplacian matrix is utilised in the applications of image processing [25, 78] due to the combinatorial Laplacian matrix provides useful bases for images such as the DC component of the classical transforms. The eigenvectors have been utilised in analysing graph spectra both algebraic and analytic wise [79]. The eigenvectors of the graph provide an effective representation of the graph connectivity and the graph structure.

2.4.1.1 Graph Fourier Transform (GFT)

The traditional Fourier transform for regular signals is defined as the expansion of a function $\mathbf{x}(t)$ in terms of the complex exponentials:

$$\mathbf{X}(\omega) = < \mathbf{x}, e^{i\omega t} > = \int_{\mathbb{R}} \mathbf{x}(t) e^{-i\omega t} dt. \tag{2.15}$$

In this equation, $e^{i\omega t}$ is the eigenfunction of one-dimensional Laplace operator:

$$-\frac{\partial^2}{\partial t^2} e^{i\omega t} = \omega^2 e^{iwt}. \tag{2.16}$$

Based on the above definitions, the graph Fourier transform is defined as the expansion of a function $\mathbf{x}$ in terms of the eigenvectors of the graph variation operators by projecting the graph

signal $\mathbf{x}$ onto the eigenvector $\mathbf{u}$ in $\mathbb{R}^2$ which are the basis functions. The eigenvectors of graph Laplacian provide a harmonic analysis of signals of the graph which is a similar interpretation of traditional Fourier transform. The basis functions are fixed in the traditional Fourier transform, while the basis functions are unfixed in the graph Fourier transform, which depend on the graph connectivity between the vertices and the type of graph Laplacian. Figure 2.4 shows an example of a graph signal on random graph in vertex domain and its graph Fourier coefficients. The graph Fourier transform has a similar frequency concept as in the traditional Fourier analysis, where the graph Laplacian eigenvectors are varied slowly across the graph when the eigenvalues are close to the smallest eigenvalue λ_0. While the graph Laplacian eigenvectors are varied rapidly when the eigenvalues are close to the largest eigenvalue λ_{max} as shown in Figure 2.5. We show the basis functions of the sensor and swiss-roll graphs with 8 nodes as an example in Figure 2.6 and Figure 2.7. It is important to mention that the first eigenvector of the combinatorial graph Laplacian is a constant vector and depends on the number of nodes of graph, which is equal to $1/\sqrt{N}$ at each node for all types of graphs, which is similar to the DC component of the classical transforms. The graph Fourier transform is defined as given [25,26]:

$$\mathbf{X}(\ell) = \sum_{i=0}^{N-1} \mathbf{x}(i)\mathbf{u}_\ell(i). \tag{2.17}$$

The inverse Graph Fourier Transform is defined as follows:

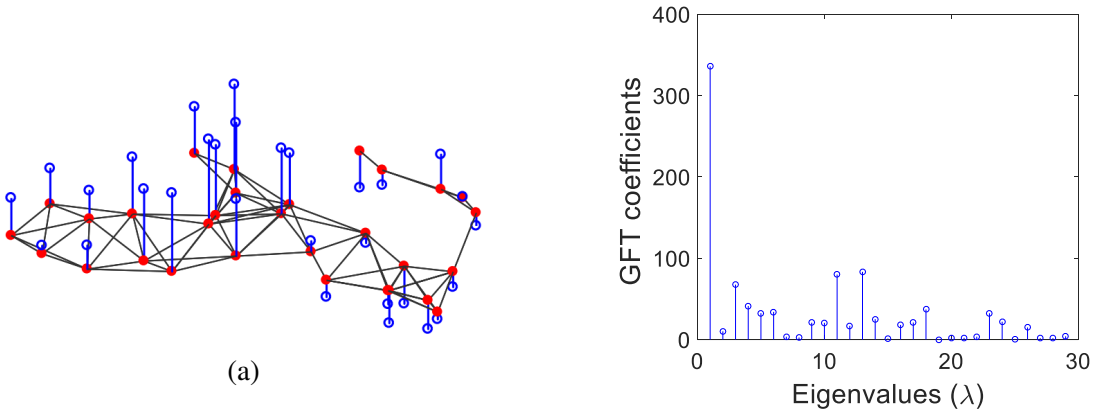
$$\mathbf{x}(i) = \sum_{\ell=0}^{N-1} \mathbf{X}(\ell)\mathbf{u}_\ell^t(i), \tag{2.18}$$

where t is a transpose operation.

The graph Fourier transform satisfies the condition of the Parseval's theorem, that means the sum of the square signals of graph is equal to the sum of the square graph Fourier coefficients as shown in the following equation:

$$\|\mathbf{x}\|_i^2 = \sum_{i=1}^{N} |\mathbf{x}(i)|^2 = \sum_{\ell=0}^{N-1} |\mathbf{X}(\lambda_\ell)|^2 = \|\mathbf{X}\|_\ell^2 \tag{2.19}$$

Most of the energy of the graph Fourier coefficients is concentrated in the first half; this part represents the low-frequency coefficients, which correspond to the smaller eigenvalues (from



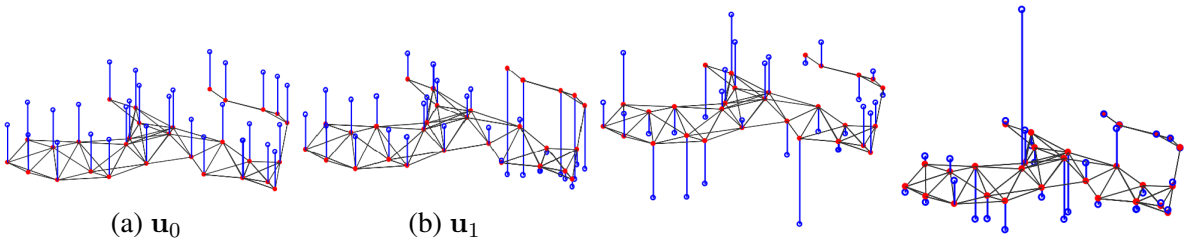
(a) (b)

Figure 2.4: Graph signal and its spectrum. (a) Signal on random network graph (vertex domain). The black lines, red circles, and blue lines indicate the edges, nodes, and signals, respectively. (b) Graph Fourier coefficients.

DC component to coefficient associated with the eigenvalue $\frac{\lambda_{max}}{2}$). The second half of the graph Fourier coefficients represents the high-frequency coefficients, which are associated with the larger eigenvalues [26].

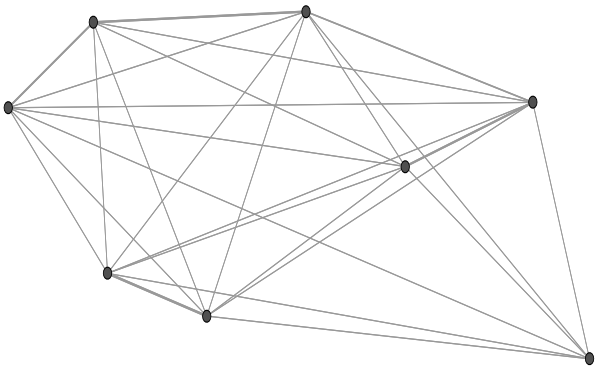
2.4.1.2 Graph Wavelet Transform (GWT)

Several graph wavelet transform approaches have recently been proposed [2, 80–90], one being perfect reconstruction transforms [2, 80, 81, 87]. However, some of them are only applicable in very restricted situations; for example, those in [80, 81] can only be used in the case of the bipartite graphs (which is a graph whose vertices can be divided into two disjoint and independent sets V_1 and V_2 such that every edge connects a vertex in V_1 to one in V_2). In [91], an

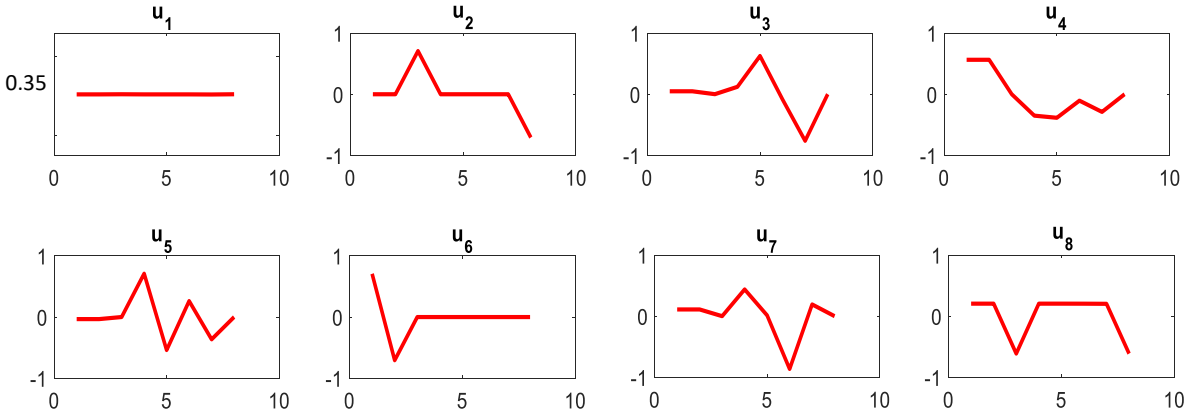


(a) $\mathbf{u}_0$ (b) $\mathbf{u}_1$ (c) $\mathbf{u}_9$ (d) $\mathbf{u}_{29}$

Figure 2.5: Four graph Laplacian eigenvectors of a random graph. The signals' component values are represented by the blue bars coming out of the vertices. We note that $\mathbf{u}_{29}$ contains many more zero crossings than the constant eigenvector $\mathbf{u}_0$ and the smooth Fiedler vector $\mathbf{u}_1$.



(a) Sensor graph



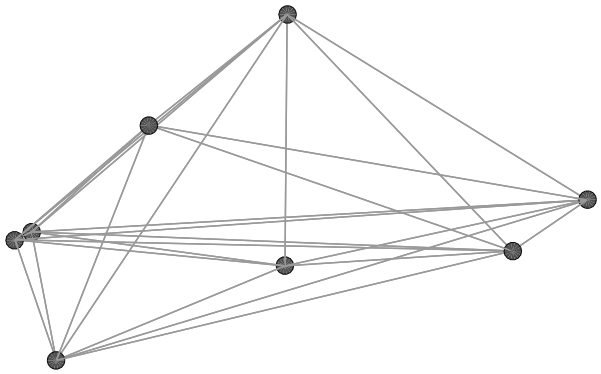
(b) The basis functions of sensor graph

Figure 2.6: Sensor graph with 8 nodes and its basis function. (a) Sensor graph. (b) The basis functions of sensor graph.

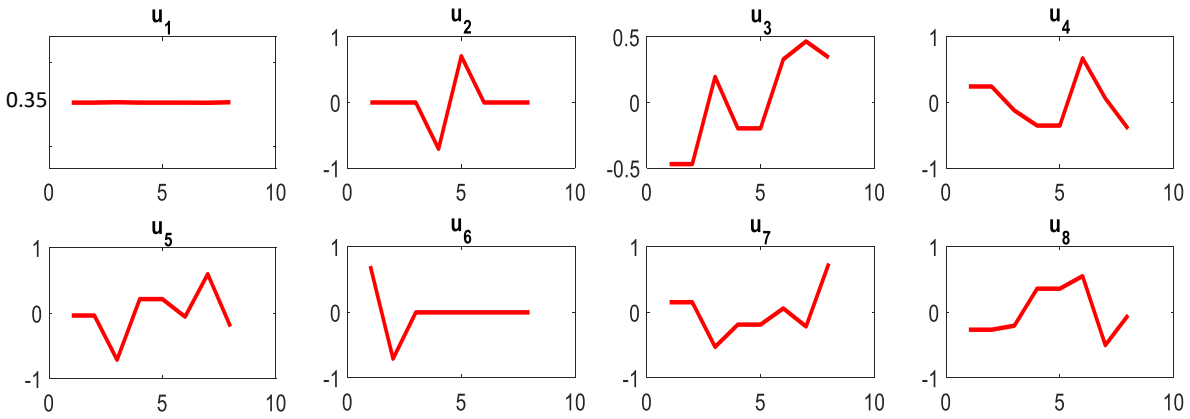
M-channel filter bank for arbitrary graphs is proposed, which requires the use of interpolation in the synthesis side. Other perfect reconstruction approaches in the vertex domain transform that can be applied to a specific kind of graph include [88, 92, 93]. In [2], a critically sampled filter bank is proposed for arbitrary graphs with perfect reconstruction. The proposed method has a symmetric structure in the analysis and synthesis sides.

In the time domain, a sampling of a signal is performed by downsampling the signal by a factor of two before upsampling the signal by a factor of two by adding zeros. While in the frequency domain, two components are obtained: the original frequency content of the signal and the modulated version of the original spectrum [94–96].

For the graph signal, downsampling and upsampling in the graph vertex domain is done by removing some nodes based on a specific condition before inserting zeros [64, 97–100]. In the graph spectral domain, the obtained signal cannot be separated into two components, main and



(a) Swiss-roll graph



(b) The basis functions of swiss-roll graph

Figure 2.7: Swiss-roll graph with 8 nodes and its basis function. (a) Swiss-roll graph. (b) The basis functions of swiss-roll graph.

aliasing. Recently, a graph spectral domain sampling approach was proposed that is similar to signal sampling in traditional signal processing [101]. Table 2.3 illustrates the properties of the most common graph wavelet filter banks methods.

Table 2.3: Comparisons between various graph wavelet filter banks methods

Method	Vertex domain sampling	Spectral domain sampling	Orthogonality	Perfect reconstruction
[2]	-	✓	Orthogonal and Biorthogonal	✓
[102]	✓	-	Orthogonal	✓
[103]	✓	-	Biorthogonal	✓
[92]	✓	-	-	✓
[91]	✓	-	-	✓

2.4.1.2.1 Sampling of signals of graph

This section presents the sampling approaches of the graph signal in the graph node domain and the graph spectral domain.

Sampling in the graph vertex domain

The most common graph sampling method used in the graph vertex domain is [97, 98, 100]:
The process of downsampling the graph signals can be defined in as:

Definition 1: Let $\mathcal{G}1 = (\mathcal{V}1, \mathcal{E}1)$ and $\mathcal{G}2 = (\mathcal{V}2, \mathcal{E}2)$ be the original and downsampled graphs, respectively. The original signal of graph $x \in R^{|\mathcal{V}_1|}$ and the downsampled signal $x_d \in R^{|\mathcal{V}_2|}$.

$$x_d[n] = x[n'], \quad \text{if } v_1, n' \in \mathcal{V}_1 \text{ corresponds to } v_2, n \in \mathcal{V}_2. \tag{2.20}$$

In the graph node domain, the upsampling process of the signals of graph signal can be defined as:

Definition 2: Let $\mathcal{G}_1$ and $\mathcal{G}_2$ are defined as in Definition 1. The original signal $x \in R^{|\mathcal{V}_2|}$ and the upsampled signal $x_u \in R^{|\mathcal{V}_1|}$ as follows:

$$x_u[n] = \begin{cases} x[n'], & \text{if } v_{n'} \in \mathcal{V}_2 \text{ corresponds to } v_n \in \mathcal{V}_1, \\ 0, & \text{otherwise .} \end{cases} \tag{2.21}$$

Sampling in the graph spectral domain

In the graph spectral domain, the sampling of the graph signals is as follows [101]:
The process of downsampling the graph signals can be defined in as:

Definition 3: Let $\mathbf{L}_1 \in R^{N \times N}$ and $\mathbf{L}_2 \in R^{N/2 \times N/2}$ be the original and downsampled graph Laplacian matrices, respectively. The eigendecompositions of $\mathbf{L}_1$ and $\mathbf{L}_2$ are given as $\mathbf{L}_1 = \mathbf{U}_1\Lambda_1\mathbf{U}_1$ and $\mathbf{L}_2 = \mathbf{U}_2\Lambda_2\mathbf{U}_2$, where $\Lambda_\ell = diag(\lambda_{\ell,0}, \lambda_{\ell,1}, ..., \lambda_{\ell,max})$. In the graph spectral domain, the downsampled graph signal $\mathbf{X}_d \in R_{N/2}$ is given as follows:

$$\mathbf{X}_d[i] = \mathbf{X}[i] + \mathbf{X}[N - i - 1], \tag{2.22}$$

where $i = 0, ..., N/2 - 1$.

In the graph spectral domain, the Upsampling process of graph signals is defined as follows:

Definition 4: Let $\mathbf{L}_1 \in R_{N\times N}$ and $\mathbf{L}_3 \in R_{2N\times 2N}$ be the graph Laplacian matrices of the original and upsampled graphs. We can define the upsampled signal of the graph in the graph spectral domain $\mathbf{X}_u$ as:

$$\mathbf{X}_u[i] = \begin{cases} \mathbf{X}[i], & i = 0, ..., N-1 \ , \\ \mathbf{X}[2N - i - 1], & i = N, ..., 2N - 1.. \end{cases} \tag{2.23}$$

2.4.1.2.2 Two-channels graph wavelet filter banks

The most common graph filter bank methods designed for bipartite graphs are outlined in [81, 102]. These methods constitute perfect reconstructions in cases of bipartite graphs. Non-bipartite graphs should be converted to bipartite graphs to get a perfect reconstruction. In [2], a two-channel filter bank approach is proposed for all graph types. This method satisfies the condition of perfect reconstruction for any graph type and sampling the graph signal in the spectral domain. In addition, this method uses the same structure on both sides of the analysis and synthesis filter banks.

A graph signal $\mathbf{x}$ is decomposed by two-channels wavelet filter bank $\{H_k, G_k\}_{k=0,1}$ into two sub-bands: low-pass graph signal and a high-pass graph-signal component. The low sub-band captures the low frequency of the graph signals (smooth) while the high sub-band captures the high frequency of the graph signals (detail). In this thesis, we use a structure similar to that of the two-channel graph filter banks designed in [2] because it provides for perfect signal reconstructions for all graph types and has properties similar to those of the classical wavelet transform (for more details, see: [2]). Figure 2.8 illustrates the two-channels filter banks on graphs.

where H_k and G_k are the the kth filters in the both sides of analysis and synthesis filter banks, respectively, such that $H_k = UH_k(\Lambda)U^t$ and $G_k = UG_k(\Lambda)U^t$ as given:

$$H_k(\Lambda) = diag(H_k(\lambda_0), H_k(\lambda_1), ..., H_k(\lambda_{N-1})),$$

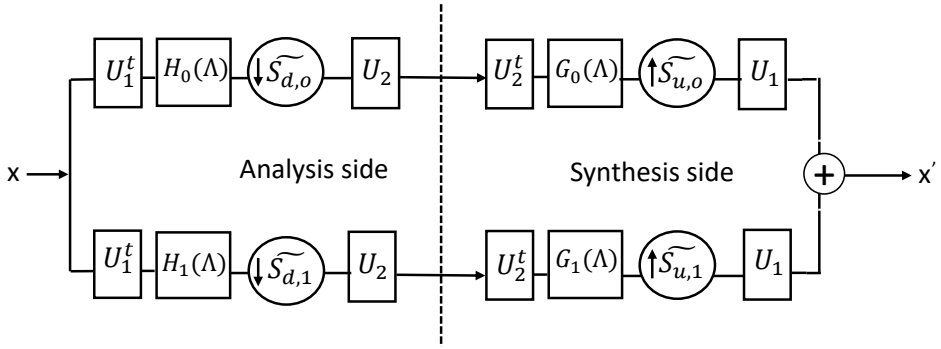
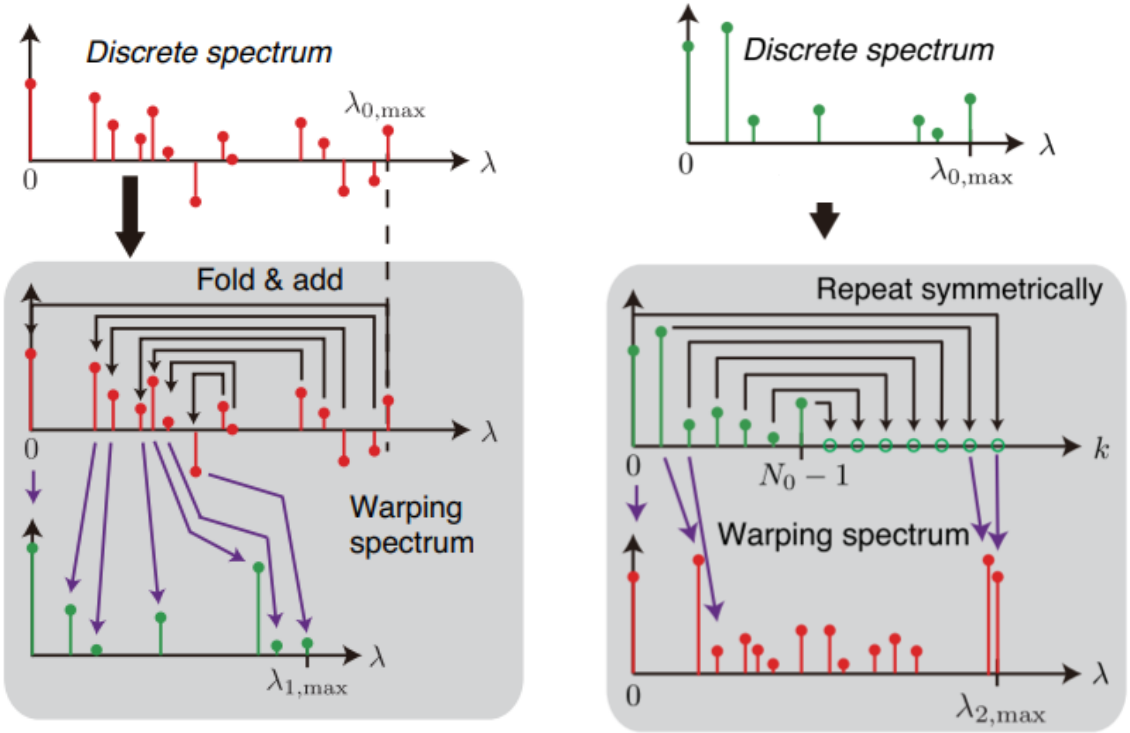


Figure 2.8: Two channels graph wavelet filter banks [2].

$$G_k(\Lambda) = diag(G_k(\lambda_0), G_k(\lambda_1), ..., G_k(\lambda_{N-1})).$$

The downsampling matrix $\tilde{S}_d$ is defined as $\tilde{S}_d = [I_{N/2}\ J_{N/2}]$, where I is the identity matrix and J is the anti-diagonal matrix. The upsampling matrix $\tilde{S}_u$ can be defined as $\tilde{S}_u = [I_N\ J_N]^t$, where t is the transpose operation. Figure 2.9 shows the downsampling in the graph spectral domain.



(a) Downsampling

(b) Upsampling

Figure 2.9: Downsampling and upsampling in graph spectral domain [2]. (a) Downsampling. (b) Upsampling.

The GWT coefficients (low-frequency and high-frequency) are defined based on the following equations respectively:

$$\mathbf{Y}_L = U H_0(\Lambda) U^t \mathbf{x}, \tag{2.24}$$

$$\mathbf{Y}_H = U H_1(\Lambda) U^t \mathbf{x}, \tag{2.25}$$

$$\mathbf{Y} = \begin{cases} \mathbf{Y}_L, & \text{Low-frequency coefficients ,} \\ \mathbf{Y}_H, & \text{High-frequency coefficients .} \end{cases} \tag{2.26}$$

where $\mathbf{x}$ is the graph signal, $\mathbf{Y}_L$ and $\mathbf{Y}_H$ are the low and high frequencies GWT coefficients, respectively.

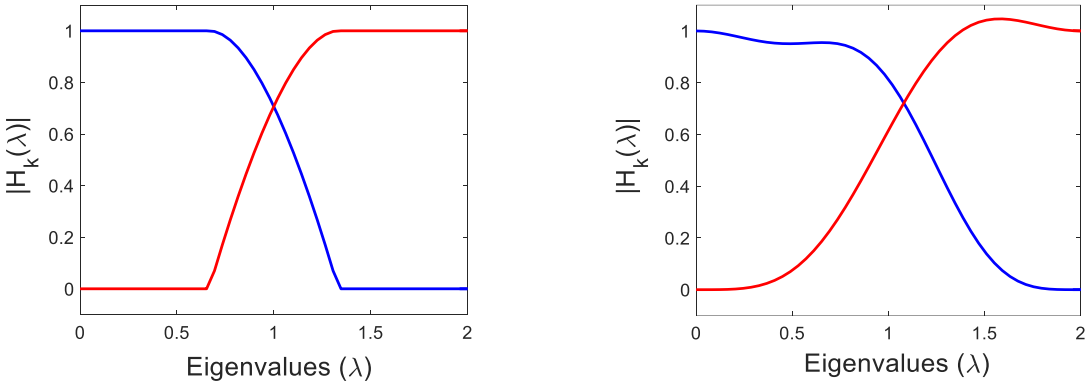
Two kinds of graph filters have been used in this thesis:
1- GraphQMP filter [102]: This filter is an orthogonal filter. The Meyer wavelet kernel $H_0(\lambda)$ is designed first then the rest filters are designed based on it as follows:

$$\begin{aligned} H_1(\lambda) &= H_0(2-\lambda), \\ G_0(\lambda) &= H_0(\lambda), \\ G_1(\lambda) &= H_1(\lambda) = H_0(2-\lambda), \end{aligned}$$

where H_0 satisfy the perfect reconstruction condition as given:

$$H_0^2(\lambda) + H_0^2(2-\lambda) = c^2, \tag{2.27}$$

Figure 2.10 (a) shows the orthogonal Meyer wavelet kernel [102].
2- GraphFC [87]: A method has been proposed for converting time domain filters $H(\omega)$ into graph spectral filters $H(\lambda)$ through a frequency mapping from $\omega \in [0,\ \pi]$ to $\lambda \in [0,\ \lambda_{max}]$ [87]. In this approach, the perfect reconstruction conditions are always satisfied as long as the set of time domain filters are perfect reconstruction (in the time domain). The analysis filter characteristics based on the CDF 9/7 filters [104] are shown in Figure 2.10 (b).



(a) (b)

Figure 2.10: Orthogonal and Bi-orthogonal GWT kernels where the blue curve is the low-pass filter and the red curve is the high-pass filter. (a) Meyer wavelet kernel. (b) Bi-orthogonal 9/7 kernel. blue line: low-pass, red line high-pass

2.5 Graph spectral domain irreversible data hiding

Several works have proposed irreversible data hiding techniques for protecting structured data on Cartesian grids. However, there is little work on data hiding techniques for protecting and authenticating unstructured data such as graph data. In recent years, the most sensitive datasets are captured in large graphs. A significant challenge facing the data owners is how to share sensitive graphs with collaborators or authorized users, e.g. ISP's network topology graphs with a third party networking equipment vendor. One way to protect the graph data is to embed an author's signature s for intellectual property protection (IPP). Existing research on protecting graph data is based on the graph vertex domain using graph colouring. It is a well-known that a graph colouring problem is considered as a type of hard problems. The graph colouring problem is to label the nodes of a graph with minimal number of colours such that nodes connected by an edge are not labelled with the same colour. The most common approaches are: adding extra edges [7]; Maximal Independent Set (MIS) [105]; inserting new nodes [6, 106] and hiding sub-graphs created from the secret data [8]. The main idea of the first approach is to add an extra edge between two nodes based on the binary message, these two nodes have to be coloured by different colours which may not be necessary in the original graph. A maximal independent set (MIS) of a graph is a subset of nodes such that nodes in the subset are not connected and those not in the subset are connected to at least one node of the subset. The essence of this approach is

to select one or more set (s) MIS according to binary message, assign each MIS with one colour and then colour the rest of the graph. This approach takes advantage of the fact that nodes in one MIS can all be labelled by a single colour. Another approach is inserting new nodes to the original graph and connect them depending on the binary message. Current approaches can provide limited node or edge privacy, but significantly modify the graph reducing its utility. An alternative approach is proposed by Zhao et al. [8] in the form of graph watermarks. Graph watermarks are small graphs tailor-made for a given graph dataset, a secure graph key, and a secure user key.

The majority of graph data hiding approaches are essentially related to a mesh mostly based on watermarking. A mesh is a specific type of graph, which is itself a mathematical structure most simply defined as being a collection of vertices and edges together with a relationship stating that every edge of the graph connects either two vertices or a vertex to itself. More formally a mesh is a graph that defines the shape of an object in modelling and includes polygonal faces formed by components of the graph. Additionally, a mesh is a type of simple graph, in which there are no edges connecting a vertex to itself. The data hiding approaches are primarily classified into two categories based on the embedding domain: vertex domain and spectral domain. In the vertex domain, the secret data are hidden by adjusting the mesh coordinates while in the spectral domain, the mesh coefficients are modified. Several approaches have been proposed in the vertex domain. Based on localised geometrical changes to the selected nodes a watermarking approach is proposed by Bors to improve robustness against noise perturbation and object cropping [107]. In the same context, Gao et al. [108] propose a watermarking method based on using affine invariants ratios to embed the watermark. This method provides a high payload and robustness against the cropping noise affine transformations. Based on modifying the distribution of the vertex norm, watermarking methods are proposed in [109, 110]. In [111], another work has made proposal based on modifying the geodesic distances. Using the distance between the centre of the mass and the vertices of a mesh surface to embed the watermark are proposed in [112–114]. Hou et al. [115] propose a robust approach for the 3D printing process that depends on the node locations, which are spread cylindrically. A semi-fragile watermarking approach is proposed by Borah et al. [116]. The approach detects tampering in mesh vertices and manipulation in mesh topology through a bit substitution method. Robust watermarking is

also achieved by exploring the machine learning and genetic algorithm in [117]. The K-means algorithm and particle swarm optimization are employed to choose the mesh vertices for hiding the watermark. The results in [117] illustrate that the method improves the imperceptibility of watermarked mesh and robustness against attack. A watermarking algorithm aimed at copyright protection based on hiding a grey-scale image in the mesh vertices is suggested in [118]. This method is robust against common attacks such as scaling, rotation and translation. In order to minimize the embedding distortion, Corsini et al. [119] propose two objective metrics for watermarking distortion by measuring the roughness of the mesh surface. Medimegh et al. [120] suggest a watermarking approach to embed the watermark on extracted silent regions.

Spectral (frequency) domain data hiding has been shown to be an effective method for protecting images and videos due to developments in signal transforms [16, 18, 19, 121–124]. Existing approaches to graph data hiding in the spectral domain are essentially related to a mesh. In general, these approaches are categorised by type of transform. The most common type is the combinatorial Laplacian transform, with which the spectral coefficients are obtained by projecting the mesh coordinates onto the graph Laplacian matrix according to the mesh connectivity [125, 126]. Using combinatorial Laplacian transform a non-blind data hiding approach is proposed by Ohbuchi et al. [127] wherein the secret bits are hidden by modifying the low-frequency and medium-frequency mesh coefficients . However, the computational cost of this approach is high ($O(N^3)$). Additionally, the approach is sensitive to any change in mesh connectivity. Ohbuchi et al. [128] expand upon [127] by splitting the mesh into patches and embedding the secret bits in each patch. This method has a lower computational cost than does the previous method and is resistant to both cropping and mesh simplification. In the same context, Abdallah et al. [129] propose a robust watermarking approach based on embedding watermarking bits in spectral coefficients. [130, 131] suggest other works based on manifold harmonics and the Dirichlet manifold harmonic transform [132] for meshes. Feng et al. [133] present a new algorithm that uses feature segmentation through DCT transform and redundancy information; the method is robust against similarity transformations and signal processing attacks. For 3D point-cloud watermarking, Qi et al. [134] suggest a blind algorithm that extracts the feature points to improve watermark transparency and noise immunity. [135] presents a new approach to hiding watermark bits based on distance normalisation modulation; this approach is robust

against cropping, noise, reorder and similarity transformation. Another blind algorithm using DCT for copyright protection is presented in [136]; it demonstrates high watermark robustness. Liu et al. [137] propose a watermarking algorithm based on embedding the watermark bits in selected feature vertices; the algorithm is robust against geometric attacks.

The second type is a multi-resolution analysis based on the wavelet transform. The main idea is to embed the watermark in the norm of the wavelet coefficients vector. In line with regular wavelet decomposition, Kanai et al. [138] propose a watermarking method based on the lazy wavelet transform. In which the watermark is embedded in the norms of the wavelet coefficient vectors that are higher than a threshold determined by the user; the method is resistant to similarity transformations. In the same context, Uccheddu et al. [139] suggest a watermarking algorithm in which the embedding process depends on adjusting the norm of the wavelet coefficient. Based on the irregular wavelet transform on mesh [140], Kim et al. [141] propose a watermarking approach using a correlation-based scheme in which the watermark is hidden in groups of wavelet coefficient vectors; this approach is robust against geometric attacks but is not robust against mesh-connectivity modifications. Wang et al. [142] propose a watermarking algorithm for embedding the watermark using various levels of resolution; the coefficients with the lowest resolution are used for a robust watermark while the coefficients with the highest resolution are used for a fragile watermark. Other works based on the wavelet transform, where the watermark bits are hidden in different levels of resolution, are suggested in [142, 143]. Kim et al. [144] propose a watermarking method using the B-spline model through which the watermark bits are hidden in the spectral coefficients using the spread spectrum method. Hachani et al. [145] suggest a watermarking framework using the irregular wavelet transform. The proposed embedding process entails quantizing the wavelet coefficient vector norms to hide the watermark bits. This approach is robust against common geometric attacks and its payload is relatively high. The other application of blind watermarking is copyright protection, which is based on modulating the norm of the wavelet coefficients, as shown in [146]. The proposed approach is robust against several popular attacks, including additive noise, rotation, translation and Laplacian smoothing. Another method for copyright protection is proposed by Hamidi et al. in [147], which uses saliency and wavelet transform on a mesh; its embedding process depends on using the quantization index modulation algorithm.

There is limited research on protecting and authenticating graph data by adding nodes or edges. However, these methods are less resistant to many attacks because they depend on the vertex domain and the computational complexity of these methods is high. Moreover, they are insecure because the embedded data can be easily detected by comparing graph topologies. In mesh-based data hiding, the secret data are embedded in the mesh coordinates or coefficients without considering the graph signal. In addition, existing work on graph data hiding has a limitation in terms of identifying the relationship between embedding distortion and the selected coefficients for reducing embedding distortion and the relationship between the extraction of secret bits and the effect of the attacks for improving method performance.

2.6 Graph-based reversible data hiding

Several research have been proposed in reversible data hiding. The majority of them are based on images [148–163]; there is limited research on graph-based RDH because most RDH algorithms for images cannot be applied to graphs. The main mesh-based methods for graph-based RDH are classified into two groups: Difference Expansion (DE) and Histogram Shifting (HS).

Difference-expansion-based RDH hides the secret bits by expanding the differences between the neighbouring coordinate values of the host mesh [164]. In this context, Wang et al. [165] propose a method for embedding secret bits based on modulating the difference between the neighbouring coordinates. Another approach is proposed by Lu et al. in [166] that uses predictive vector quantization; this approach can recover the original mesh data after extracting the secret bits.

The first RDH method using histogram shifting was proposed by Ni et al. [3]. In this method, the histogram of the host media is generated to identify the zero point (or minimum point) and the peak point (or maximum point) for use in embedding the secret bits. Hong et al. [167] propose another approach based on combining prediction and histogram shifting (PHS); this method has a high payload and robustness against histogram analysis. For copyright protection of a 3D model, Li et al. [168] suggest a robust RDH approach using homomorphic encryption and histogram shifting. The proposed method is robust against Gaussian noise, translation and scaling; additionally, it has less distortion in the decrypted models. Lee et al. [169] propose an

approach using joint compression to generate levels of details (LoDs); the secret bits are hidden in the regions of each level by modifying the geometry information of the mesh vertices through histogram shifting. The reversibility of this method is not guaranteed for all vertices and the embedding capacity depends on the number of the regions of the LoDs.

Many recent RDH schemes combine the DE or HS and the prediction error (PE) by taking advantage of correlation among neighbouring vertices to achieve better performance. Jiang et al. [170] propose an RDH algorithm to hide the secret bits using the optimal 3D prediction error histogram modification combined with recursive construction coding. With this approach, the vertices of the mesh are divided into two categories embedded and referenced before the secret bits are hidden by adjusting the prediction error triple. Shah et al. [171] propose a two-tier RDH-ED method based on the homomorphic Paillier cryptosystem; this method recovers both the secret data and the original data without error and offers high embedding rates.

Based on the correlation between the neighbouring vertices, Luo et al. [172] propose an approach to hiding the secret bits by taking advantage of the high correlation among neighbouring nodes. In line with prediction-error expansion, Wu et al. [173] propose an approach to hide the secret bits by expanding the difference between the predicted and real positions of the mesh vertices; this prediction method is more precise when PEs are generated with smaller values. However, the use of partial prediction contexts limits predictive accuracy [170, 173]. In order to obtain an accurate prediction, Zhang et al. [174] suggest using a ring prediction context that predicts the position of the central vertex precisely relative to the partial prediction context. With this approach, the central vertex is predicted based on its one-ring neighbours before the secret bits are hidden by adjusting the central vertex; this method has a high payload and low embedding distortion. Borah et al. [175] propose an approach based on a prediction error histogram for mesh authentication; this method is able to restore the original mesh data and minimize embedding distortion.

Some of the existing RDH algorithms cannot be reversed. For instance, the RDH using the least significant bits (LSB) is a lossy method due to bits replacement. Another example is the RDH using quantization, which is also not invertible due to the quantization error. In addition, the RDH using prediction error is not error-free, due to truncation error and round-off error. Most of these algorithms can easily apply when the data are positive integers. Moreover, some

of the RDH algorithms require that some extra data be stored to precisely restore the original signal. Other algorithms use lossless compression to compress the embedded data to reduce the distortion in the host media.

Existing work on graph-based reversible data hiding relies primarily on mesh. In mesh-based reversible data hiding, the mesh coordinates are used to hide secret data without considering the graph signal. In addition, there is no work on graph reversible data hiding that identifies the relationship between embedding distortion and embedding parameters, which would help reduce embedding distortion, or the relationship between the extraction of secret data and the effect of the attacks on improving robustness.

2.7 Concluding remarks

This chapter presented a background of data hiding techniques and graph spectral theory. We reviewed relevant works on spectral domain irreversible and reversible data hiding. From the literature, we can conclude the following:

1. Most of the existing work on protecting graph data is based on the graph vertex domain, which is not robust against many attack types and insecure because the embedding process depends on graph topology. In order to solve this issue, we propose a data hiding algorithm in the graph Fourier domain that is robust against many attack types and that hides the secret bits in the graph coefficients without changing the graph topology. This algorithm has high embedding capacity because it depends on graph size, as shown in Chapter 3.

2. The majority of reversible data hiding approaches for graphs are related to mesh as explained previously). In these approaches, the secret bits are hidden in the mesh coordinates without considering the graph signal. We propose a new reversible data hiding algorithm in the graph spectral domain with two new models to minimise embedding distortion and enhance robustness. The embedding process depends on the graph signal without changing graph topology, as shown in Chapter 4.

3. The discrete wavelet transform is considered to be a powerful tool in signal processing

because it can represent signal contents in two domains: spatial and frequency. Additionally, it provides multi-scale representations of signals; therefore, the wavelet transform is considered to be an optimal choice for multimedia data hiding. We use the advantages of the wavelet transform to propose a new data hiding algorithm in the graph wavelet domain, as shown in Chapter 5.

4. There is limited work addressing the embedding distortion and the robustness of data hiding in terms of identifying the coefficients which can reduce the embedding distortion or can retain the embedded data after the attacks. In Chapter 3, Chapter 4 and Chapter 5, we identify the relationship between embedding distortion performance metrics and the spectral coefficients selected to embed secret data in order to reduce the embedding distortion. Moreover, we propose robustness models based on establishing the relationship between secret data extraction and the effects of attacks to enhance the robustness of data hiding against attacks.

The next chapter introduces the graph Fourier domain irreversible data hiding for graph data.

Chapter 3

Graph Fourier domain irreversible data hiding for graph data

3.1 Introduction

This chapter proposes a novel method for irreversible data hiding for graph data. Recent years have seen an increase in the applications of social networks and sensor networks which can be represented as a weighted graph as discussed in chapter 2. Unfortunately, the techniques of the classical signal processing cannot be applied to those graphs that have irregular structure. In this chapter, a spread-spectrum data hiding method is proposed for protecting and authenticating the graph data based on exploiting the spectral decomposition of graph data. However, the majority of the existing work interests in protecting and authenticating the data that are represented on regular structures (with following the Cartesian grid) such as images and videos while there is a limited work on protecting and authentication irregular structure data. The most popular methods for authenticating the graph data are based on inserting additional vertices [7]; adding more edges [6]; hiding sub-graphs [8] as shown in chapter 2. As these methods depend on the vertex domain, they cannot withstand various types of attacks. The cost of the computational complexity of these methods is high. They are also insecure because the secret bits can be detected when the original graph is available by comparing the topologies of the graphs [7, 8]. In addition, since the embedding process is based on the topology of the graph, the embedding capacity is small.

On the other hand, spread-spectrum data hiding has proven to be an effective approach in digital multimedia protection in accordance to the advances in signal transform [9–22, 176]. In this chapter, we propose a novel spread-spectrum data hiding method for unstructured graph data. The proposed method exploits the recently advances in graph signal processing on graph spectral decomposition of the Laplacian matrix, which captures the nodes connectivity [25, 26]. The key point of the proposed methods using graph spectral domain is to embed the secret data in the graph coefficients (by considering the graph signal) without changing the graph topology (which is considered in the existing work). Therefore, we consider the graph topology as a secret key available to the receiver. If the graph is compromised the secret data cannot be extracted, then the graph data cannot be authenticated. For example, we extract some important information from the graph topology such as $(1/\sqrt{N})$, summation of the fiedler vector, the number of zeros crossing in the last eigenvector, the maximum eigenvalue, the summation of the eigenvalues. If the graph topology is changed, for example, if the number of the nodes is changed this will change the basis functions (eigenvectors) and the value of the first eigenvector (constant eigenvector) which is $(1/\sqrt{N})$ will change. Also, removing some edges will change the adjacency matrix A and will generate new basis functions and eigenvalues. Therefore, these important information are obtained from the graph structure and sent to the receiver in a separate file to the receiver. So any change in this information means the graph is tampered and the secret data are destroyed.

Spread spectrum data hiding is one of the most secure techniques of data hiding because the secret data are spread over many frequency bands so that the energy in one band is undetectable. Cox et al. [177] propose a secure algorithm for watermarking images, and a methodology for digital watermarking that may be generalized to audio, video, and multimedia data. In this paper, a watermark is imperceptibly inserted in a spread-spectrum like fashion into the perceptually most significant spectral components of the data in order to make the watermark robust to signal processing operations (such as lossy compression, filtering, requantization, etc.), and common geometric transformations (such as cropping, scaling, translation, and rotation) provided that the original image is available and that it can be succesfully registered against the transformed watermarked image. Kumar et al. [178] propose a secure spread-spectrum watermarking algorithm for digital images using discrete wavelet transform (DWT) domain. The

watermark such as patient identification or doctors signature is embedded into host digital radiological image for potential telemedicine applications. Simulation results show that the proposed method achieves higher security and robustness against various attacks. Liu et al. propose a watermarking technique using Double Random Phase Encoding spread-space spread-spectrum (DRPE SS-SS). The watermark is encrypted using a simulation of the optical DRPE process. This produces a random complex image, which is then processed to form a real valued random image with a low number of quantization levels. This signal is added to the host image.This algorithm is designed to utilize the capability of the DRPE to reversibly spread the energy of the watermarking information in both the space and spatial frequency domains. The results presented indicate that the DRPE SS-SS method is robust to spatial cropping and both low and high pass filtering [179].

For any data hiding system, the main requirements are minimum embedding distortion and high robustness. The existing vertex-domain graph data hiding methods are focused on minimising the distortion [119, 120, 180] as well as improving the robustness against the attacks [109–113, 115]. Similarly, in [21], two mathematical models have been proposed to minimise the embedding distortion and make the watermark bits robust for scalable decoding attacks [15, 16] for general spread spectrum watermarking.

This chapter proposes a novel data hiding algorithm in graph Fourier domain by exploring the emerging field of graph signal processing for spread-spectrum data hiding. The proposed data hiding methodology includes two new models, the embedding distortion minimisation model and the robustness model. In the embedding distortion minimisation model, the relationship between the error distortion metric and the selected GFT coefficients to be modified is established in order to minimise the embedding distortion. To enhance the robustness, we establish the relationship between the extraction of the secret data and the effect of the attack, namely, noise addition and deleting nodes data. Finally, the conditions of the proposed models are combined in order to satisfy the two basic requirements of the data hiding. Two scenarios are considered: non-blind and blind data hiding. Blind data hiding is a useful method where the original graph signal is unavailable in the extraction process. The basic contributions of this chapter are:

1. Proposing new models to minimise the embedding distortion for graph Fourier domain

blind and non-blind data hiding.

2. Proposing new models to make the embedded data robust against the attacks for graph Fourier domain blind and non-blind data hiding.

The rest of this chapter is organised as the following: Section 3.2 describes the proposed methodology including the proposed graph Fourier domain data hiding, followed by the embedding distortion minimisation and the robustness models. Section 3.3 presents the evaluation of the proposed method performance. Finally, the concluding remarks are given in Section 3.4.

3.2 Proposed Methodology

This section presents the proposed irreversible data hiding algorithm in graph Fourier domain. We consider two embedding scenarios: non-blind and blind data hiding. Figure 3.1 shows the block diagram of the proposed method.

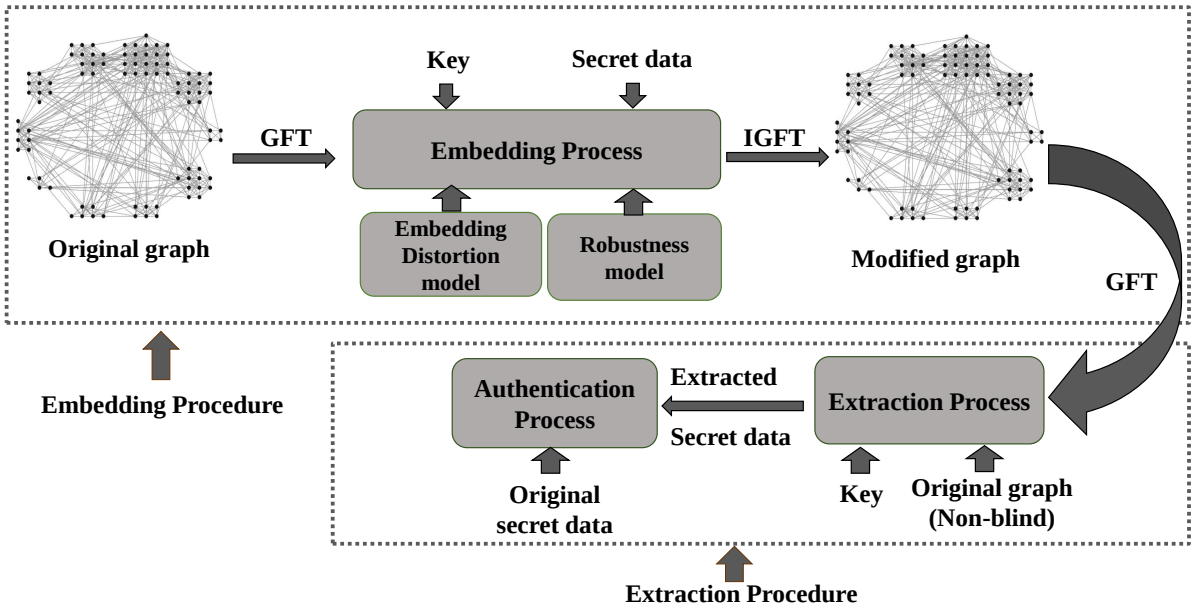


Figure 3.1: The block diagram of the proposed irreversible data hiding framework.

3.2.1 Graph Fourier Transform (GFT)

We suppose that $\mathcal{G}$ is an undirected graph without self loops and multiple edges between nodes, the adjacency matrix with edge weights, $\mathbf{A}$, can be defined as in Eq. (2.9). The combinatorial graph Laplacian matrix, $\mathbf{L}$, is calculated as in Eq. (2.11). The Graph Fourier Transform (GFT) and its inverse are defined as in Eq. (2.17) and Eq. (2.18).

3.2.2 GFT domain data hiding

3.2.2.1 Non-blind data hiding

The popular form of non-blind data hiding is the magnitude-based multiplicative watermarking [13]. Firstly, we calculate the graph Fourier coefficients using Eq. (2.17), then the secret bits are hidden in the GFT coefficients $\mathbf{X}$ as follows:

$$\mathbf{X}_w = \mathbf{X}(1 + \alpha w_b), \tag{3.1}$$

where $\mathbf{X}_w$ is the modified GFT coefficient, α is the data hiding parameter and w_b is the embedded bit. The inverse GFT is performed on the modified GFT coefficients using Eq. (2.18) to obtain the modified graph.

The extraction process requires the original coefficients to extract the embedded data. The embedded bits are extracted based on the embedding key which is sent to the receiver in a separate file. The embedding key includes N, w_0, w_1, length of the secret bits and α. The GFT is performed on the modified graph, then the extracted bit is obtained as given in the following equation:

$$w'_b = \frac{\mathbf{X}_w - \mathbf{X}}{\alpha \mathbf{X}}, \tag{3.2}$$

where w'_b ($b \in \{0, 1\}$) is the extracted bit. Let w_0 and w_1 are the selected embedded bits for embedding a 0 and 1, respectively, where $w_0 < w_1$. The extracted secret bit b' is determined

according to a threshold T, where $T = (w_0 + w_1)/2$, whereas:

$$b' = \begin{cases} 0 \quad , & \text{if } \ w'_b < T, \\ 1 \quad , & \text{if } w'_b \geq T. \end{cases} \tag{3.3}$$

3.2.2.2 Blind data hiding

We propose a blind method using a prediction-based graph data hiding. We first calculate the graph Fourier coefficients using Eq. (2.17), then, we sort them in descending order, $\mathbf{X}_s(m)$. Before we choose the GFT coefficients to embed the secret data, we test each three sorted spectral coefficients, $\mathbf{X}_s(m-1) \geq \mathbf{X}_s(m) \geq \mathbf{X}_s(m+1)$, if and only if they satisfy following condition:

$$\mathbf{X}_s(m-1) \geq \Big\lfloor \frac{\mathbf{X}_s(m-1) + \mathbf{X}_s(m+1)}{2} \Big\rfloor + w_b \geq \mathbf{X}_s(m+1). \tag{3.4}$$

The spectral coefficients are used in the embedding process. Otherwise (if the condition does not satisfy) the first coefficient is skipped. We start from the second coefficient and check the three coefficients again and so on. We use a code (0) for no skip coefficient and (1) for skipping coefficients followed by the locations of the coefficients. The secret key includes this information and it is sent to the receiver separately.

For embedding the secret data, a non-overlapping 3×1 running window is passed through the selected graph Fourier coefficients to hide the secret bit in the median GFT coefficient at each sliding position, as given in the following equation:

$$\mathbf{X}_{s_w}(m) = \Big\lfloor \frac{\mathbf{X}_s(m-1) + \mathbf{X}_s(m+1)}{2} \Big\rfloor + w_b, \tag{3.5}$$

where $\mathbf{X}_{s_w}$ is the modified coefficient, $\lfloor \mathbf{X} \rfloor$ refers to rounding of $\mathbf{X}$ to the largest integer number smaller than $\mathbf{X}$ and $w_b > 0$ is the secret bit. To obtain the modified graph, we perform the inverse GFT on the modified GFT coefficients by using Eq. (2.18).

The secret bits are extracted based on the embedding key which is sent to the receiver in a separate file. The embedding key includes N, w_0, w_1, length of the secret bits, number of skipped coefficients and the positions of the skipped coefficients. The graph Fourier transform is applied on the modified graph, followed by sorting in descending order to get sorted modified

graph Fourier coefficients, $\mathbf{X}_w(m)$. Then, the embedded bit from each 3×1 running window with coefficients, $\mathbf{X}_w(m-1) \geq \mathbf{X}_w(m) \geq \mathbf{X}_w(m+1)$, is extracted based on the secret key as shown in the following equation:

$$w'_b = \mathbf{X}_w(\ell) - \left\lfloor \frac{\mathbf{X}_w(\ell-1) + \mathbf{X}_w(\ell+1)}{2} \right\rfloor. \tag{3.6}$$

where w'_b ($b \in \{0, 1\}$) is the extracted bit. Let w_0 and w_1 are the chosen secret bits values for embedding a 0 and 1 , respectively. The extracted secret bit b' is determined according to a threshold T, where $T = (w_0 + w_1)/2$, as shown in Eq. (3.3).

3.2.3 Authentication Process

Authentication is applied based on comparing the extracted secret bits with the original secret bits using the Hamming Distance (HD) as defined as in Eq. (2.6).

3.2.4 Embedding distortion minimisation

A model is proposed for minimising the embedding distortion in graph Fourier domain based on establishing the relationship between the error distortion using mean square error (μ) and the chosen graph Fourier coefficient for data hiding. We define MSE (μ) in vertex domain between the original graph signal $\mathbf{x}$ and modified graph signal $\mathbf{x}_w$ as given:

$$\mu = \frac{1}{N} \sum_{i=0}^{N-1} (\mathbf{x}(i) - \mathbf{x}_w(i))^2, \tag{3.7}$$

where N is the number of graph nodes. Since the graph Fourier transform forms an orthogonal set of eigenvectors, according to the Parseval's Theorem, $\|\mathbf{x}\|^2 = \|\mathbf{X}\|^2$, where $\mathbf{x}$ is the signal of graph in vertex domain and $\mathbf{X}$ is the graph Fourier coefficient [25]. Because the graph Fourier transform is orthonormal, we can extend this to the sum of the error power in the input graph signal, $\Delta\mathbf{x}$, and to the sum of the error power in the GFT domain $\Delta\mathbf{X}$ as the following:

$$\sum_i |\Delta\mathbf{x}(i)|^2 = \sum_\ell |\Delta\mathbf{X}(\ell)|^2. \tag{3.8}$$

From Eq. (3.7) and Eq. (3.8), we obtain:

$$\mu = \frac{1}{N} \sum_{\ell} |\Delta \mathbf{X}(\ell)|^2. \tag{3.9}$$

We suggest two data hiding scenarios: non-blind and blind.

Proposition 3.1 (Non-blind)

For non-blind approach, the MSE (μ) of the modified graph is proportional to the energy sum of the chosen GFT coefficients for embedding:

$$\mu \propto \sum_{\ell=0}^{N-1} |\mathbf{X}(\ell)|^2. \tag{3.10}$$

Proof. In non-blind approach, the modified coefficients $\mathbf{X}_w(\ell)$ are calculated as follows:

$$\begin{aligned} \Delta \mathbf{X}(\ell) &= \mathbf{X}_w(\ell) - \mathbf{X}(\ell), \\ &= \mathbf{X}(\ell) + \mathbf{X}(\ell)\alpha w_b - \mathbf{X}(\ell), \\ \Delta \mathbf{X}(\ell) &= \mathbf{X}(\ell)\alpha w_b, \end{aligned}$$

where $\Delta \mathbf{X}(\ell)$ is the modification value due to embedding the secret bit. Since the GFT is orthonormal and from Eq. (3.9), thereby leading to the the relationship between the MSE and the selected GFT coefficients:

$$\mu \propto \sum_{\ell=0}^{N-1} |\mathbf{X}(\ell)|^2.$$

The embedding distortion is decreased when the value of $\Delta \mathbf{X}$ is decreased, in other words, when we choose the GFT coefficients with low values (because α and w_b can be considered as constants).

□

Proposition 3.2 (Blind)

In a blind approach, for any embedding coefficient triple $\mathbf{X_s}(m-1) \geq \mathbf{X_s}(m) \geq \mathbf{X_s}(m+1)$, the MSE ($\mu$) of the modified graph is proportional to the gradient difference of the embedding coefficient triple $[(\mathbf{X_s}(m-1) - \mathbf{X_s}(m)) - (\mathbf{X_s}(m) - \mathbf{X_s}(m+1))]^2$ as follows:

$$\mu \propto [(\mathbf{X_s}(m-1) - \mathbf{X_s}(m)) - (\mathbf{X_s}(m) - \mathbf{X_s}(m+1))]^2. \tag{3.11}$$

Proof. For any three sorted spectral coefficients, $\mathbf{X_s}(m-1) \geq \mathbf{X_s}(m) \geq \mathbf{X_s}(m+1)$, the modification value due to embedding the secret bits $\Delta\mathbf{X_s}(m)$ using the prediction algorithm is estimated from Eq. (3.5) as follows:

$$\begin{aligned}
\Delta\mathbf{X_s}(m) &= \mathbf{X}_{sw}(m) - \mathbf{X}_s(m), \\
&= \Big\lfloor \frac{\mathbf{X_s}(m-1) + \mathbf{X_s}(m+1)}{2} \Big\rfloor + w_b - \mathbf{X_s}(m), \\
\Delta\mathbf{X_s}(m) &= \Big\lfloor \frac{\mathbf{X_s}(m-1) + \mathbf{X_s}(m+1)}{2} \Big\rfloor - \mathbf{X_s}(m).
\end{aligned}$$

By substituting $\mathbf{X}_s(m-1)$ with $\mathbf{X}_s(m) + \Delta_1$ and $\mathbf{X}_s(m+1)$ with $\mathbf{X}_s(m) - \Delta_2$, based on the sorted coefficients, $\mathbf{X_s}(m) + \Delta_1 \geq \mathbf{X_s}(m) \geq \mathbf{X_s}(m) - \Delta_2$.

$$\Delta\mathbf{X_s}(m) = \Big\lfloor \frac{\mathbf{X_s}(m-1) + \mathbf{X_s}(m+1)}{2} \Big\rfloor - \mathbf{X_s}(m),$$

$$\Delta\mathbf{X_s}(m) = \Big\lfloor \frac{\mathbf{X_s}(m) + \Delta_1 + \mathbf{X_s}(m) - \Delta_2}{2} \Big\rfloor - \mathbf{X_s}(m).$$

The minimum error distortion is obtained when the difference between $\mathbf{X}_{s_w}(m)$ and $\mathbf{X}_s(m)$ is close to 0:

$$\Big\lfloor \frac{\mathbf{X_s}(m-1) + \mathbf{X_s}(m+1)}{2} \Big\rfloor - \mathbf{X_s}(m) = 0,$$

$$\Big\lfloor \frac{\mathbf{X_s}(m) + \Delta_1 + \mathbf{X_s}(m) - \Delta_2}{2} \Big\rfloor - \mathbf{X_s}(m) = 0,$$

$$\mathbf{X_s}(m) + \Delta_1 + \mathbf{X_s}(m) - \Delta_2 = 2\mathbf{X_s}(m),$$

$$(\mathbf{X_s}(m) + \Delta_1) - \mathbf{X_s}(m) = \mathbf{X_s}(m) - (\mathbf{X_s}(m) - \Delta_2),$$

$$[(\mathbf{X_s}(m) + \Delta_1) - \mathbf{X_s}(m)] - [\mathbf{X_s}(m) - (\mathbf{X_s}(m) - \Delta_2)] = 0.$$

Since the GFT is orthonormal and from Eq. (3.9) we obtain:

$$\mu \propto \sum |\Delta \mathbf{X_s}(m)|^2. \tag{3.12}$$

Thereby leading to

$$\mu \propto \sum (\Big\lfloor \frac{\mathbf{X_s}(m-1) + \mathbf{X_s}(m+1)}{2} \Big\rfloor - \mathbf{X_s}(m))^2.$$

Thereby leading to the relationship between the MSE μ and the selected GFT coefficient triple:

$$\mu \propto (\mathbf{X_s}(m-1) - \mathbf{X_s}(m)) - (\mathbf{X_s}(m) - \mathbf{X_s}(m+1))^2.$$

Therefore for minimising μ, for each hiding coefficient triple, $\lfloor 0.5(\mathbf{X_s}(m-1) + \mathbf{X_s}(m+1)) \rfloor - \mathbf{X_s}(m)$ have to be close to 0 or in other words the gradient difference, $[(\mathbf{X_s}(m-1) - \mathbf{X_s}(m)) - (\mathbf{X_s}(m) - \mathbf{X_s}(m+1))]^2$ have to be close to 0 (when the gradient difference is close to zero this means the $\Delta \mathbf{X_s}(m)$ is close to zero and this leads to reducing the MSE because the modified coefficient is very close to the original coefficient).

□

3.2.5 On enhancing robustness

Tthe robustness model is proposed to enhance the robustness of the data hiding against the attacks. The main idea to the proposed model is to find the GFT coefficients which are able to retain the secret bit after the attack based on establishing the relationship between the extraction of the secret bits and the effect of the attacks. Two data hiding scenarios are proposed to analyse the robustness against the attacks in graph Fourier domain: non-blind and blind algorithms. We consider two attacks types namely, noise addition and deleting nodes data on graphs. The modified graph Fourier coefficients $\mathbf{X}_w(\ell)$ are modified based on the modification value due to attack Δ_a as the following:

$$\mathbf{X}'_w(\ell') = \mathbf{X}_w(\ell) + \Delta_a, \tag{3.13}$$

where $\mathbf{X}'_w(\ell')$ are the modified graph Fourier coefficients after the attack. The value of modification due to attack Δ_a can be in the range:

$$\Delta_{a_{min}} \leq \Delta_a \leq \Delta_{a_{max}}, \tag{3.14}$$

where $\Delta_{a_{min}}$ and $\Delta_{a_{max}}$ are the minimum and maximum modification values. The value of modification Δ_a depends on the attack type. For instance, the value of modification due to additive noise depends on the value of the noise variance (σ^2), while the value of modification of deleting nodes data depends on the number of the node data that are deleting and their locations in the graph. We propose the robustness models based on adopting the robustness models which are proposed in discrete wavelet transform [1].

3.2.5.1 The non-blind model

A model is derived to find the GFT coefficients can retain the secret bits after the attacks. We establish the relationship between the selected coefficients to hide the secret bits and the robustness against attacks. The basic form of the data embedding in the non-blind approach is:

$$\mathbf{X}_w(\ell) = \mathbf{X}(\ell) + \Delta, \tag{3.15}$$

where $\mathbf{X}(\ell)$ is the GFT coefficient to be modified, $\mathbf{X}_w(\ell)$ is the modified coefficient and Δ is the modification value due to data hiding.

$$\Delta = \mathbf{X}_w(\ell) - \mathbf{X}(\ell), \tag{3.16}$$

$$\Delta = \alpha \mathbf{X}(\ell) w_b, \tag{3.17}$$

where α is the data hiding parameter and $w_b (b \in \{0, 1\})$ is the secret bit. Based on substituting the Eq. (3.16) in Eq. (3.15), we obtain:

$$\mathbf{X}_w(\ell) = \mathbf{X}(\ell) + \alpha \mathbf{X}(\ell) w_b. \tag{3.18}$$

$$= \mathbf{X}(\ell)(1 + \alpha w_b).$$

The relationship between the original graph Fourier coefficient and modified graph Fourier coefficient is:

$$\mathbf{X}(\ell) = \frac{\mathbf{X}_w(\ell)}{1 + \alpha w_b}. \tag{3.19}$$

The secret bit w_b' is extracted based on the following equation:

$$w_b' = \frac{\mathbf{X}_w(\ell) - \mathbf{X}(\ell)}{\alpha \mathbf{X}(\ell)}. \tag{3.20}$$

At this point, we consider three cases of the secret bits: hiding only $b = 0$ bit, hiding only $b = 1$ bit and hiding $b = 0$ and $b = 1$ bit.

Proposition 3.3

The original GFT coefficients for hiding a bit with value $b = 1$ and retain intact after the attacks are in the range:

$$\frac{\mathbf{X}_w'(\ell')}{1 + \alpha w_1} \leq \mathbf{X}(\ell) \leq \frac{\mathbf{X}_w'(\ell')}{1 + \alpha T}. \tag{3.21}$$

Proof. To obtain the embedded bit $b = 1$, we need to get $w_b' \geq T$:

$$\frac{\mathbf{X}_w(\ell) - \mathbf{X}(\ell)}{\alpha \mathbf{X}(\ell)} \geq T. \tag{3.22}$$

Since $\mathbf{X}_w(\ell)$ and $\mathbf{X}(\ell)$ have the same sign and $|\mathbf{X}_w(\ell)| > |\mathbf{X}(\ell)|$ (the GFT coefficient can retain secret bit accurately), then

$$\mathbf{X}_w(\ell) \geq \mathbf{X}(\ell)(1 + \alpha T).$$

In the case of no attack, the modified coefficient $\mathbf{X}_w(\ell)$ after embedding the secret bit $w_b = 1$ will be in the range:

$$\mathbf{X}(\ell)(1 + \alpha T) \leq \mathbf{X}_w(\ell) \leq \mathbf{X}(\ell)(1 + \alpha w_1). \tag{3.23}$$

The secret bit can extract accurately when the GFT coefficients in the range:

$$\frac{\mathbf{X}_w(\ell)}{1 + \alpha w_1} \leq \mathbf{X}(\ell) \leq \frac{\mathbf{X}_w(\ell)}{1 + \alpha T}. \tag{3.24}$$

To extract the secret bit correctly after the attack, we need:

$$\mathbf{X}'_w(\ell') \;\geq\; \mathbf{X}_w(\ell). \tag{3.25}$$

After the attack, the modified coefficients will be in the region:

$$\mathbf{X}_w(\ell) + \Delta_{a_{min}} \leq \mathbf{X}_w(\ell) \leq \mathbf{X}_w(\ell) + \Delta_{a_{max}}. \tag{3.26}$$

By considering the values in this region:

$$\begin{aligned}
&\mathbf{X}_w(\ell) + \Delta_{a_{min}} \leq \mathbf{X}_w(\ell) \leq \mathbf{X}_w(\ell) + \Delta_{a_{max}},\\
&\mathbf{X}'_w(\ell') = \mathbf{X}_w(\ell) + \Delta_{a_{max}},\\
&\Rightarrow\; \mathbf{X}'_w(\ell') > \mathbf{X}_w(\ell).
\end{aligned}$$

In terms of the original coefficients, $\mathbf{X}(\ell)$ is:

$$\mathbf{X}_w(\ell) + \Delta_{a_{min}} \;\leq\; \mathbf{X}(\ell)(1 + \alpha w_1) \leq\; \mathbf{X}_w(\ell) + \Delta_{a_{max}},$$

$$\frac{\mathbf{X}_w(\ell) + \Delta_{a_{min}}}{1 + \alpha w_1} \leq \mathbf{X}(\ell) \leq \frac{\mathbf{X}_w(\ell) + \Delta_{a_{max}}}{1 + \alpha w_1}.$$

By substituting $(\mathbf{X}_w(\ell) + \Delta_{a_{min}})$ and $(\mathbf{X}_w(\ell) + \Delta_{a_{max}})$ with $\mathbf{X}'_w(\ell')$ we get:

$$\frac{\mathbf{X}'_w(\ell')}{1 + \alpha w_1} \leq \mathbf{X}(\ell) \leq \frac{\mathbf{X}'_w(\ell')}{1 + \alpha w_1}.$$

Since $w_1 = 1$, $w_1 > T$, therefore, $\dfrac{1}{1 + \alpha T} \;>\; \dfrac{1}{1 + \alpha w_1}$, then we get the range of the graph Fourier coefficients which can retain the secret bits after the attack as:

$$\frac{\mathbf{X}'_w(\ell')}{1 + \alpha w_1} \leq \mathbf{X}(\ell) \leq \frac{\mathbf{X}'_w(\ell')}{1 + \alpha T}.$$

□

Proposition 3.4

The original GFT coefficients for hiding a bit value $b = 0$ and retain intact after the attacks are in the range

$$\frac{\mathbf{X}'_w(\ell')}{1+\alpha T} < \mathbf{X}(\ell) < \frac{\mathbf{X}'_w(\ell')}{1+\alpha w_0}. \tag{3.27}$$

Proof. To obtain the embedded bit $b = 0$, we need to get $w'_b < T$ as:

$$\frac{\mathbf{X}_w(\ell) - \mathbf{X}(\ell)}{\alpha \mathbf{X}(\ell)} < T, \tag{3.28}$$

$$\mathbf{X}_w(\ell) < \mathbf{X}(\ell)(1+\alpha T).$$

In the case of no attack, the modified coefficient $\mathbf{X}_w(\ell)$ after embedding the secret bit $w'_b = 0$ will be in the range:

$$\mathbf{X}(\ell)(1+\alpha w_0) \ < \mathbf{X}_w(\ell) \ < \mathbf{X}(\ell)(1+\alpha T). \tag{3.29}$$

And the secret bit can extract accurately when the GFT coefficients in the range:

$$\frac{\mathbf{X}_w(\ell)}{1+\alpha T} < \mathbf{X}(\ell) < \frac{\mathbf{X}_w(\ell)}{1+\alpha w_0}. \tag{3.30}$$

To obtain a correct extraction of the embedded bits after attack, we need:

$$\mathbf{X}'_w(\ell') < \mathbf{X}_w(\ell). \tag{3.31}$$

After the attack, the modified coefficients will be in the region:

$$\mathbf{X}_w(\ell) + \Delta_{a_{min}} \leq \mathbf{X}_w(\ell) \leq \mathbf{X}_w(\ell) + \Delta_{a_{max}}. \tag{3.32}$$

By considering the values in the range:

$$\mathbf{X}_w(\ell) + \Delta_{a_{min}} \ \leq \ \mathbf{X}_w(\ell) \leq \ \mathbf{X}_w(\ell) + \Delta_{a_{max}}.$$

In terms of the original coefficients, $\mathbf{X}(\ell)$:

$$\mathbf{X}_w(\ell) + \Delta_{a_{min}} < \mathbf{X}(\ell)(1 + \alpha w_0) < \mathbf{X}_w(\ell) + \Delta_{a_{max}},$$

$$\frac{\mathbf{X}_w(\ell) + \Delta_{a_{min}}}{1 + \alpha w_0} < \mathbf{X}(\ell) < \frac{\mathbf{X}_w(\ell) + \Delta_{a_{max}}}{1 + \alpha w_0}.$$

By substituting $(\mathbf{X}_w(\ell) + \Delta_{a_{min}})$ and $(\mathbf{X}_w(\ell) + \Delta_{a_{max}})$ with $\mathbf{X}'_w(\ell')$ we get:

$$\frac{\mathbf{X}'_w(\ell')}{1 + \alpha w_0} < \mathbf{X}(\ell) < \frac{\mathbf{X}'_w(\ell')}{1 + \alpha w_0}.$$

Since $w_0 = 0$, $w_0 < T$, therefore, $\frac{1}{1 + \alpha T} < \frac{1}{1 + \alpha w_0}$, and by substituting $\mathbf{X}_w(\ell) + \Delta_{a_{min}}$ and $\mathbf{X}_w(\ell) + \Delta_{a_{max}}$ with $\mathbf{X}'_w(\ell')$, then we get the range of the original GFT coefficients which can retain the correct secret bit after the attack:

$$\frac{\mathbf{X}'_w(\ell')}{1 + \alpha T} < \mathbf{X}(\ell) < \frac{\mathbf{X}'_w(\ell')}{1 + \alpha w_0}. \qquad \square$$

Finally, we have to combine the proposition 3.3 and proposition 3.4 to derive the region of the GFT coefficients that are capable of retaining both $b = 1$ and $b = 0$ after the attacks. The original GFT coefficients which can retain the correct secret bit after the attack should be in the range:

$$\frac{\mathbf{X}'_w(\ell')}{1 + \alpha w_1} \leq \mathbf{X}(\ell) \leq \frac{\mathbf{X}'_w(\ell')}{1 + \alpha w_0}. \tag{3.33}$$

In the case of no attack, the modified coefficients $\mathbf{X}_w(\ell)$ after embedding the secret bit $b = 0$ and $b = 1$ will be in the range:

$$\mathbf{X}(\ell)(1 + \alpha w_0) \leq \mathbf{X}_w(\ell) \leq \mathbf{X}(\ell)(1 + \alpha w_1). \tag{3.34}$$

And the secret bit can be extracted accurately when the GFT coefficients in the range:

$$\frac{\mathbf{X}_w(\ell)}{1 + \alpha w_1} \leq \mathbf{X}(\ell) \leq \frac{\mathbf{X}_w(\ell)}{1 + \alpha w_0}. \tag{3.35}$$

Figure 3.2 displays the GFT coefficients range which is able to keep the secret bits after the attacks.

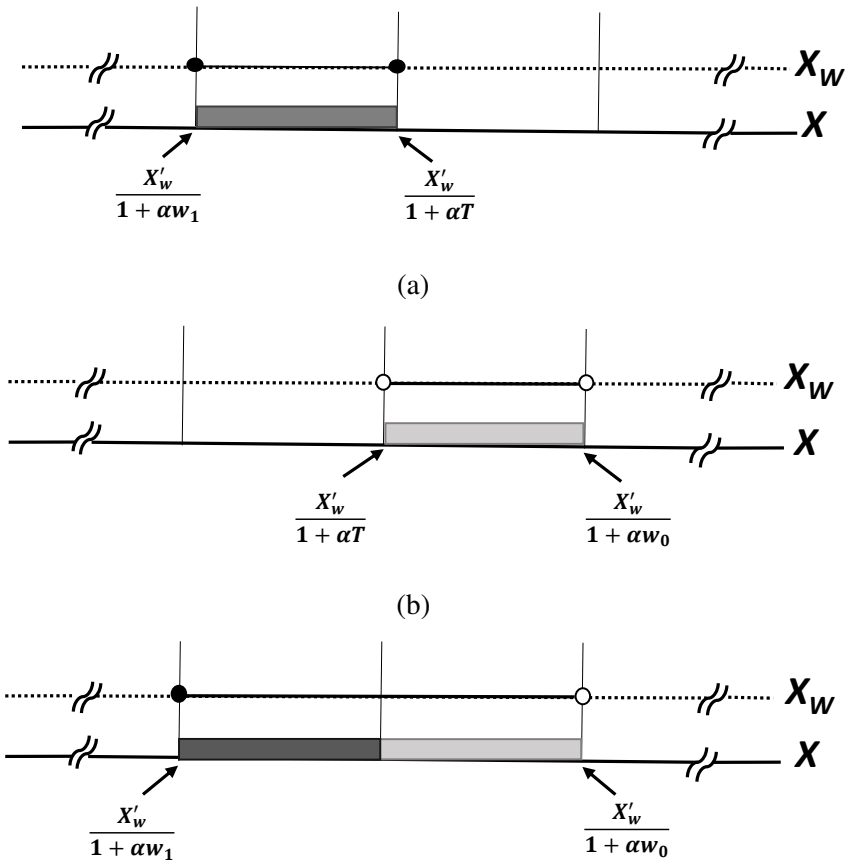


Figure 3.2: The GFT coefficients range which is able to extract the secret bits correctly. (a) Hiding only $b = 1$. (b) Hiding only $b = 0$. (c) Hiding $b = 0$ and $b = 1$.

3.2.5.2 The Blind model

A new model is proposed to identify the GFT coefficients that are able to keep the secret bits after the attack in the graph Fourier domain for a blind approach using a prediction-based graph data hiding. The modified coefficients are given as:

$$\mathbf{X}_w(\ell) = \left\lfloor \frac{\mathbf{X}(\ell-1) + \mathbf{X}(\ell+1)}{2} \right\rfloor + w_b. \tag{3.36}$$

After the attack, to extract the secret data w'_b, we have new graph Fourier coefficients values $\mathbf{X}'_w(\ell'-1), \mathbf{X}'_w(\ell')$ and $\mathbf{X}'_w(\ell'+1)$:

$$w'_b = \mathbf{X}'_w(\ell') - \left\lfloor \frac{\mathbf{X}'_w(\ell'-1) + \mathbf{X}'_w(\ell'+1)}{2} \right\rfloor. \tag{3.37}$$

We consider three cases of the secret bits: embedding only $b = 0$ bit, embedding only $b = 1$ bit and embedding $b = 0$ and $b = 1$ bit.

Proposition 3.5

The original GFT coefficients for hiding a bit value $b = 1$ and retain intact after the attacks are in the range:

$$\left\lfloor \frac{X'_w(\ell'-1) + X'_w(\ell'+1)}{2} \right\rfloor + T \leq X'_w(\ell') < \left\lfloor \frac{X'_w(\ell'-1) + X'_w(\ell'+1)}{2} \right\rfloor + w_1. \quad (3.38)$$

Proof. To obtain the secret bit $b = 1$, we need to get $w'_b \geq T$, that means:

$$\mathbf{X}_w(\ell) - \left\lfloor \frac{\mathbf{X}_w(\ell-1) + \mathbf{X}_w(\ell+1)}{2} \right\rfloor \geq T. \quad (3.39)$$

Since $|\mathbf{X}_w(\ell)| > |\mathbf{X}(\ell)|$, then

$$\mathbf{X}_w(\ell) \geq \left\lfloor \frac{\mathbf{X}_w(\ell-1) + \mathbf{X}_w(\ell+1)}{2} \right\rfloor + T. \quad (3.40)$$

In the case of no attack, the modified coefficient $\mathbf{X}_w(\ell)$ after embedding the secret bit $w'_b = 1$ will be in the range:

$$\left\lfloor \frac{X_w(\ell-1) + X_w(\ell+1)}{2} \right\rfloor + T \leq X_w(\ell) < \left\lfloor \frac{X_w(\ell-1) + X_w(\ell+1)}{2} \right\rfloor + w_1. \quad (3.41)$$

After the attack, we have only the reconstructed coefficients, $\mathbf{X}'_w(\ell')$. For correct extraction of the secret bit, we need:

$$\mathbf{X}'_w(\ell') \geq \mathbf{X}_w(\ell). \quad (3.42)$$

The modified coefficients after the attack will be in the region:

$$\mathbf{X}'_w(\ell') = \mathbf{X}_w(\ell) + \Delta_a, \quad (3.43)$$

where $\mathbf{X}_w(\ell) + \Delta_a$ are the modified coefficients after the attack .

$$\mathbf{X}_w(\ell) + \Delta_a \geq \mathbf{X}_w(\ell), \tag{3.44}$$

$$\left\lfloor \frac{\mathbf{X}_w(\ell-1) + \mathbf{X}_w(\ell+1)}{2} \right\rfloor + \Delta_a + T \geq \left\lfloor \frac{\mathbf{X}_w(\ell-1) + \mathbf{X}_w(\ell+1)}{2} \right\rfloor + T. \tag{3.45}$$

Since $w_1 = 1$, $w_1 > T$, from Eq. (3.36) and by considering the modified coefficients after the attack we get:

$$\left\lfloor \frac{X'_w(\ell'-1) + X'_w(\ell'+1)}{2} \right\rfloor + T \leq X'_w(\ell') < \left\lfloor \frac{X'_w(\ell'-1) + X'_w(\ell'+1)}{2} \right\rfloor + w_1.$$

□

Proposition 3.6

The original GFT coefficients for hiding a bit value $b = 0$ and retain intact after the attacks are in the range:

$$\left\lfloor \frac{X'_w(\ell'-1) + X'_w(\ell'+1)}{2} \right\rfloor + w_0 \leq X'_w(\ell') < \left\lfloor \frac{X'_w(\ell'-1) + X'_w(\ell'+1)}{2} \right\rfloor + T. \tag{3.46}$$

Proof. To obtain the secret bit $b = 0$, we need to get $w'_b < T$, which means:

$$\mathbf{X}_w(\ell) - \left\lfloor \frac{\mathbf{X}_w(\ell-1) + \mathbf{X}_w(\ell+1)}{2} \right\rfloor < T, \tag{3.47}$$

$$\mathbf{X}_w(\ell) < \left\lfloor \frac{\mathbf{X}_w(\ell-1) + \mathbf{X}_w(\ell+1)}{2} \right\rfloor + T. \tag{3.48}$$

In the case of no attack, the modified coefficient $\mathbf{X}_w(\ell)$ after embedding the secret bit $b = 0$ will be in the range:

$$\left\lfloor \frac{X_w(\ell-1) + X_w(\ell+1)}{2} \right\rfloor + w_0 \leq X_w(\ell) < \left\lfloor \frac{X_w(\ell-1) + X_w(\ell+1)}{2} \right\rfloor + T. \tag{3.49}$$

After the attack, we have only the reconstructed coefficients, $\mathbf{X}'_w$. For correct extraction of the secret bit, we need:

$$\mathbf{X}'_w(\ell') < \mathbf{X}_w(\ell). \tag{3.50}$$

The modified coefficients after the attack will be:

$$\mathbf{X}'_w(\ell') = \mathbf{X}_w(\ell) + \Delta_a, \tag{3.51}$$

where $\mathbf{X}_w(\ell) + \Delta_a$ are the modified coefficients after the attack.

$$\mathbf{X}_w(\ell) + \Delta_a \leq \mathbf{X}_w(\ell), \tag{3.52}$$

$$\left\lfloor \frac{\mathbf{X}_w(\ell-1) + \mathbf{X}_w(\ell+1)}{2} \right\rfloor + \Delta_a + T \geq \left\lfloor \frac{\mathbf{X}_w(\ell-1) + \mathbf{X}_w(\ell+1)}{2} \right\rfloor + w_0. \tag{3.53}$$

Since $w_0 = 0$, $w_0 < T$, From Eq. (3.36) and by considering the modified coefficients after the attack, we get:

$$\left\lfloor \frac{X'_w(\ell'-1) + X'_w(\ell'+1)}{2} \right\rfloor + w_0 \leq X'_w(\ell') < \left\lfloor \frac{X'_w(\ell'-1) + X'_w(\ell'+1)}{2} \right\rfloor + T.$$

□

We combine the above propositions to identify the region of GFT coefficients that are capable of retaining both $b = 1$ and $b = 0$ after the attacks. The GFT coefficients range that can keep the secret bits is:

$$\left\lfloor \frac{X'_w(\ell'-1) + X'_w(\ell'+1)}{2} \right\rfloor + w_0 \leq X'_w(\ell') < \left\lfloor \frac{X'_w(\ell'-1) + X'_w(\ell'+1)}{2} \right\rfloor + w_1. \tag{3.54}$$

Figure 3.3 displays the GFT coefficients range which is able to retain the secret bits after the attacks.

3.2.6 Joint robust-low distortion data hiding

The proposed models, embedding distortion minimisation and robustness are combined for satisfying the basic requirements of the graph data hiding. Two embedding algorithms are considered. For the non-blind algorithm, in order to combine the model of the embedding distortion minimisation with the robustness model, we select the GFT coefficients that satisfy the condition in Eq. (3.33) for satisfying the robustness condition, then, we select the GFT coefficients

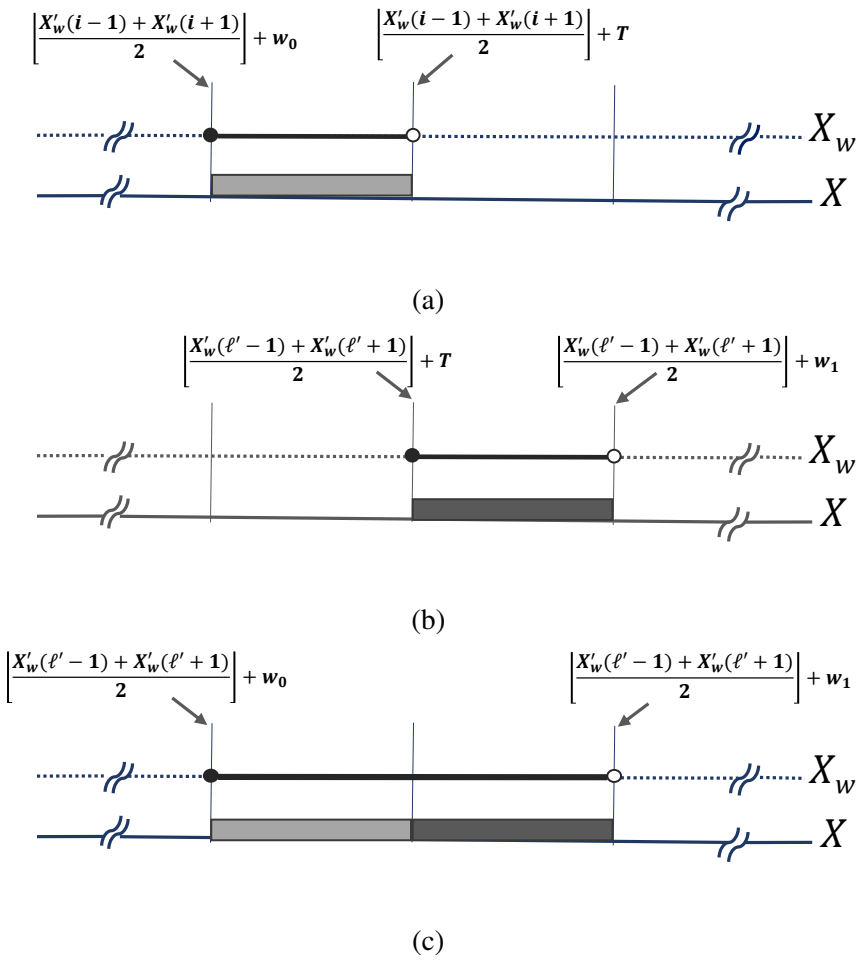


Figure 3.3: The range of the graph Fourier coefficients which is able to extract the secret bits correctly. (a) Hiding only $b = 0$. (b) Hiding only $b = 1$. (c) Hiding $b = 0$ and $b = 1$.

(from the above GFT coefficients) that satisfy the condition in Eq. (3.10) (in other words, the GFT coefficients have the lowest values) to minimise the embedding distortion i.e., the GFT coefficients which satisfy the above two conditions are selected to embed the secret data. For the blind algorithm, we combine the proposed models of the embedding distortion minimisation and the robustness based on selecting the GFT coefficients that satisfy the condition in Eq. (3.54) to meet the robustness, then, we select the GFT coefficients which satisfy the condition in Eq. (3.11) (in our model the GFT coefficient triple which has the gradient difference close to 0) in order to minimise the embedding distortion i.e., the GFT coefficients that satisfy the both conditions are chosen for embedding the secret bits.

3.3 Performance evaluation

The experimental simulations verify the proposed models using two data hiding scenarios: non-blind and blind. The experimental simulations are divided into two types: evaluation of the performance of the embedding distortion and evaluation of the robustness performance. The proposed models are evaluated by comparing the performance of the data hiding methods with using the proposed models (embedding distortion minimisation and robustness) and the data hiding methods without using the proposed models. We would like to indicate that we have identified many limitations regarding the comparison of the proposed methods with previous methods. We could not find a method where a framework similar to the proposed method was considered. None of the existing graph data hiding methods embed the secret bits into the graph signal; instead, they embed the secret data in the mesh coordinates or the graph topology. Due to lack of any other comparable work, it is not possible to compare our experimental results with other works. In that context, we consider the results without the proposed models as the baseline. Therefore, we calculate the results by using the data hiding algorithms without using the proposed models (embedding distortion minimisation and robustness) to show improvements when the proposed models were applied using the same data hiding algorithms.

3.3.1 Experimental set up

The proposed GFT domain data hiding algorithms with the proposed models, namely, embedding distortion minimisation and robustness were tested using the dataset of graph watermarking [181]. This dataset includes 11 types of graphs: Sensor, Spiral, Swiss-roll, Sphere, Minnesota, Community, Cube, Torus, David-sensor-network, Air foil and Bunny. The graphs (with specified connectivity) were generated using the Toolbox for signal processing on graphs (GSPBox) [182]. The Toolbox for GSPBox provides the graph structure without graph signal values. This dataset incorporates the graph signals, using a correlated input, such as, a natural image. The graph data (signal) values are obtained from 5 standard test images data: Lena, Barbara, Gold Hill, Baboon and Peppers to form graph signals: $signal\ 1$, $signal\ 2$, $signal\ 3$, $signal\ 4$ and $signal\ 5$, respectively. We considered these images because they are the most common standard images and used in data hiding methods. The dataset includes a total of 160

various graphs with a different number of nodes and various graph signals. In this dataset, four sets of graphs are considered according to the number of the graph nodes in each set, $N = \{500,\ 2500,\ 5000,\ 10000\}$ nodes, where N is the number of the nodes in each graph and each set has more than 35 different types of graphs. All the graphs have the same structure including five fields: $\mathcal{G} = \{N,\ Coordinates,\ Type, \mathbf{A},\ \mathbf{x}\}$, where N is the number of the graph nodes, $Coordinates$ are the graph coordinates, $Type$ is the graph type, $\mathbf{A}$ is the adjacency matrix, $\mathbf{x}$ is the graph signal. The length of the signal depends on the number of the graph nodes, for example, when $N = 2500$ nodes, the graph signal is generated from 2-D image data with (50×50) pixels, then converted to 1-D signal, row by row from left to right. Figure 3.4 shows the types of graphs that are used in the graph dataset with and without the graphs edges.

The reasons of using images data are: first, the majority of the data hiding methods are based on images. Also, we can compare the proposed methods with these methods. Second, due to lack of datasets using the environmental conditions; in other words, graph dataset that provides the graph structure with the graph signal in order to use it. In addition, due to lack of other comparable work that use the environmental data.

3.3.2 Evaluation of the performance of the embedding distortion

This section presents two types of empirical results: verification of the embedding distortion minimisation model and evaluation of the performance of the embedding distortion for non-blind and blind data hiding. For performance metrics, MSE of the modified graphs were calculated for evaluation of the embedding distortion. We chose MSE in our performance evaluation of the proposed methods instead of using the Hausdorff distance for two reasons. First, MSE calculates the differences between the graph signal values in the original and the modified graphs. Second, MSE is usually employed for evaluating changes in all values of the graph signal as in changes in the colour intensities assigned to image samples (pixels of a two-dimensional image or points in a 3D cloud). While the Hausdorff distance is used for evaluating distances in a Euclidean space (Changing the coordinates of the graph or the points in a 3D cloud).

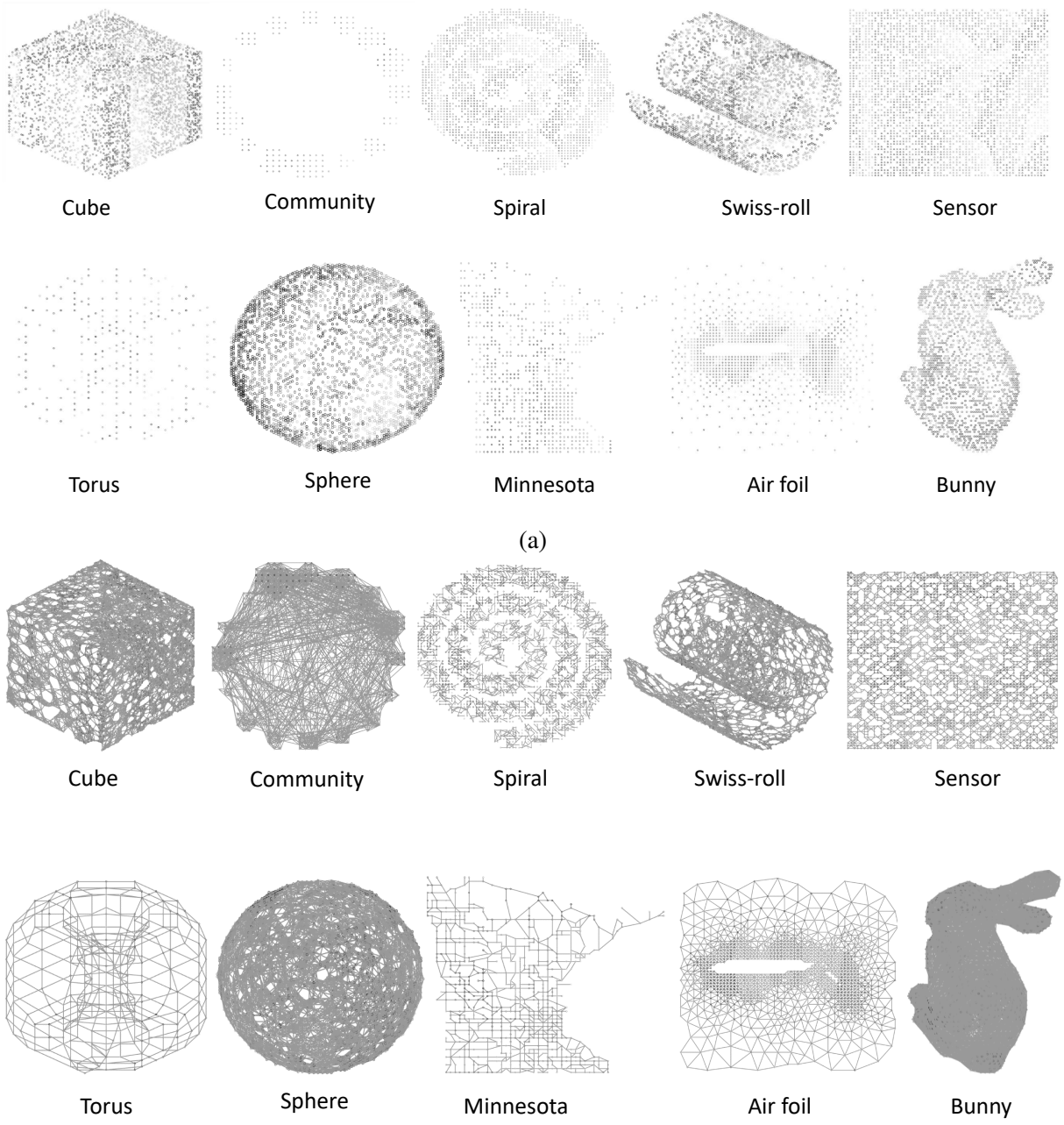


Figure 3.4: Graph dataset. (a) Graphs types. (b) Graph types with edges.

3.3.2.1 Verification of the embedding distortion minimisation model of the non-blind data hiding

The proposition 3.1 is verified in the empirical simulations. The energy sum of the chosen GFT coefficients and the MSE of the modified graphs are calculated using graph dataset. We

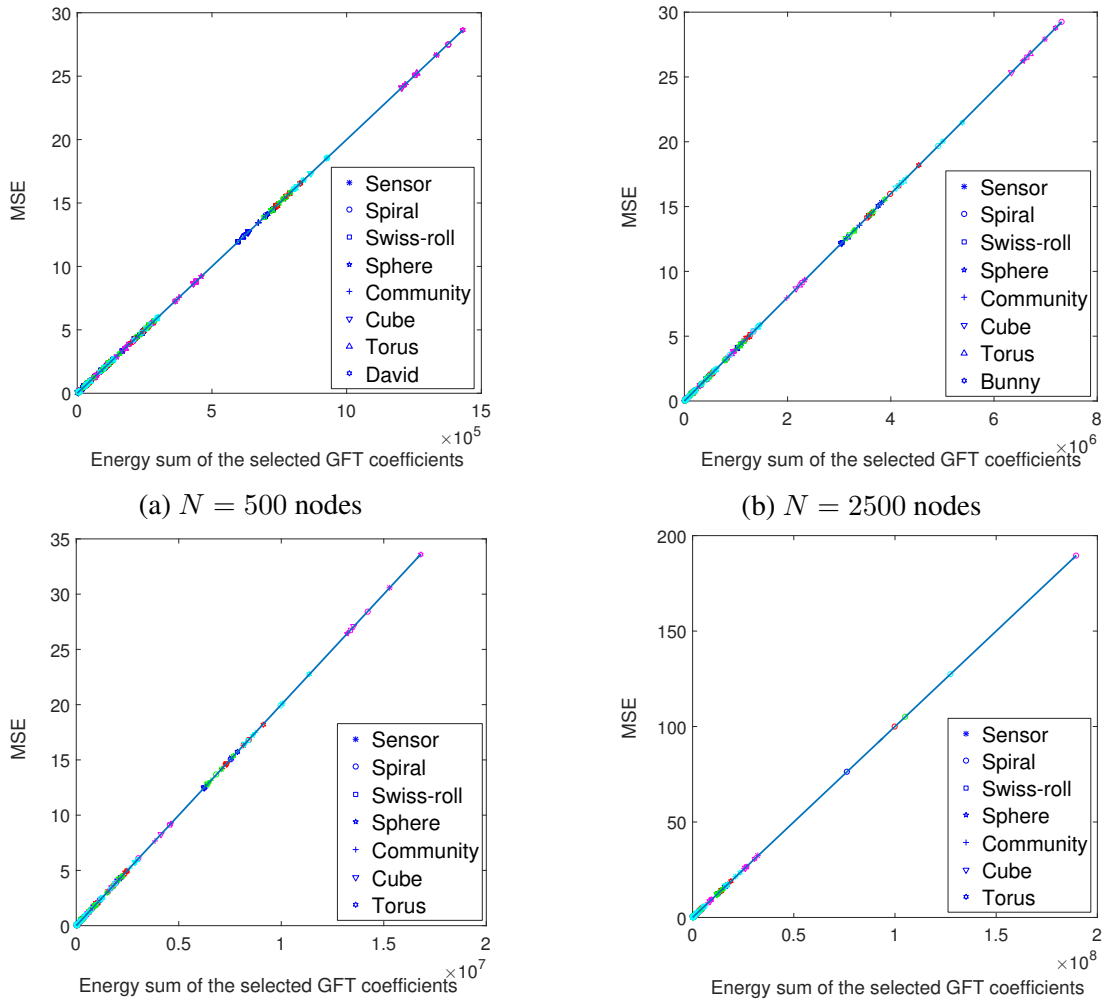
consider pseudo-random binary sequences as the secret bits, with three scenarios ($\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0, 1\}$), to hide in the graph Fourier coefficients for various graphs. In these experiments, 8 graphs types with a different number of graph nodes ($500,\ 2500,\ 5000,\ 10000$) and five graph signals are used to embed the same number of secret bits per group. We use different colors to represent each signal to display different types of graphs for various signals, where blue colour for graph signal 1, red colour for graph signal 2 , magenta colour for graph signal 3 , green colour for graph signal 4 and cyan colour for graph signal 5. We obtain two sets of results to verify the effects of embedding three scenarios of the secret bits as given:

In the experiment Set 1, the GFT coefficients are divided into five groups based on their values by taking into consideration all the GFT coefficients which are corresponding to the eigenvectors from 1 to $N - 1$ except the first coefficient corresponding to the eigenvector 0. We embed the same number of the secret bits in each group. For example, if we have a graph with 500 nodes, we divide its coefficients into 5 groups, each group has 100 coefficients. Then, we embed 100 secret bits in each group. After that, the energy sum of the chosen GFT coefficients to be modified and MSE of the modified graphs are calculated using the same $\alpha = 0.1$ for all groups separately. In these experiments, we consider the case when the secret bits $\mathbf{w} = \{1\}$.

In the experiment Set 2, we have considered the case when the secret bits $\mathbf{w} = \{0, 1\}$, where the numbers of 0s and 1s are equal in $\mathbf{w}$. Embedding performance is calculated in a similar way to that mentioned in experiment Set 1 to notice the trend.

We can observe that the distortion in the experiment Set 1 is double the distortion of the experiment Set 2 due to embedding the double number of 1s as illustrated in Figure 3.5 and Figure 3.6 respectively. While there is no distortion in case of embedding $\mathbf{w} = \{0\}$ only.

The empirical results demonstrate that the sum of the energy of the GFT coefficients selected for modification correlates strongly with the MSE of the modified graph. The relationship between the MSE of the modified graph and the energy sum of the chosen graph Fourier coefficients is a linear proportionality (where $y = m_1 x + \beta$,m_1 is the slope of the graph, and β is the y-intercept; in the proposed model, y-axis = MSE and x-axis = the energy sum of the selected coefficients). This proves that the MSE of the modified graph is linearly proportional to the energy sum of the selected GFT coefficients. The proposed model was supported by simulation results for the graph dataset.



(a) $N = 500$ nodes

(b) $N = 2500$ nodes

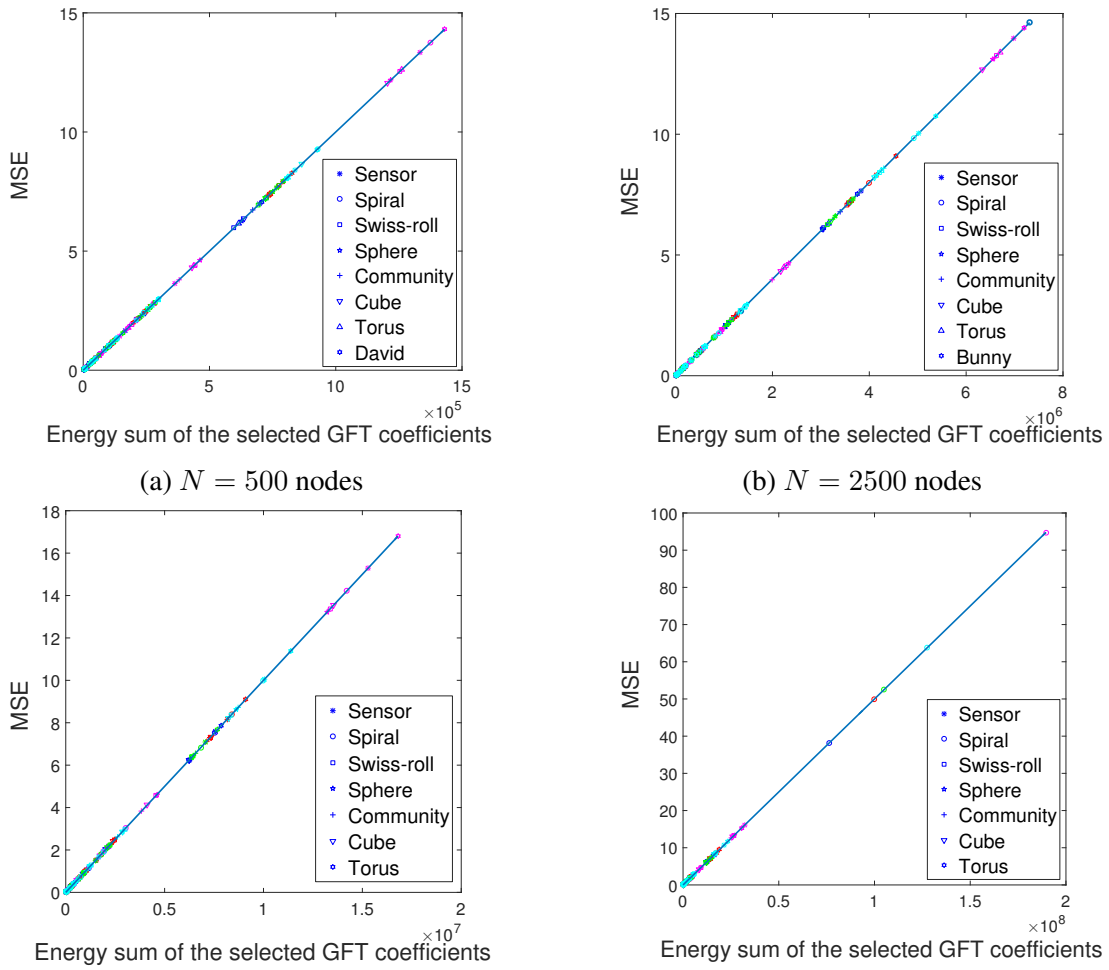
(c) $N = 5000$ nodes

(d) $N = 10000$ nodes

Figure 3.5: Verification of embedding distortion of non-blind data hiding: MSE of the modified graph vs. sum of energy when $\mathbf{w} = \{1\}$, for individual graphs with different number of nodes $N = 500,\ 2500,\ 5000,\ 10000$, respectively for 5 graph signals where the colours, blue, red, magenta, green and cyan represent the graph signals $1,\ 2,\ 3,\ 4\ and\ 5$, respectively and the blue line demonstrates the MSE is linearly proportional to the energy sum (where $y = m_1 x + \beta$).

3.3.2.2 Verification of the embedding distortion minimisation model of the blind data hiding

The proposition 3.2 is verified in the simulation results. The MSE of the modified graphs and the gradient differences have been calculated for the test graphs. In these experiments, 4 graph types with a different number of graph nodes $N = \{500,\ 2500,\ 5000,\ 10000\}$. We consider pseudo-random number sequences as the secret data, with five scenarios ($\mathbf{w} = \{0, 0.1, 0.2, 0.3, 0.4\}$), to hide in the GFT coefficients of the graph dataset. Four sets of empirical results are obtained to



(a) $N = 500$ nodes

(b) $N = 2500$ nodes

(c) $N = 5000$ nodes

(d) $N = 10000$ nodes

Figure 3.6: Verification of embedding distortion of non-blind data hiding: MSE of the modified vs. sum of energy when $\mathbf{w} = \{0,\ 1\}$, for individual graphs with different number of nodes $N = 500, 2500, 5000, 10000$, respectively for 5 graph signals where the colours, blue, red, magenta, green and cyan represent the graph signals $1,\ 2,\ 3,\ 4\ and\ 5$, respectively and the blue line demonstrates the MSE is linearly proportional to the energy sum (where $y = m_1 x + \beta$).

verify the effects of embedding five scenarios of the secret bits. In these experimental simulations, the sorted GFT coefficients are divided into nine groups based on based on their gradient difference values by taking into consideration all the GFT coefficients which are corresponding to the eigenvectors from 1 to $N - 1$ except the first coefficient corresponding to the eigenvector 0. We embed the same number of the secret bits in each group. For example, if we have a graph with 500 nodes, we divide its coefficients into 9 groups, each group has 100 coefficients. Then, we embed 50 secret bits in each group. After that, the MSE of the modified graph has been calculated for all groups separately using different embedding scenarios. The empirical results

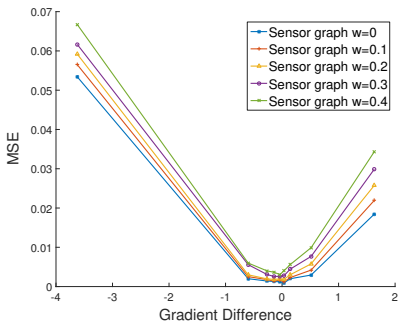
demonstrate that there is a strong correlation between the MSE of the modified graph and the gradient difference of any GFT coefficient triple. It can be observed that the minimum distortion is obtained (low MSE) when the gradient difference is close to zero. The proposed model is supported by the extensive simulation results using the graph dataset and various embedding scenarios as shown in Figure 3.7 and Figure 3.8.

3.3.2.3 Performance evaluation of the embedding distortion of the non-blind data hiding

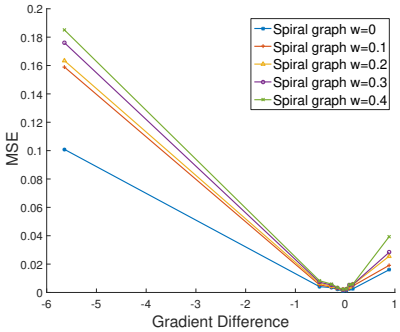
The performance of the embedding distortion of non-blind data hiding is evaluated at various embedding capacities using graph dataset. In these experiments, two sets of graphs $N = 5000$ and $N = 10000$ are utilised to evaluate the performance of the proposed method, where each set of graphs includes 35 different graph types and using the same $\alpha = 0.1$. We consider pseudo-random binary sequences as secret bits, $\mathbf{w} = \{0, 1\}$. MSE of the modified graphs are calculated by using the original non-blind algorithm with the embedding distortion minimisation model by embedding the secret bits in the GFT coefficients that have the lowest values and MSE of the modified graphs are calculated by using the same non-blind algorithm without using the proposed model by embedding the same secret bits in the GFT coefficients which are selected randomly (without considering their values). Figure 3.9 shows the sensor graph with $N = 5000$ nodes before and after embedding the secret bits with length 900 bits. We can notice that the non-blind algorithm with the proposed model provides lower distortion over the original algorithm without the model. As shown in Figure 3.10, the distortion is improved by an average of 99% and 94.5% for $N = 10000$ and $N = 5000$ nodes, respectively. In addition, we can observe that the embedding distortion is increased when the embedding capacity is increased.

3.3.2.4 Performance evaluation of the embedding distortion of the blind data hiding

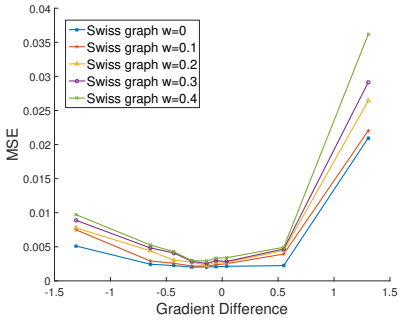
We evaluate the embedding distortion performance of blind data hiding at various embedding capacities using graph dataset. In these experiments, two sets of graphs with $N = 5000$ and $N = 10000$ nodes, respectively are utilised for evaluating the performance of the method, where each set of graphs has 35 different graph types. We consider pseudo-random binary sequences as the secret bits, to represent $\mathbf{w} = \{0, 1\}$. We calculate the MSE of the modified graphs by using the original blind algorithm with the embedding distortion minimisation model by embedding



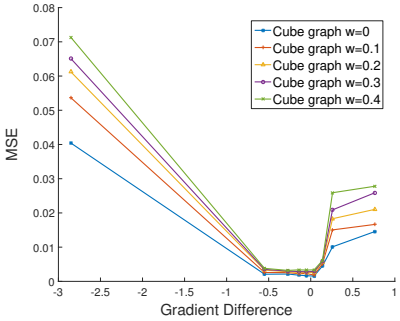
(a) Sensor graph with $N = 500$ nodes



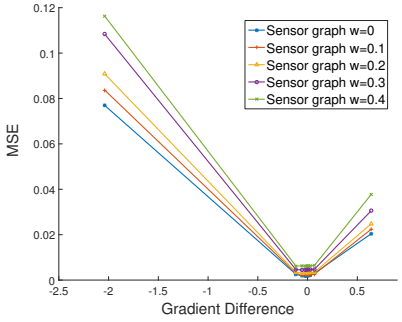
(b) Spiral graph with $N = 500$ nodes



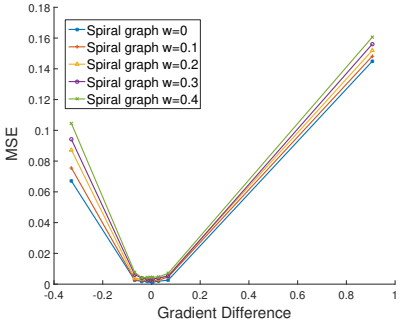
(c) Swiss-roll graph with $N = 500$ nodes



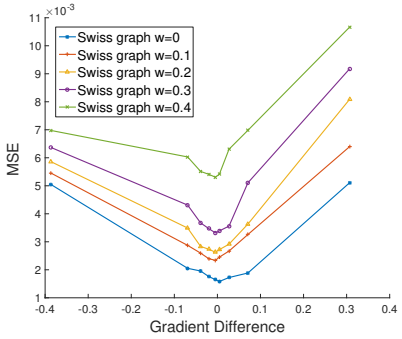
(d) Cube graph with $N = 500$ nodes



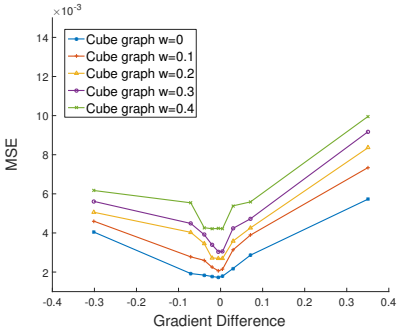
(e) Sensor graph with $N = 2500$ nodes



(f) Spiral graph with $N = 2500$ nodes

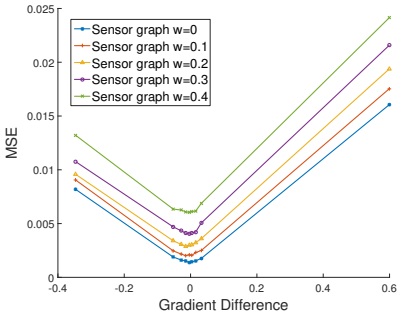


(g) Swiss-roll graph with $N = 2500$ nodes

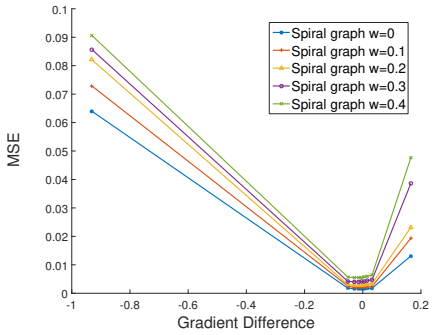


(h) Cube graph with $N = 2500$ nodes

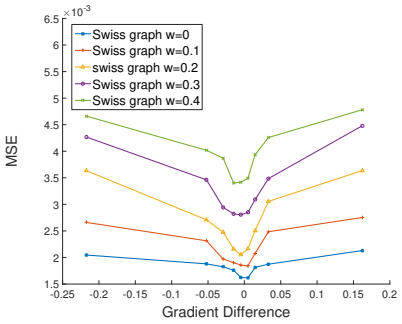
Figure 3.7: Verification of embedding distortion of blind data hiding: MSE of the modified graph vs. gradient difference, for individual graphs with number of nodes $N = 500$ and $N = 2500$ respectively.



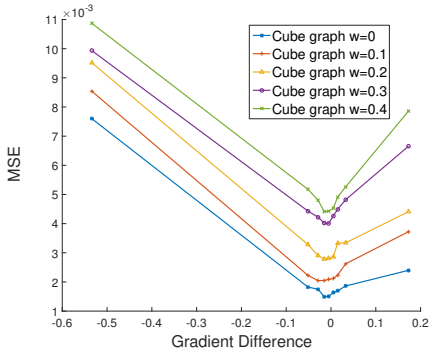
(a) Sensor graph with $N = 5000$ nodes



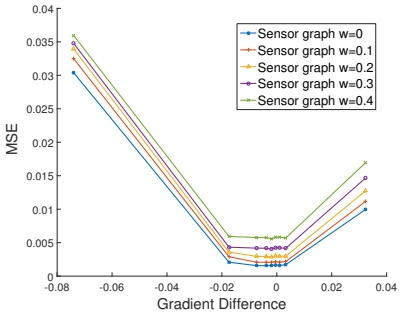
(b) Spiral graph with $N = 5000$ nodes



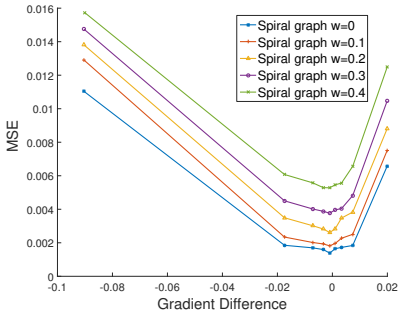
(c) Swiss-roll graph with $N = 5000$ nodes



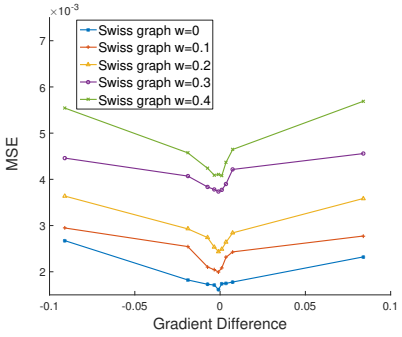
(d) Cube graph with $N = 5000$ nodes



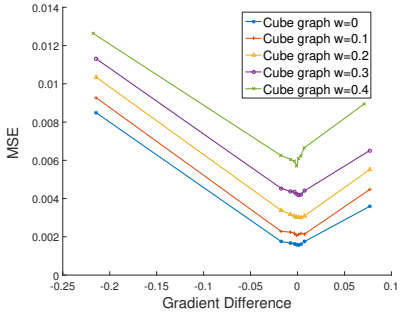
(e) Sensor graph with $N = 10000$ nodes



(f) Spiral graph with $N = 10000$ nodes



(g) Swiss-roll graph with $N = 10000$ nodes

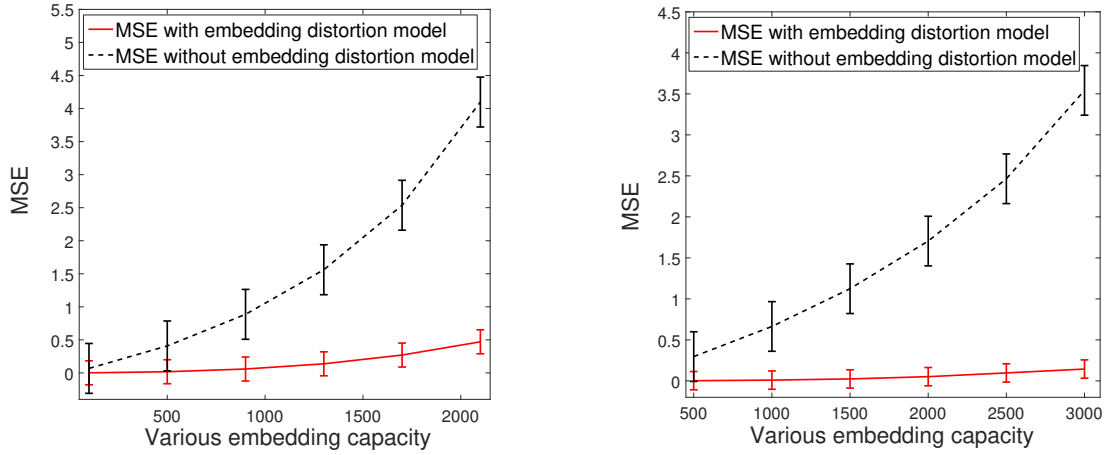


(h) Cube graph with $N = 10000$ nodes

Figure 3.8: Verification of embedding distortion of blind data hiding: MSE of the modified graph vs. gradient difference, for individual graphs with number of nodes $N = 5000$ and$N = 10000$, respectively.

(a) (b)

Figure 3.9: Sensor graph. (a) Original Sensor graph. (b) Modified Sensor graph.

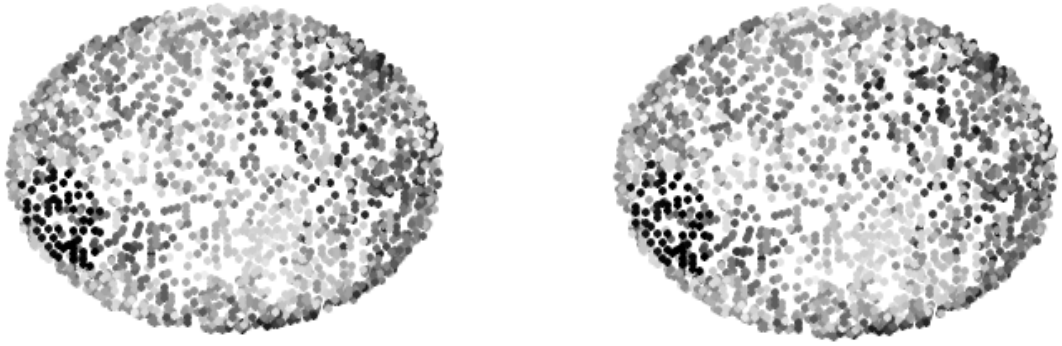


(a) $N = 5000$ (b) $N = 10000$

Figure 3.10: Embedding distortion performance of the non-blind algorithm using graphs with different numbers of nodes for various embedding capacities. (a) $N = 5000$. (b) $N = 10000$.

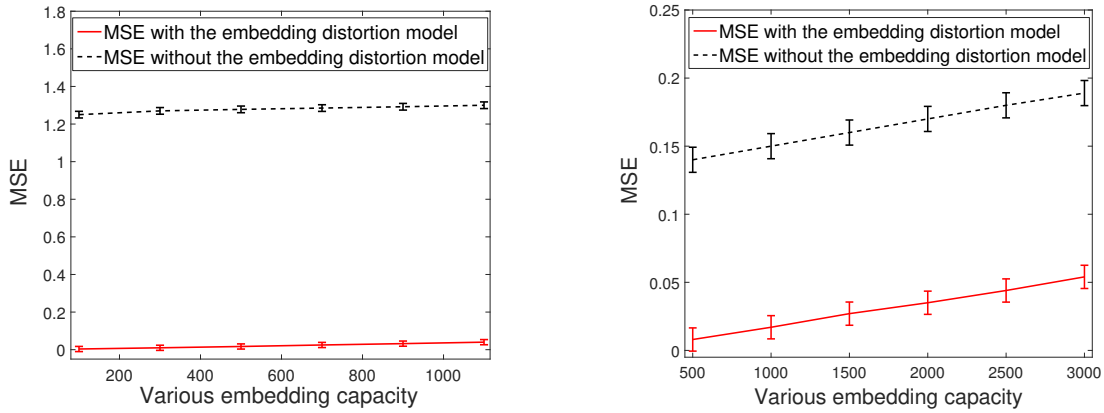
the secret bits in the GFT coefficients triple which have gradient difference close to 0 and MSE of the modified graphs are calculated by using the same blind algorithm without using the proposed model by embedding the same secret bits in the GFT coefficients triple which are selected randomly (without considering their gradient difference). Figure 3.11 shows the sphere graph with $N = 5000$ nodes before and after embedding the secret bits with length 500 bits. The empirical results show that the blind algorithm with the proposed model achieves lower distortion over the original algorithm without the model. As shown in Figure 3.12, the distortion is improved by an average of 80% and 99% for $N = 10000$ and $N = 5000$ nodes, respectively. Moreover, we can see that the embedding distortion is increased when the embedding capacity is increased. The blind data hiding has a less distortion compared to the non-blind data hiding

because the blind algorithm has a less embedding capacity according to the embedding model. In addition, decreasing the number of graph nodes N make a less differences between the GFT coefficients and this leads to reduce the differences between the MSE values.



(a) (b)

Figure 3.11: Sphere graph (a) Original Sphere graph. (b) Modified Sphere graph.



(a) $N = 5000$ (b) $N = 10000$

Figure 3.12: Embedding distortion performance of the blind algorithm using graphs with different numbers of nodes for various embedding capacities. (a) $N = 5000$. (b) $N = 10000$.

3.3.3 Evaluation of the performance of the robustness model

This section presents the performance evaluation results of the proposed robustness model for non-blind and blind data hiding. For performance metrics, we have selected the Hamming Distance (HD). Hamming Distance of the extracted secret data (often referred as Bit Error Rate (BER) in communication systems) were calculated for robustness evaluation.

3.3.3.1 Performance evaluation of the robustness model of the non-blind data hiding

The robustness model of non-blind data hiding is verified in the experimental simulations using graph dataset with $N = 2500$ graph nodes.. Two sets of the expriments are obtained for verifying the robustness model. In the expriments Set 1, the Hamming Distance (HD) of the extracted secret bits has calculated after the attack using the non-blind algorithm with the proposed model by selecting the graph Fourier coefficients that satisfy the specific conditions (in Eq. (3.21), Eq. (3.27), and Eq. (3.33)) to embed the secret bits and the Hamming Distance (HD) of the extracted secret bits has calculated after the attack by using the non-blind algorithm without using the proposed model (by embedding the same secret bits in the GFT coefficients randomly). We consider two attack types, namely, noise addition and deletion nodes data. For noise addition, we add the noise to all signal of the modified graph using various $\sigma^2 = \{0.01,\ 0.05,\ 0.1,\ 0.2,\ 0.3\}$ values the following equation:

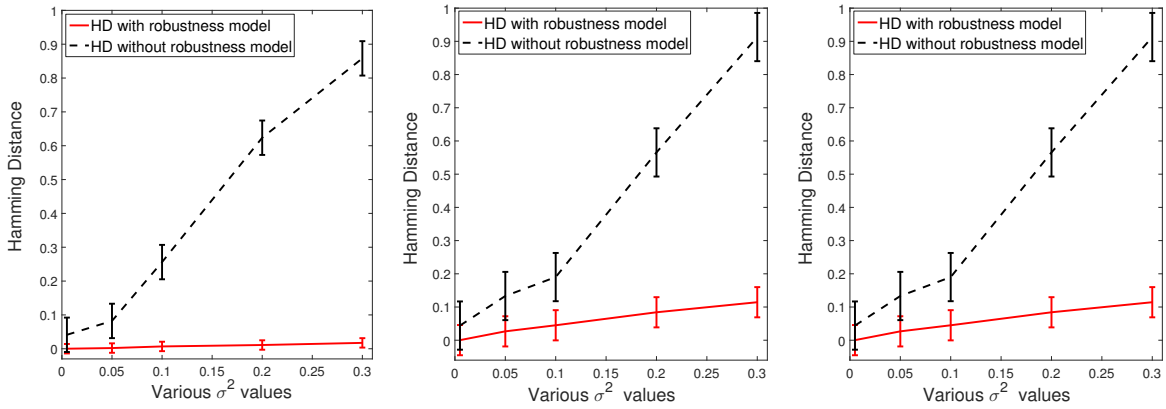
$$\mathbf{x}'_w = \mathbf{x}_w + \sigma^2 \times randn(N). \tag{3.55}$$

For deletion nodes data, we delete a different number of nodes data ($5,\ 10,\ 50,\ 100$ nodes) randomly by modifying the GFT coefficient values to zero. Pseudo-random binary sequences are considered as the secret bits, for three scenarios: $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0, 1\}$ to hide in the graph Fourier coefficients using $\alpha = 0.5$.

We can notice that the non-blind algorithm with the proposed model achieves higher robustness over the original algorithm without the model. As shown in Figure 3.13, the robustness against the additive noise is improved by an average of 99 %, 91% and 91 % for three embedding scenarios, $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$, respectively.

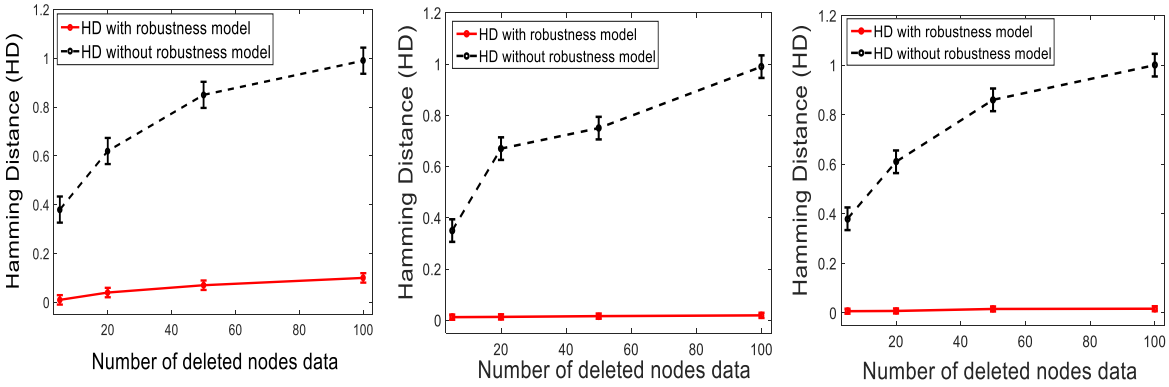
In addition, the robustness of the proposed method has been evaluated against deleting various numbers of nodes data randomly. Figure 3.14 illustrates improving the robustness using the proposed model after deletion nodes data by an average of 99.8 %, for three embedding scenarios, $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$, respectively.

In the experiment Set 2, we calculate the Hamming Distance (HD) of the extracted secret bits after the attacks. We have selected 5 binary logos, ieee, arrow, logo-inverse, medicine and logo-university to embed in the GFT coefficients. The binary logos with (30×30) bits$\mathbf{w} = \{0, 1\}$



(a) Hiding $b = 1$ (b) Hiding $b = 0$ (c) Hiding $b = 0,\ 1$

Figure 3.13: Hamming distance (HD) of the extracted secret bits using non-blind algorithm after noise addition for different values of σ^2 using $\alpha = 0.5$. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.



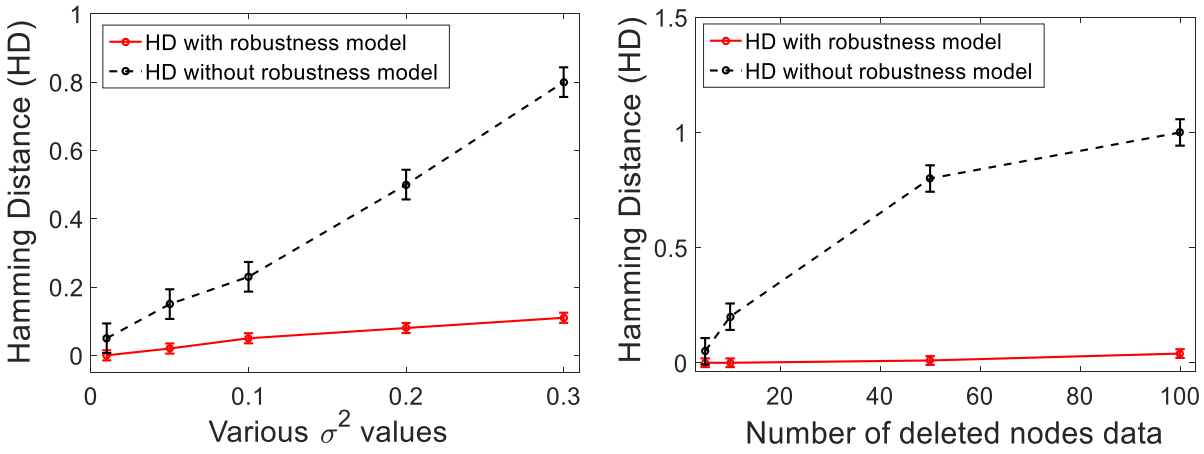
(a) Hiding $\mathbf{w} = \{1\}$ (b) Hiding $\mathbf{w} = \{0\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 3.14: Hamming distance (HD) of the extracted secret bits using the non-blind algorithm after deleting various number of nodes data using $\alpha = 0.5$. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.

are embedded in the GFT coefficients using graph dataset with $N = 2500$ graph nodes. The HD is calculated using the non-blind algorithm with using the proposed robustness model (by embedding the secret bits in the GFT coefficients that satisfy the specific condition in Eq. (3.33)). Also, we calculate the HD using the non-blind algorithm without using the proposed model (by embedding the secret bits in the GFT coefficients which are selected randomly).

We can notice that the non-blind algorithm with the proposed model achieves higher robustness over the original algorithm without the model. As shown in Figure 3.15, the robustness against the additive noise is improved by an average of 90% and 99.6% after deletion nodes

data.



(a) (b)

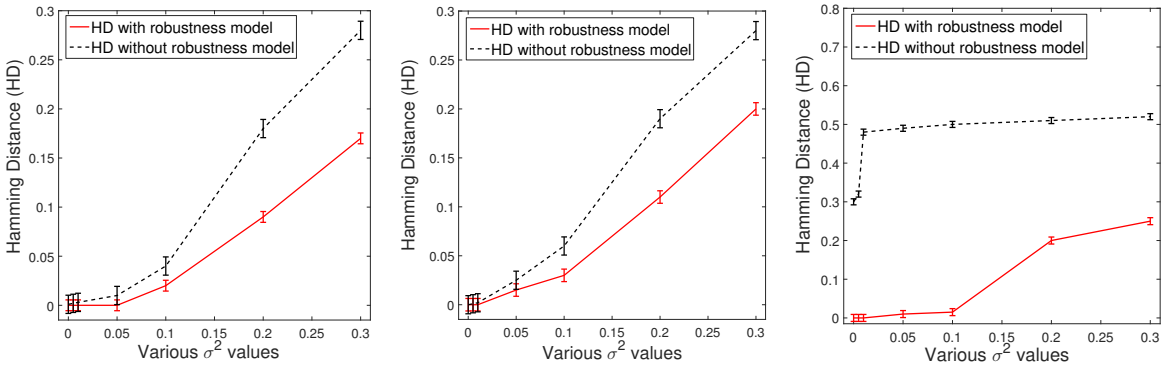
Figure 3.15: Robustness performance of non-blind data hiding after the attacks. (a) Additive noise. (b) Deleting nodes data.

3.3.3.2 Performance evaluation of robustness model of the blind data hiding

We verify the robustness model of the blind algorithm by the experimental results using a graph dataset with $N = 5000$ nodes. We calculate the Hamming distance (HD) of the extracted secret data after the attacks using the original algorithm with using the robustness model based on selecting the graph Fourier coefficients that satisfy the specific conditions (in Eq. (3.38), Eq. (3.46), and Eq. (3.54)) to embed the secret data and we calculate the Hamming distance (HD) of the extracted secret data after the attacks using the original algorithm without using the robustness model by embedding the secret bits in the GFT coefficients randomly. We consider pseudo-random number sequences as the secret data for representing three scenarios, $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0, 1\}$ to embed in the graph Fourier coefficients. Two types of attacks are considered, additive noise and deleting nodes data (as illustrated in the previous subsection).

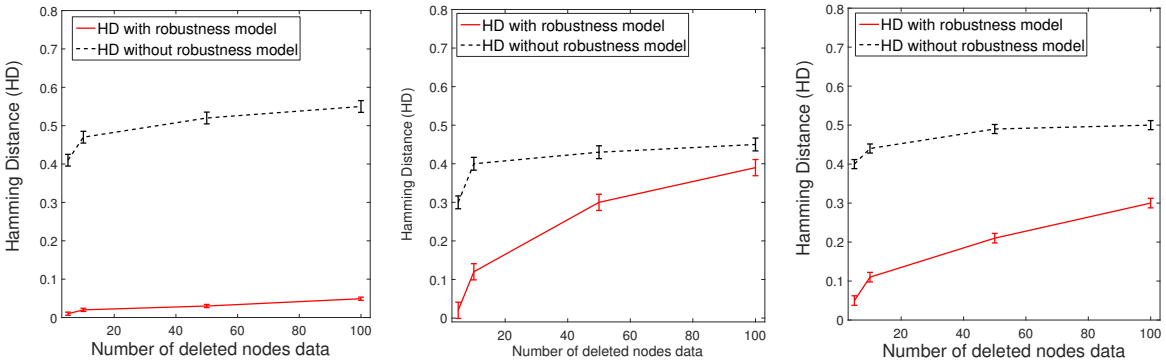
The results show improving the robustness of the proposed method after the attacks by using the robustness model. Figure 3.16 illustrates that the robustness against the additive noise is enhanced by an average of 62%, 30% and 87% for three embedding scenarios, $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$, respectively. In addition, we have evaluated the robustness of the proposed method against deleting various numbers of nodes data randomly. Figure 3.17

demonstrates improving the robustness using the proposed model after deletion nodes data by an average of 95%, 54% and 64% for three embedding scenarios, $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$, respectively.



(a) Hiding $\mathbf{w} = \{1\}$ (b) Hiding $\mathbf{w} = \{0\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 3.16: Hamming distance (HD) of the extracted the secret bits using the blind algorithm after noise addition for different values of σ^2. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.



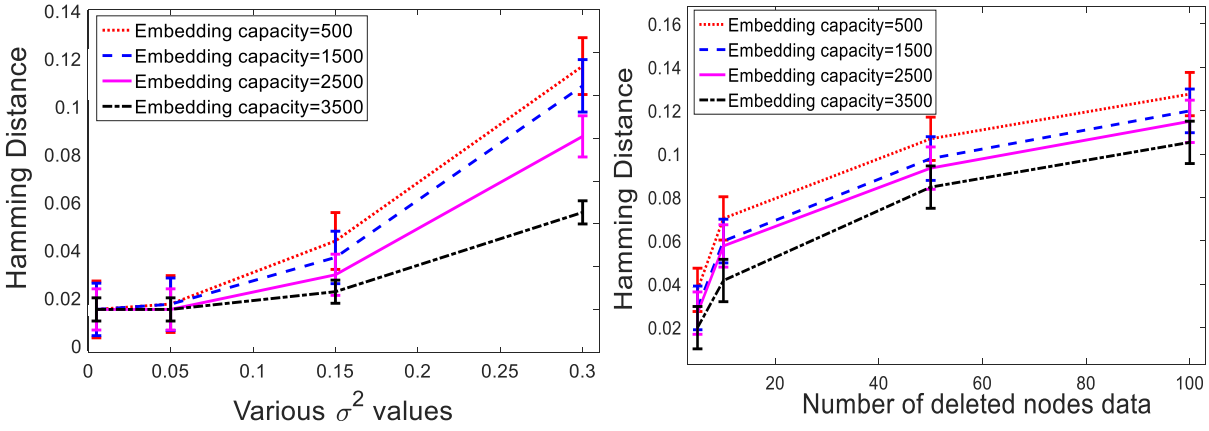
(a) Hiding $\mathbf{w} = \{1\}$ (b) Hiding $\mathbf{w} = \{0\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 3.17: Hamming distance (HD) of the secret bits using the blind algorithm after deleting various number of nodes data randomly. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.

3.3.3.3 Robustness performance of the non-blind data hiding

The robustness performance of the non-blind algorithm against the attack is evaluated at various embedding capacities using a graph dataset with $N = 5000$. Two types of attacks are considered, namely, additive noise and deletion nodes data.The Hamming Distance (HD) of the extracted bits have been calculated after the attacks using the non-blind algorithm with the

proposed robustness model (by embedding the secret bits in the GFT coefficients which satisfy the robustness conditions) and using the $\alpha = 0.5$. Pseudo-random binary sequences are considered as the secret bits, $\mathbf{w} = \{0, 1\}$ to embed in the GFT coefficients. The experimental results show that the robustness of the non-blind algorithm is increased when the embedding capacity is increased. Figure 3.18 shows enhancing the robustness of the proposed method after additive noise for various values of σ^2 and deleting different number of nodes data randomly.



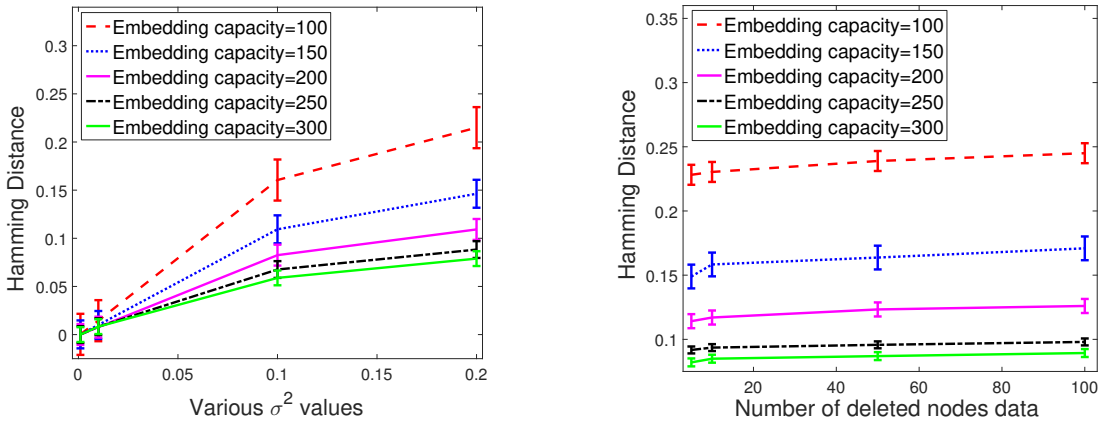
(a) After additive noise

(b) After deletion nodes data

Figure 3.18: Robustness performance of the non-blind algorithm using the robustness model against attacks for various embedding capacities. (a) Additive noise. (b) Deletion nodes data.

3.3.3.4 Robustness performance of the blind data hiding

The robustness performance of the blind algorithm has evaluated against the attacks at various embedding capacities using graph dataset with $N = 2500$ nodes. We calculate the Hamming Distance (HD) of the extracted bits using the proposed method (by embedding the secret bits in the GFT coefficients which satisfy the robustness conditions) after the additive noise for various values of σ^2 and after deletion of a various number of nodes data randomly. We consider pseudo-random number sequences as the secret bits, to represent $\mathbf{w} = \{0, 1\}$. The empirical results demonstrate that the robustness is increased when increasing the embedding capacity as shown in Figure 3.19.



(a) After additive noise

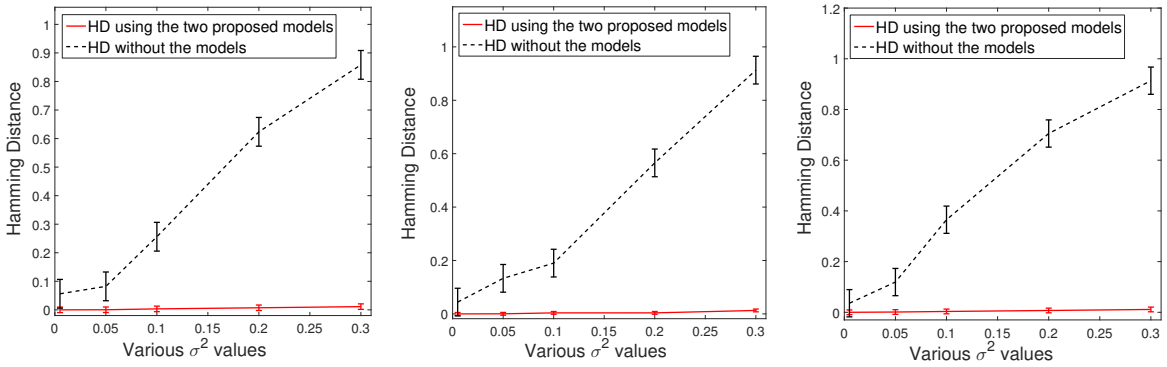
(b) After deletion nodes data

Figure 3.19: Robustness performance of the blind algorithm using the proposed model against attacks using various embedding capacities. (a) Additive noise. (b) Deletion nodes data.

3.3.4 Joint robust-low distortion data hiding

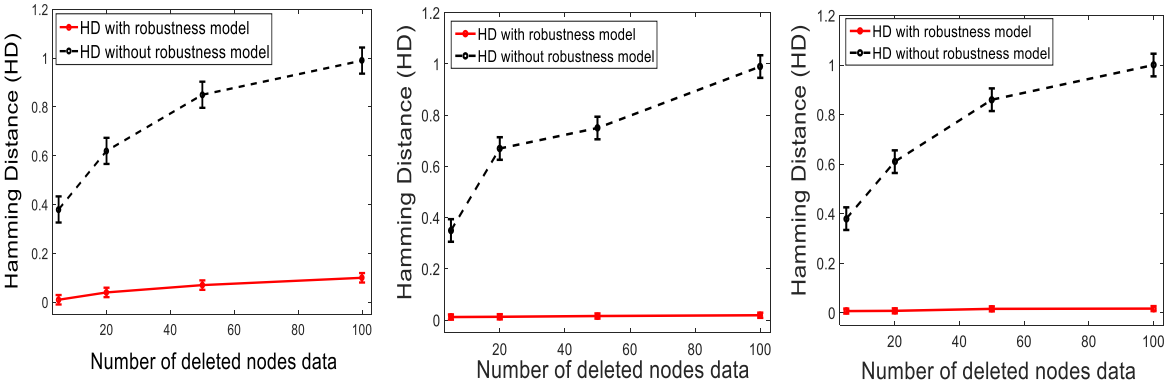
For obtaining a data hiding approach with low distortion and high robustness to attacks, we combine the embedding distortion minimisation model with the robustness model for blind and non-blind approaches. We calculated the Hamming Distance (HD) of the extracted secret bits after the additive noise and deleting nodes data by using the data hiding methods with using the two proposed models (based on selecting the GFT coefficients that satisfy the robustness conditions, followed by selecting GFT coefficients with low values for non-blind algorithm and the GFT coefficients triple which have gradient differences close to 0 for blind algorithm for minimising the embedding distortion from the above GFT coefficients to embed the secret bits) and we calculated the Hamming Distance (HD) of the extracted secret bits after the additive noise and deleting nodes data by using the same data hiding methods without using the proposed models (by selected the GFT coefficients randomly to embed the same secret bits). We observed that the Hamming Distance (HD) of the extracted secret bits was decreased by using the proposed models which means the robustness of the data hiding methods is improved by using the models. The experimental results demonstrate that the robustness of the proposed methods are improved by an average of 93% and 99.8% for non-blind and by an average of 60% and 71% for blind after the additive noise and deletion nodes data. Figure 3.20, Figure 3.21, Figure 3.22 and Figure 3.23 illustrate the robustness performance of the proposed methods after the noise addition for various values of σ^2 and deleting different number of nodes data

for non-blind and blind data hiding, respectively.



(a) Hiding $\mathbf{w} = \{1\}$ (b) Hiding $\mathbf{w} = \{0\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 3.20: Hamming distance (HD) of the extracted secret bits after noise addition for different σ^2 values using the non-blind algorithm with the two models. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.

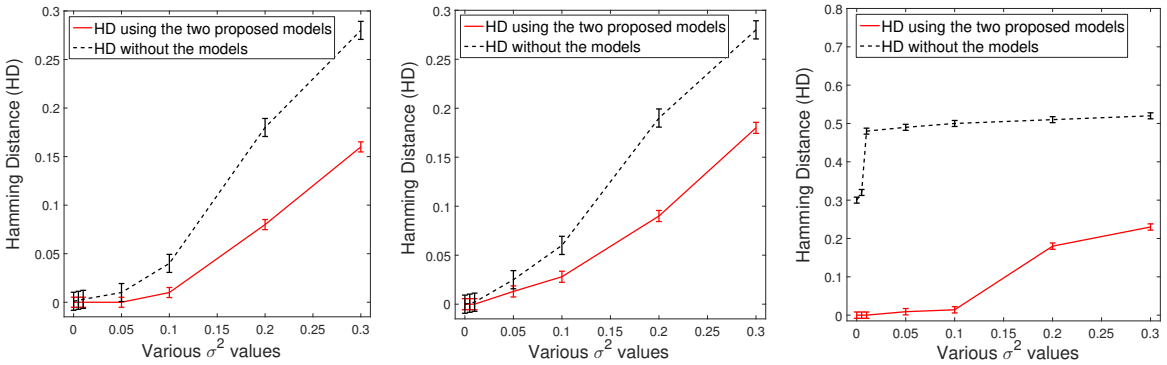


(a) Hiding $\mathbf{w} = \{1\}$ (b) Hiding $\mathbf{w} = \{0\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 3.21: Hamming distance (HD) of the secret bits after deletion various number of random nodes data using the non-blind algorithm with the two models. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.

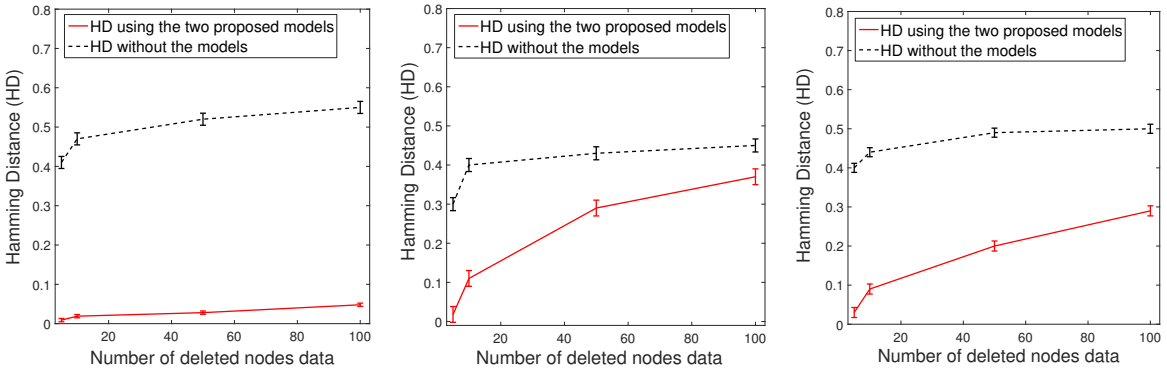
3.4 Concluding remarks

This chapter proposes a novel graph Fourier domain data hiding by considering two scenarios of data hiding: non-blind and blind. Two new models have been proposed to minimise the embedding distortion on the modified graph and to make the secret bits robust for two kinds of attacks, namely, noise addition and nodes data deletion. The embedding distortion minimisation model



(a) Hiding $\mathbf{w} = \{1\}$ (b) Hiding $\mathbf{w} = \{0\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 3.22: Hamming distance (HD) of the secret bits after noise addition for different values of σ^2 using blind algorithm with the two models. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.



(a) Hiding $\mathbf{w} = \{1\}$ (b) Hiding $\mathbf{w} = \{0\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 3.23: Hamming distance (HD) of the secret bits after deletion various of random nodes data using the blind algorithm with the two models. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.

of non-blind algorithm requires to choose the coefficients with low values, while the distortion minimisation model of blind algorithm requires to select the coefficients triple with the gradient difference close to 0 for minimising the distortion. We propose the robustness model to enhance the robustness against the attacks by choosing the graph Fourier coefficients that satisfy the certain conditions for embedding the secret data. The experimental results demonstrate that the proposed methods using the embedding distortion minimisation model have achieved lower distortion over the original methods by more than 94% and 80% for non-blind and blind algorithms, respectively. The proposed methods have compared in terms of the robustness against the attacks. We can see that the robustness of the proposed methods are improved by an average

of 93% and 99.8% for non-blind and by an average of 60% and 71% for blind after the additive noise and deletion nodes data. In the next chapter, a reversible data hiding approach for graph data is proposed using graph Fourier domain.

Chapter 4

Graph Fourier domain reversible data hiding for graph data

4.1 Introduction

The previous chapter proposed an irreversible data hiding algorithm in the graph Fourier domain. This chapter proposes a reversible data hiding algorithm in the graph Fourier domain. Due to the increasing volumes of data recorded on non-Cartesian grids in applications such as sensor networks, IoT applications, weather data, and medical data, the protection of these data has become a paramount interest. The traditional data hiding methods are not acceptable in several applications, such as military electronic data, remote sensing data, and medical images, because of the distortion in the host media due to embedding the secret data. Alternatively, reversible data hiding methods are used to recover the secret data and the original host data with error-free after extracting the secret content [183–189].

Most of the reversible data hiding algorithms on a graph rely primarily on the mesh. The reversible data hiding algorithms are classified into four categories depending on the embedding domain. These categories are the vertex domain, compressed domain, transform domain, and encrypted domain. In the vertex domain, the secret data are hidden by modifying the vertex coordinates [173, 190] or based on the distance between the faces and the centroid [191] with low computational complexity. Jhou et al. [192] have proposed hiding the secret data by classifying the selected few digits of nodes according to proper conditions. In the compressed domain, the

secret bits are hidden using predictive vector quantization [166, 193]. In the transform domain, the transform coefficients are used to hide the secret bits [172, 194]. The encrypted domain hides the secret bits in the encrypted mesh coordinates using an encryption key to meet the requirements of preserving privacy [171, 195].

The graph Fourier domain data hiding technique has proven to be a very effective approach for protecting graph data, due to advances in signal transforms, as shown in chapter 3. In this chapter, we propose a new reversible data hiding algorithm in the graph Fourier domain using histogram shifting for unstructured data, which are represented as a weighted graph. The proposed approach adopts the histogram-shifting process to provide a perfect reconstruct for both the embedded data and the original graph signal that is distortion-free, especially for the non-integer and negative coefficients. Histogram shifting is a successful algorithm to be used for restoring the integer and non-integer data, compared to image-based RDH algorithms that are difficult to apply to non-integer data, such as graph data.

This chapter proposes a reversible data hiding approach in the graph Fourier domain. We propose an embedding distortion-minimisation model to reduce embedding distortion and a robustness model to select the embedding coefficients, which are resistance for attacks. We identify the relationship between the error distortion metric (using MSE) and the value of the embedding data to minimise the embedding distortion. In addition, the relationship between the extraction process and the effect of the attack namely, noise addition is also identified to enhance the robustness of the proposed method. These two models are combined to satisfy the requirements of data hiding. The proposed approach does not require the storage of additional data to restore the original signal distortion-free; nor does it utilise compression to minimise the embedding distortion. The primary contributions of this chapter are:

1. Proposing a new graph Fourier domain histogram-shifting algorithm for reversible data hiding on non-integer data.

2. Proposing new models to minimise embedding distortion in the host graph signal after embedding and to make the embedded data robust to additive noise.

The rest of the chapter is organised as the following: Section 4.2 describes the proposed methodology including the proposed graph Fourier domain reversible data hiding algorithm,

followed by the embedding distortion minimisation and the robustness enhancing models. The performance evaluation is discussed in Section 4.3. The concluding remarks is presented in Section 4.4.

4.2 Proposed Methodology

4.2.1 Graph Fourier Transform (GFT)

We consider that $\mathcal{G}$ is an undirected graph without self-loops and multiple edges between nodes. We define the adjacency matrix with edge weights, $\mathbf{A}$, and the combinatorial graph Laplacian matrix, $\mathbf{L}$, as in Eq. (2.9) and Eq. (2.12). The Graph Fourier Transform (GFT) and its inverse are defined as in Eq. (2.17) and Eq. (2.18).

4.2.2 Reversible data hiding algorithm

This section presents the proposed RDH method based on histogram shifting in the graph Fourier domain (Figure 4.1). The proposed method aims to obtain a sharpness histogram of the coefficients by leveraging the use of the graph Fourier transform to decrease embedding distortion and increase embedding capacity. In addition, this method can increase robustness due to advances in signal transforms. By using the graph Fourier transform, we obtain a GFT histogram that has several peak points with high values and many zero points, thus increasing the embedding rates. The proposed method involves two procedures: embedding and extraction. The primary steps of the embedding procedure are as follows:

1. The GFT coefficients are calculated using Eq. (2.17).

2. The histogram of the magnitudes of the GFT coefficients is generated.

3. All magnitudes of the GFT coefficients are scanned to determine the peak point, $h(X_{Max})$, and zero point, $h(X_{Min})$, where X_{Max} and X_{Min} are the magnitudes of the GFT coefficients that have the largest and lowest (or zero in most cases) repetition, respectively.

4. Next, all of the coefficients' magnitudes between $X_{Max} + q$ and $X_{Min} - q$ are shifted toward the direction of X_{Min} based on the value of shifting bin q, which depends on the

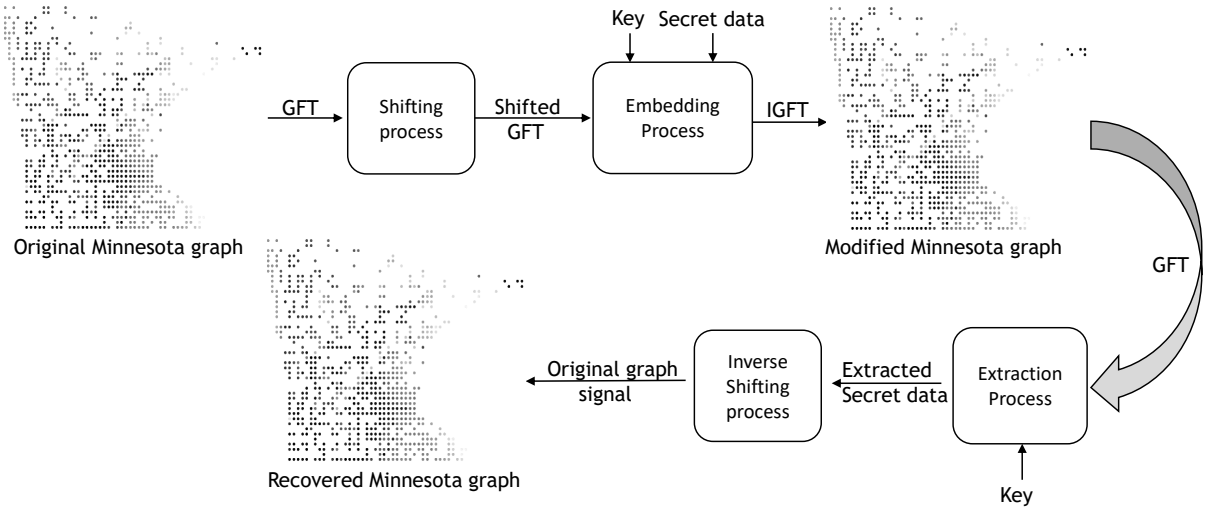


Figure 4.1: The block diagram of the proposed reversible data hiding approach.

value of the embedding bit $w_b(b = \{0, q\})$. We consider the shifting bin $q = 1$ based on the embedding bit value of $w_b(b = \{0, 1\})$.

5. The magnitudes of the GFT coefficients that are greater than X_{Min} and less than X_{Max} remain unchanged. We discussed the case when $X_{Max} < X_{Min}$ because this is the most common case according to our GFT coefficients. The second case when $X_{Max} > X_{Min}$, in this case the same steps are repeated. The only difference is to shift the magnitudes of the GFT coefficients toward the direction of X_{Min} based on the value of shifting bin q (based on reducing the magnitudes of the GFT coefficients by q).

6. The secret bits are embedded into the GFT coefficients' magnitudes X_{Max}. If the bit is 0, the GFT coefficient's magnitude remains without change in X_{Max}. Otherwise, the GFT coefficient's magnitude will be in $X_{Max} + q$ by adding $q = 1$ when the embedded bit is 1. Let $\mathbf{X}$ and $\mathbf{X_w}$ represent the original and the modified GFT coefficient's magnitudes, respectively. For a single embedding, we obtain:

$$\mathbf{X}_w(\lambda) = \begin{cases} \mathbf{X}(\lambda) + q, \ \mathbf{X}(\lambda) \in [X_{Max} + q, X_{Min} - q], \\ \mathbf{X}(\lambda) + w_b, \ \mathbf{X}(\lambda) = X_{Max}, \\ \mathbf{X}(\lambda), \text{otherwise} \ . \end{cases} \tag{4.1}$$

where w_b is the secret data to be hidden.

7. The IGFT is performed on the modified GFT coefficients to obtain the modified graph signal.

The extraction process is the reverse process of data embedding. The embedded bits are extracted based on the embedding key which is sent to the blind extractor in a separate file. The embedding key includes a number of shifting times, peak points, X_{Max}, zero points, X_{Min}, the shifting bin, q, N, w_0, w_1 and length of the secret bits. The extraction process includes the following steps:

1. The modified GFT coefficients are calculated using Eq. (2.17).

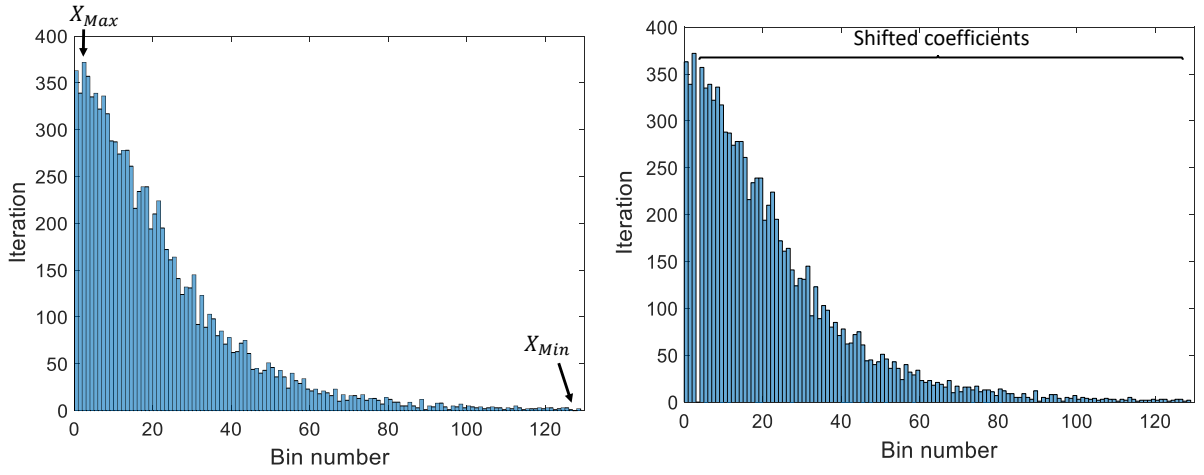
2. Scan the GFT coefficients to find the X_{Max} and $X_{Max} + q$.

3. The extracted bit is 0 when the GFT coefficient's magnitude is X_{Max}; the extracted bit is 1 when the GFT coefficient's magnitude is $X_{Max} + q$.

4. To recover the original coefficient's magnitudes perfectly, all embedded bits are subtracted from the modified coefficient's magnitudes in the range $[X_{Max}, X_{Max} + q]$.

5. All of the magnitudes of the coefficients in the range $[X_{Max} + 2q, X_{Min}]$ are shifted back by subtracting one unit ($q = 1$) for example.

For instance, assume that $w_b(b = \{0, 1\})$ for the embedding bit and the shifting bin is $q = 1$. We consider p and r to be the coefficient magnitudes. $h(X_{max})$ is $h(p)$, where$h(p)$ represents the frequency of occurrence for magnitude p in the histogram of the GFT coefficients; $h(X_{min})$ is $h(r)$, which refers to the frequency of occurrence for magnitude r of the GFT coefficients, with zero frequency. At this point, all of the GFT coefficient magnitudes in the range $[p+1, r-1]$ are shifted (right) toward point r by one unit, while the remaining GFT coefficient magnitudes, which are greater than r and less than p, are left without modification, as illustrated in Figure 4.2. The secret bits are hidden in the GFT coefficients, which have a magnitude of p. If the embedded bit is 0, the GFT coefficient magnitude p remains unchanged. Otherwise, the GFT coefficient magnitude moves to $p + 1$. In the extraction procedure, the embedded data are



(a) (b)

Figure 4.2: Histogram of the GFT coefficients of Sensor graph with 10000 nodes and shifting bin $q = 1$. (a) Before the shifting process. (b) After the shifting process

extracted from the modified GFT coefficient magnitudes in the range $[p, p+1]$. The secret bit is 0 when the GFT magnitude is p; otherwise, the embedded bit is 1. To restore the original GFT coefficient magnitudes, the embedded bits are first subtracted from the modified GFT coefficient magnitudes in the range $[p, p+1]$. Next, all GFT coefficient magnitudes in the range $[p+2, r]$ are shifted to the left (back) by one unit ($q = 1$).

Based on our experiments using our graph data set (as illustrated in Section 4.3) for various graph types with different graph signals, we have determined that the peak points in the histogram of the GFT coefficients are concentrated in the range $[0, 20]$. Therefore, these ranges are selected to be X_{Max} for embedding the secret data.

The following algorithms, i.e. algorithm 1 and algorithm 2, describe the embedding and the extraction processes, respectively, where $w = \{0, q\}$ are the embedded bits sequence.

The embedding capacity depends on the value of the shifting bin q and the number of peak points that are used in the embedding process. For example, when the shifting bin value is large, it causes the peak point to be high. In other words, it increases the number of the GFT coefficient X_{Max}. In addition, the embedding capacity can be increased by using several peak points for embedding the data. For example, if the peak value $h(X_{Max})$ is equal to the number of bits to be hidden in this case, only one peak point is utilised in the embedding process and the coefficients are shifted only one time. This case is referred to as a single embedding. However,

Algorithm 1: **Embedding procedure**
Input = GFT coefficients magnitudes (X), secret bits(w), X_{Max} and X_{Min}.
Output= Modified GFT coefficients magnitudes (X_w).

$N \leftarrow$ Number of GFT coefficients magnitudes X.
for ℓ=1 to N **do**
    if $(X(\ell) \geq X_{Max} + q$ AND $X(\ell) < X_{Min})$ **then**
        $X_w(\ell) \leftarrow \mathrm{X}(\ell) + \mathrm{q}$.
    End if
End for
$\ell \leftarrow 1$.
$i_1 \leftarrow 1$.
While $(\ell \leq N$ AND $i_1 \leq Length(w))$
if $(X(\ell) \geq X_{Max}$ AND $X(\ell) < X_{Max} + q)$ **then**
    $X_w(\ell) \leftarrow \mathrm{X}(\ell) + \mathrm{w}(\mathrm{i}_1)$.
    $i_1 \leftarrow i_1 + 1$.
End if
$\ell \leftarrow \ell + 1$.
End While

when the embedding capacity is greater than the first peak point value, more than one peak point value is used (for example, the second peak point value and so on) and the coefficients are shifted many times; this case is referred to as a multiple embedding.

The modification value Δ_s , describing the shifting process for a single embedding, is determined based on the shifting bin value q. It is defined in as:

$$\Delta_s(\mathbf{X}) = \begin{cases} q, & \mathbf{X}(\lambda) \in [X_{Max} + q, X_{Min} - q], \\ 0, & \text{otherwise .} \end{cases} \tag{4.2}$$

To increase the embedding capacity, multiple embeddings are utilised. In this case, we determine several peak and zero points. The number of peak points and zero points that are utilised depends on the embedding capacity C. For a single embedding, only one peak point and zero point are utilised to hide the secret data when the length of the secret data is equal to

Algorithm 2: **Extraction procedure**
Input = Modified GFT coefficients magnitudes, $\mathbf{X_w}$ $\mathbf{X_{Max}}$ and $\mathbf{X_{Min}}$.
Output= Recovered GFT coefficients magnitudes (X_e) and secret bits(w').

$N \leftarrow$ Number of GFT coefficients magnitudes $\mathrm{X_w}$.
$\ell \leftarrow 1$.
$i_1 \leftarrow 1$.
While ($\ell \leq N$ AND $i_1 \leq Length(w)$)
if ($X_w(\ell) \geq X_{Max}$ AND $X_w(\ell) < X_{Max} + q$) **then**
    $w'(i_1) \leftarrow 0$.
elseif ($X_w(\ell) \geq X_{Max} + q$ AND $X_w(\ell) < X_{Max} + 2q$)
    $w'(i_1) \leftarrow$ q.
    $i_1 \leftarrow i_1 + 1$.
End if
$\ell \leftarrow \ell + 1$.
End While
$\ell \leftarrow 1$.
$i_1 \leftarrow 1$.
While ($\ell \leq N$ AND $i_1 \leq Length(w)$)
if $X_w(\ell) \geq X_{Max}$ AND $X_w(\ell) < X_{Max} + 2q$ **then**
    $X_e(\ell) \leftarrow \mathrm{X_w}(\ell) - \mathrm{w'(i_1)}$.
    $i_1 \leftarrow i_1 + 1$.
End if
$\ell \leftarrow \ell + 1$.
End While
for $\ell = 1$ to N **do**
    if $X_w(\ell) \geq X_{Max} + 2q$ AND $X_w(\ell) < X_{Min} + q$ **then**
        $X_e(\ell) \leftarrow X_w(\ell) - q$.
    End if
End for
for $\ell = 1$ to N **do**
    if $X_w(\ell) < X_{Max}$ OR $X_w(\ell) > X_{Min}$ **then**
        $X_e(\ell) \leftarrow X_w(\ell)$.
    End if
End for

the number of the GFT coefficient magnitudes', X_{Max}, as follows:

$$C = h(X_{Max}), \tag{4.3}$$

where $h(X_{Max})$ refers to the frequency of occurrence for magnitudes of the GFT coefficient X_{Max} in the histogram. When the secret bits', $w_b(b = \{0, 1\})$, and the number of the 0 and 1

are distributed equally, only half of the GFT coefficient magnitudes' X_{Max} are shifted by $q = 1$. The average distortion due to embedding 1 is defined as follows:

$$\delta(X_{Max}) = \frac{1}{2} \times C \tag{4.4}$$

The total modification value Δ_T in the GFT coefficient magnitudes using a histogram shifting algorithm is defined as given:

$$\Delta_T(\mathbf{X}) = \begin{cases} \delta(X_{Max}), & \mathbf{X}(\lambda) = X_{Max}, \\ q, & \mathbf{X}(\lambda) \in [X_{Max} + q, X_{Min} - q], \\ 0, & \text{otherwise .} \end{cases} \tag{4.5}$$

At this point, we must consider embedding distortion in the methods of reversible data hiding. Generally, in the traditional data hiding algorithms, the embedding distortion comes only from embedding the secret bits in the host media. In the reversible data hiding algorithms, there are two types of distortion: embedding distortion and reversibility distortion. The first type results from hiding the secret bits within the host media, whereas the second type results from the reversibility process to restore the original host data as follows:

$$D_T = D_E + D_R, \tag{4.6}$$

$$D_T = \delta(X_{Max}) + \Delta_s(\mathbf{X}),$$

$$= \Delta_T(\mathbf{X}),$$

where D_T is the total distortion due to embedding the secret data (D_E) and the reversibility process (D_R). D_E is equal to the modification value in the GFT coefficients due to embedding the secret data ($\delta(X_{Max})$). D_R is equal to the modification value in the GFT coefficients due to the shifting process ($\Delta_s(\mathbf{X})$). The embedding distortion depends on the embedding capacity of the secret data, whereas the reversibility distortion depends on the method used to restore the original host data. For example, in some of the reversible data hiding algorithms, extra data are hidden in the host media to recover the original host data accurately without error.

Our proposed algorithm uses the shifting process to recover the original graph coefficient magnitudes. Thus, the distortion of reversibility comes from the shifting process. At this point, we can define the shifting distortion. The shifting distortion depends on many parameters. One parameter is the shifting bin q, which depends on the values of the embedding bits $w_b(b = \{0, q\})$; when q is a small value, the shifting distortion will be low. Another parameter is the number of GFT coefficients to be shifted, which depends on the position of the X_{Min}. For instance, if the X_{Min} is distant from the X_{Max}, the distortion will be bigger compared to when the X_{Min} is near the X_{Max}. The third parameter is the number of shifts (single shifting or multiple shifting); this depends primarily on the embedding capacity, the original graph signal, and the graph connectivity. For example, if the embedding capacity is not large, the GFT coefficients are shifted only one time. In addition, if the correlation between the original graph signal is high, this reduces the number of shifts. Finally, the graph connectivity has a large effect on the embedding distortion. For instance, if the histogram of GFT coefficients for a Torus graph which has a highest peak point, this increases the value of the GFT coefficients' X_{Max}. In turn, this increases the embedding capacity and decreases the number of shifts, reducing the embedding distortion. In general, the distortion produced by using multiple embedding is higher than the distortion resulting from a single embedding. This is primarily because multiple embedding embeds more data and shifts the coefficients several times. Distortion is minimised when the number of shifts is decreased.

4.2.2.1 Authentication Process

We perform the authentication based on comparing the extracted secret bits with the original secret bits using the Hamming Distance (HD) as defined as in Eq. (2.6).

4.2.3 Embedding distortion minimisation

For establishing the relationship between the error distortion using mean square error (μ) and the value of the embedded bit, we define MSE (μ) in vertex domain between the original graph

signal $\mathbf{x}$ and modified graph signal $\mathbf{x}_w$ as follows:

$$\mu = \frac{1}{N}\sum_{i=0}^{N-1}(\mathbf{x}(i) - \mathbf{x}_w(i))^2. \tag{4.7}$$

Since the GFT forms an orthogonal set of eigenvectors, according to the Parseval's Theorem, $\|\mathbf{x}\|^2 = \|\mathbf{X}\|^2$, where $\mathbf{x}$ is the graph signal in vertex domain and $\mathbf{X}$ is the GFT coefficient [25]. Since the GFT is orthonormal, we can extend this to the sum of the error power in the input graph signal, $\Delta\mathbf{x}$, and to the sum of the error power in the graph Fourier domain $\Delta\mathbf{X}$ as follows:

$$\sum_i |\Delta\mathbf{x}(i)|^2 = \sum_\ell |\Delta\mathbf{X}(\ell)|^2. \tag{4.8}$$

From Eq. (4.7) and Eq. (4.8), we get

$$\mu = \frac{1}{N}\sum_\ell |\Delta\mathbf{X}(\ell)|^2. \tag{4.9}$$

From Eq. (4.6), we can notice that the embedding distortion comes from shifting the GFT coefficients by q and from embedding $w_b(b = \{0, q\})$. At this point, we have two cases:
When $w_b = 0$, in this case there is no embedding distortion.
The second case when $w_b = q$, the embedding distortion depends on the value of w_b.
by considering $q = w_b$, we can estimate each $\Delta\mathbf{X}(\ell)$ as:

$$\Delta\mathbf{X}(\ell) = \mathbf{X}(\ell)_w - \mathbf{X}(\ell), \tag{4.10}$$

$$= \mathbf{X} + w_b - \mathbf{X},$$

$$= w_b.$$

From Eq. (4.9) and we can expand Eq. (4.10), we obtain:

$$\mu \propto \sum_\ell |\Delta\mathbf{X}(\ell)|^2, \tag{4.11}$$

$$\mu \propto \sum \mathbf{w}^2$$

$$\mu = \frac{1}{N}\sum \mathbf{w}^2,$$

$$\mu = \frac{M}{N}\mathbf{w}^2,$$

where M is the number of modified coefficients $M < N$. This leading to

$$\mu < \mathbf{w}^2. \tag{4.12}$$

The relationship between the MSE of the modified graph and the secret bit value of $\mathbf{w}$ is established. The MSE of the modified graph is less than the value of $\mathbf{w}^2$. Therefore, to minimise the MSE, the $\mathbf{w}$ value should be small. A balance must be achieved between the embedding distortion and the embedding capacity based on choosing a value for $\mathbf{w}$. A low embedding distortion and low embedding capacity are obtained when the $\mathbf{w}$ value is small, and vice versa.

4.2.4 On enhancing robustness

To improve the robustness of the proposed method, we propose a robustness model to identify the magnitudes of the GFT coefficients which are able to retain the secret data after the attack in the graph Fourier domain. We have considered the additive noise on test graphs. The modified GFT coefficients magnitudes $\mathbf{X}_w(\ell)$ are changed based on the modification value due to attack Δ_a as given:

$$\mathbf{X}'_w(\ell) = \mathbf{X}_w(\ell) + \Delta_a, \tag{4.13}$$

where $\mathbf{X}'_w(\ell)$ are the modified graph Fourier coefficients after the attack. The value of modification Δ_a depends on the attack type. For instance, the value of modification due to adding noise depends on the noise variance value σ^2.

To extract the secret bit w'_b after the attack, we have new GFT coefficients magnitudes $\mathbf{X}'_w(\ell')$, $\mathbf{X}'_{Max}$ and $\mathbf{X}'_{Min}$:

$$w'_b = \begin{cases} 0, & \mathbf{X}'_w(\ell') = \mathbf{X}'_{Max}, \\ q, & \mathbf{X}'_w(\ell') = \mathbf{X}'_{Max} + q. \end{cases} \tag{4.14}$$

where $q = 1$.

Three cases of the secret bits are considered: hiding only $b = 0$ bits, Hiding only $b = 1$ bits and hiding $b = 0, 1$ bits, where $w_0 < w_1$, $w_0 = 0$, $w_1 = q$ and $q = 1$.

Proposition 4.1

To extract the correct secret bit after embedding $b = 1$ bits, the modified graph Fourier coefficients should be in the range:

$$\mathbf{X}'_{Max}(\ell') + q \leq \mathbf{X}'_{w}(\ell') < \mathbf{X}'_{Max}(\ell') + 2q. \tag{4.15}$$

where $w = q$ and $q = 1$.

Proof. To obtain the secret bit $b = 1$, we need to get $w'_b \geq q$,
From embedding step, we have:

$$\mathbf{X_w}(\ell) = \mathbf{X_{Max}}(\ell) + w_b. \tag{4.16}$$

This leads to:

$$w_b = \mathbf{X_w}(\ell) - \mathbf{X_{Max}}(\ell). \tag{4.17}$$

We need:

$$\mathbf{X_w}(\ell) - \mathbf{X_{Max}}(\ell) > q, \tag{4.18}$$

$$\mathbf{X_w}(\ell) > \mathbf{X_{Max}}(\ell) + q, \tag{4.19}$$

In the case of no attack, the modified coefficient $\mathbf{X}_w(\ell)$ after embedding the secret bit $b = 1$ will be in the range:

$$\mathbf{X}_{Max}(\ell) + q \leq \mathbf{X}_{w}(\ell) < \mathbf{X}_{Max}(\ell) + 2q.$$

After the attack, we have only the reconstructed coefficients, $\mathbf{X}'_w$, and $\mathbf{X}'_{Max}$. For correct extraction of the secret bit, we need:

$$\mathbf{X}'_{Max}(\ell') + q \leq \mathbf{X}'_{w}(\ell') < \mathbf{X}'_{Max}(\ell') + 2q.$$

The modified coefficients after the attack will be:

$$\mathbf{X}'_w(\ell) = \mathbf{X}_w(\ell) + \Delta_a. \tag{4.20}$$

And:

$$\mathbf{X}'_{Max}(\ell') = \mathbf{X}_{Max}(\ell) + \Delta_a, \tag{4.21}$$

where $\mathbf{X}'_w(\ell) + \Delta_a$ and $\mathbf{X}'_{Max}(\ell) + \Delta_a$ are the modified coefficients after the attack and the peak point after the attack respectively, by substituting them, we obtain:

$$\mathbf{X}_{Max}(\ell') + \Delta_a + q \leq \mathbf{X}'_w(\ell') < \mathbf{X}_{Max}(\ell') + \Delta_a + 2q.$$

Since $q < 2q$, then we got:

$$\mathbf{X}'_{Max}(\ell') + q \leq \mathbf{X}'_w(\ell') < \mathbf{X}'_{Max}(\ell') + 2q.$$

□

Proposition 4.2

For embedding $b = 0$ bits, we can extract the correct secret bits when the modified coefficients magnitudes are in the range:

$$\mathbf{X}'_{Max}(\ell') + w_0 \leq \mathbf{X}'_w(\ell') < \mathbf{X}'_{Max}(\ell') + q. \tag{4.22}$$

where $w_0 = 0$ and $q = 1$.

Proof. To obtain the secret bit $b = 0$, we need to get $w'_b < q$,
From embedding step, we have:

$$\mathbf{X}_w(\ell) = \mathbf{X}_{Max}(\ell) + w_b. \tag{4.23}$$

This leads to:

$$w_b = \mathbf{X}_w(\ell) - \mathbf{X}_{Max}(\ell). \tag{4.24}$$

We need:

$$\mathbf{X}_w(\ell) - \mathbf{X}_{Max}(\ell) < q, \tag{4.25}$$

$$\mathbf{X}_w(\ell) < \mathbf{X}_{Max}(\ell) + q. \tag{4.26}$$

In the case of no attack, the modified coefficient $\mathbf{X}_w(\ell)$ after embedding the secret bit $b = 0$ will be in the range:

$$\mathbf{X}_{Max}(\ell') + w_0 \leq \mathbf{X}'_w(\ell') < \mathbf{X}_{Max}(\ell') + q.$$

After the attack, we have only the reconstructed coefficients, $\mathbf{X}'_w$, and $\mathbf{X}'_{Max}$. For correct extraction of the secret bit, we need:

$$\mathbf{X}'_{Max}(\ell') + w_0 \leq \mathbf{X}'_w(\ell') < \mathbf{X}'_{Max}(\ell') + q.$$

The modified coefficients after the attack will be:

$$\mathbf{X}'_w(\ell) = \mathbf{X}_w(\ell) + \Delta_a. \tag{4.27}$$

And:

$$\mathbf{X}'_{Max}(\ell) = \mathbf{X}_{Max}(\ell) + \Delta_a, \tag{4.28}$$

where $\mathbf{X}_w(\ell) + \Delta_a$ and $\mathbf{X}_{Max}(\ell) + \Delta_a$ are the modified coefficients after the attack and the peak point after the attack respectively, by substituting them, we obtain:

$$\mathbf{X}_{Max}(\ell') + \Delta_a + w_0 \leq \mathbf{X}'_w(\ell') < \mathbf{X}_{Max}(\ell') + \Delta_a + q.$$

Since $w_0 < q$, then we get:

$$\mathbf{X}'_{Max}(\ell') + w_0 \leq \mathbf{X}'_w(\ell') < \mathbf{X}'_{Max}(\ell') + q.$$

□

Proposition 4.3

We combine the two previous propositions to find the condition of correct extraction of the secret bits when hiding $b = 0$ and $b = 1$. The range of the graph Fourier coefficients that able to retain the secret data bits correctly is:

$$\mathbf{X}'_{Max}(\ell') + w_0 \leq \mathbf{X}'_w(\ell') < \mathbf{X}'_{Max}(\ell') + 2q. \tag{4.29}$$

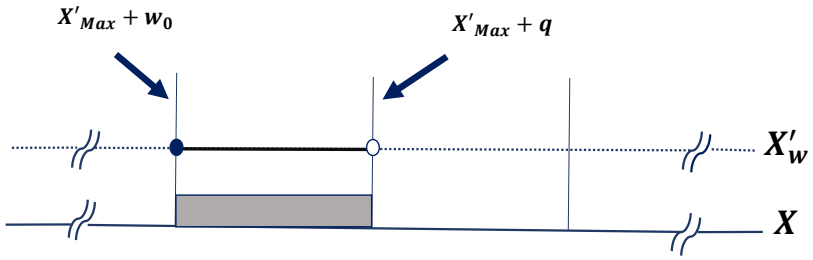
where $w_0 = 0$ and $q = 1$. Figure 4.3 displays the range of the graph Fourier coefficients which is able of retaining the embedded bits after the attacks.

4.2.5 Joint robust-low distortion reversible data hiding

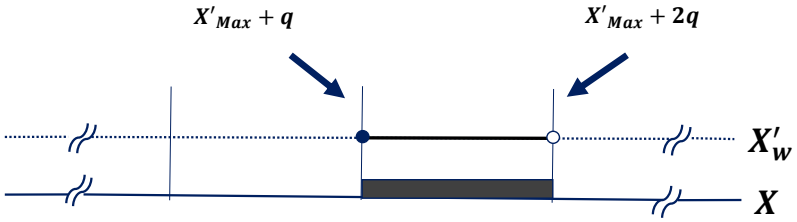
We combine the two proposed models, embedding distortion minimisation and robustness for satisfying the main requirements of the graph reversible data hiding. We select the GFT coefficients that satisfy the condition in Eq. (4.29) to satisfy the robustness condition. Then, we select a small value for $\mathbf{w}$ less than 1 (according to the Eq. (4.12) for minimising the embedding distortion) to embed in the selected GFT coefficients for minimising the embedding distortion and enhance the robustness.

4.3 Performance evaluation

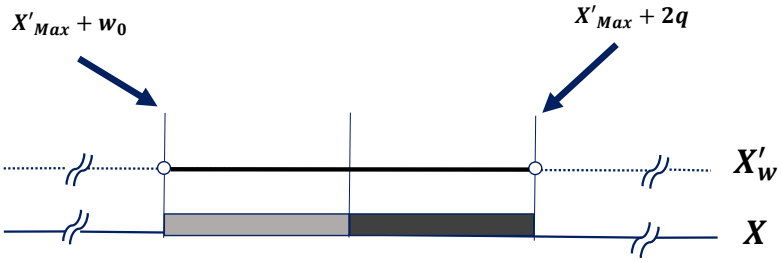
This section presents the evaluation of the proposed method performance. The experimental simulations include: verification of the embedding distortion model, evaluation the performance of the proposed algorithm in terms of embedding distortion, withstand attacks, as well as reversibility of the original graph signal. In addition, the proposed method is compared with two previous reversible data hiding methods: Ni et al. [3] using histogram shifting algorithm in pixel domain and Dragoi et al. [4]. Based on the best of our knowledge, the proposed reversible data hiding using graph spectral domain is unique, and due to lack of any other comparable work and for the same reasons which are mentioned in chapter 3, we selected the reversible data hiding



(a)



(b)



(c)

Figure 4.3: The range of graph Fourier coefficients that is able of extracting the embedded bits correctly. (a) Hiding only $b = 0$. (b) Hiding only $b = 1$. (c) Hiding $b = 0$ and $b = 1$.

on images because both the proposed methods and the images have signal(data). For making the comparison fairer, we used the same graph signals to apply in the proposed methods and the comparable methods in the comparison. We selected the methods in [3] and [4] for comparison to include two domains: pixel domain and encrypted domain. The following reasons are justifying the selection of the methods in [3] and [4]:

The reason for selecting the method in [3] is that the proposed reversible data hiding is based

on histogram shifting, therefore, for fair comparison, we have to compare the proposed method with a reversible data hiding using the histogram shifting. The reason for selecting the histogram shifting algorithm is because the spectral coefficients values are real values. Therefore, for recovering the original graph signal without any error, it is very difficult to apply the other methods such as the RDH using prediction error which is not error-free, due to truncation error and RDH using quantization, which is also not invertible due to the quantization error. Before selecting the histogram shifting algorithm, we first selected the RDH using prediction error to apply on the graph data but we recovered the graph signal with an error by using this method. After studying the most common RDH methods, we found that histogram shifting is a successful algorithm to be used for restoring the integer and non-integer data, compared to other RDH algorithms that are difficult to apply to non-integer data, such as graph data. In order to make a fair comparison we applied the same algorithm in [3] using the same parameters such as the same number of the secret bits and the same of the graph signals to demonstrate that the graph spectral domain has provide less embedding distortion, more robustness and recover the original signal with less error.

We selected the method in [4] for comparison with the proposed method for two reasons. First, we would like to compare the proposed method using histogram shifting with another RDH method using a different algorithm as in [4] RDH using a prediction algorithm. Second, RDH in encrypted images recently appeared as a promising research domain. As for RDH into clear images the correlation between image pixels is exploited, but the encryption makes the domain more challenging. This method [4] provides higher embedding bit-rates at lower distortion. Therefore, we considered this method is the best choose for comparison with the proposed method.

4.3.1 Experimental set up

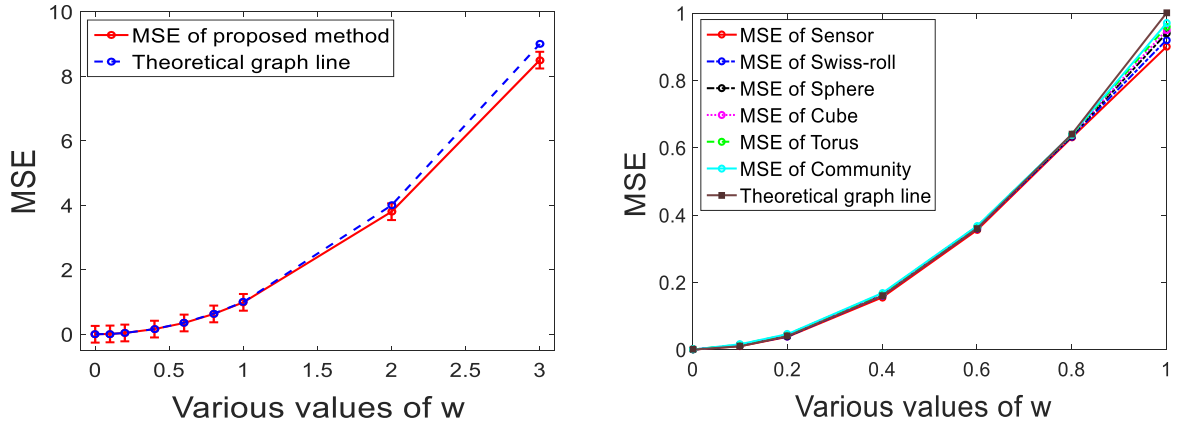
The proposed GFT reversible data hiding algorithm is tested using the graph watermarking dataset [181]. This dataset includes 160 various types of graphs with a different number of nodes and five graph signals as described in Section 3.3.1.

4.3.2 Verification of the embedding distortion model

The embedding distortion model has been verified in the empirical simulations. The MSE of the modified graph is less than the squared value of the embedded data, $\mathbf{w}^2$, where the shifting bin $q = w_b$ and w_b is the value of the embedded bit. The MSE of the modified graphs has been calculated for different values of $\mathbf{w}$, using the graph dataset. We consider the pseudo-random number sequences as the secret data, with nine scenarios, where $\mathbf{w} = \{0, 0.1, 0.2, 0.4, 0.6, 0.8, 1, 2, 3\}$, to embed in the GFT coefficients of the graph dataset. In these experiments, various graph types with several graph nodes ($N = 10000$ nodes) are used to embed various values of w. We have calculated a set of results to verify the effects of embedding nine scenarios of the secret data.

In these experiments, various values of $\mathbf{w}$ (where $\mathbf{w}$ is pseudo-random sequence of number or binary) are hidden in the peak points of the GFT coefficients. The MSE of the modified graphs has been calculated for each value of w separately. Figure 4.4 (a) demonstrates the relationship between the average value of the MSE of the modified various graph types for different values of $\mathbf{w}$: $\mathbf{w} = \{0, 0.1, 0.2, 0.4, 0.6, 0.8, 1, 2, 3\}$. We consider nine scenarios of embedding data to demonstrate the proposed embedding distortion model instead of using only two cases ($\mathbf{w} = \{0, 1\}$), which is insufficient to explain the proposed model. Note that the MSE value of the modified graph is less than the squared value, $\mathbf{w}^2$. We can see that the value of the MSE is zero, corresponding to the value of $w_b = 0$. Increasing the value of w_b increases the value of the MSE of the modified graph. To minimise the MSE, we must choose a small w_b value. Figure 4.4 (b) shows that the MSE value of the modified graph for each value of $\mathbf{w}$ is nearly the same for each graph; the difference is very small and cannot be distinguished.

To verify the proposed model, we plot the theoretical graph line by assuming that the x-axis represents various values of $\mathbf{w}$ and that the y-axis represents the corresponding quadratic values $\mathbf{w}^2$; the quadratic relationship is plotted for various values of $\mathbf{w}$. The results demonstrate a strong relationship between the proposed model and the theoretical graph line, which supports the proposed model. Notably, there is no distortion in the case of embedding $\mathbf{w} = \{0\}$ only. The simulation results demonstrate a strong correlation between the embedding value $\mathbf{w}^2$ and the embedding distortion, using the MSE of the modified graphs. The proposed model is supported by the simulation results for the graph dataset.



(a) (b)

Figure 4.4: Verification of distortion minimisation model using various values of w. (a) Average value of the MSE of the modified various types of graphs with $N = 10000$ nodes. (b) MSE of 6 types of graphs with $N = 10000$ nodes.

4.3.3 Performance evaluation of the embedding distortion

The performance of the proposed method is evaluated in terms of the embedding distortion at various embedding capacities using graph dataset. We test the proposed method using different types of graphs with various types of graph signals. Three sets of results are displayed to demonstrate the embedding distortion performance of the proposed method as following: In the experiment Set 1, we demonstrate the effect of the graph connectivity on the embedding distortion using various embedding capacities. In this experiments, we consider the secret data $\mathbf{w} = \{0.1,\ 0.2,\ 0.4,\ 0.6,\ 0.8,\ 1,\ 2,\ 3\}$ to embed in the GFT coefficients using various graphs types $\mathcal{G} = \{Torus,\ Sensor,\ Sphere,\ Cube,\ community,\ Swiss-roll\}$ and graph signal 1. The experimental results show that the graph type has an effect on the embedding distortion. We can notice that the Torus graph has a higher embedding capacity compared to the other types of graph in spite of using the same secret data as shown in Figure 4.5(a). This is mainly due to different types of connectivity present in various graphs. Lowest MSE value signifies less distortion.

In the experiment Set 2, we demonstrate the effect of using various values of $\mathbf{w}$ on the embedding distortion. In this experiments, we consider the secret data $\mathbf{w} = \{0.1,\ 0.2,\ 0.4,\ 0.6,\ 0.8,\ 1,\ 2,\ 3\}$ to embed in the GFT coefficients using various graphs types $\mathcal{G} = \{Torus,\ Sensor,\ Sphere,\ Cube,\ community,\ Swiss-roll\}$ and graph signal 2. The results illustrate the effect

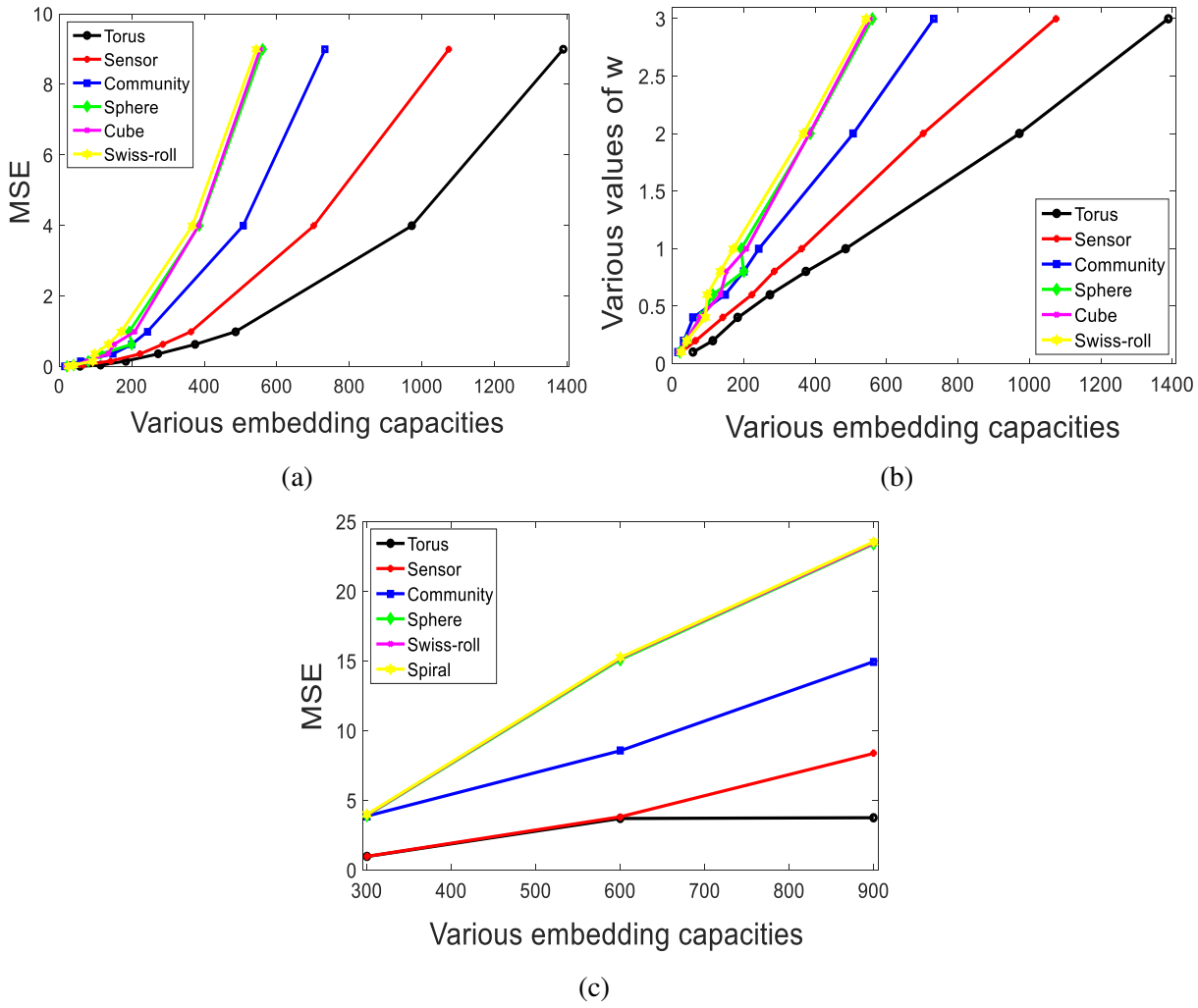


Figure 4.5: Embedding distortion performance. (a) MSE of the modified various graphs at various embedding capacities. (b) Relationship between the embedding capacity and the embedded data values of $\mathbf{w}$. (c) MSE of the modified various graphs using Multiple embedding.

of using different values of $\mathbf{w}$ on the embedding distortion for various graphs types. It can be observed that embedding capacity is increased when the value of $\mathbf{w}$ is increased. Increasing the value of $\mathbf{w}$ means increasing the value of the shifting bin q thus increasing the embedding capacity. Moreover, we can see the effect of the graph type on the embedding capacity. As shown in Figure 4.5(b) Torus graph has the highest embedding capacity compared to the other graphs types. Lowest value of $\mathbf{w}$ signifies less embedding capacity.

In the experiment Set 3, we show the effect of using multiple embedding (multiple shifting) on the performance of the proposed method. In this experiments, we consider the pseudo-random binary sequence as the secret bits $\mathbf{w} = \{0,\ 1\}$ to embed in the GFT coefficients using

various graphs types $\mathcal{G} = \{Torus,\ Sensor,\ Sphere,\ Spiral,\ community,\ Swiss-roll\}$ and graph signal 5. The experimental results demonstrate the effect of using multiple embedding on the embedding distortion. We can notice that the Torus graph has the lowest MSE value compared to the other types of graph in spite of using the same secret bits and the same embedding capacity as shown in Figure 4.5(c). This is basically due to the Torus graph needing a less number of shifting times for GFT coefficients (two times of shifting whereas the other graphs need more than two times of shifting).

The embedding rate is increased when histogram vacancies in the GFT coefficients are increased. However, the distortion due to hiding the secret data is increased. The proposed algorithm does not require to embed side information to recover the original graph data after extracted the secret data, it depends on reversing the shifting the GFT coefficients based on the shifting bin value.

4.3.4 Evaluation the robustness model

The robustness model is evaluated after the additive noise for various σ^2 values, where $\sigma^2 = \{0.0001,\ 0.0005,\ 0.001,\ 0.005,\ 0.01,\ 0.05\}$ in the experimental simulations. We calculate the Hamming Distance (HD) of the extracted bits using the reversible data hiding algorithm with the proposed robustness model (based on selecting the graph Fourier coefficients that satisfy the condition in Eq. (4.29) to embed the secret bits) and the Hamming Distance (HD) of the extracted bits using the reversible data hiding algorithm without the proposed robustness model (based on embedding the secret bits in any GFT coefficients randomly). Pseudo-random binary sequences are considered as the secret bits, $\mathbf{w} = \{0,\ 1\}$ to hide in the GFT coefficients of 7 graphs types with $N = 10000$ nodes. The results show that the method robustness is enhanced when using the proposed model. As illustrated in Figure 4.6, the proposed method has achieved higher robustness by an average of 15% over the original algorithm without using the robustness model.

4.3.5 Reversibility performance

We evaluate the proposed method in terms of the reversibility of the original graph signal after the embedded bits have been extracted for different embedding rates. The proposed method has proved that it is able to restore the original graph signal with a free distortion for any payload in the case when no attack. This is mainly due to shifting process which provides a lossless recovery of the original signal without requiring any side information except for one case if the histogram of the coefficients does not have zero points, at this case, the coefficients with the minimum number of the repetition are used as zero points and this leads to lose these coefficients due to the shifting process. In order to restore the original host signal without any error, these few numbers of the coefficients are added to the embedded data as a part of payload. Usually, this happens when the payload is big and this needs to shift the coefficients for many times, also to use several peaks and zero points. The proposed method overcomes this problem and the problems of the underflow and overflow of the data after the shifting process due to using

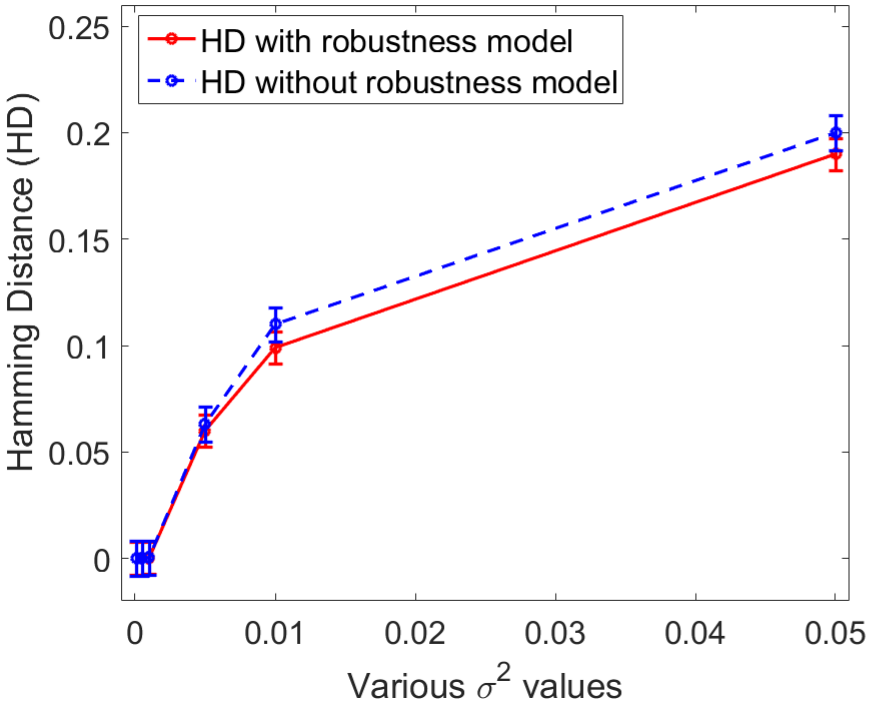


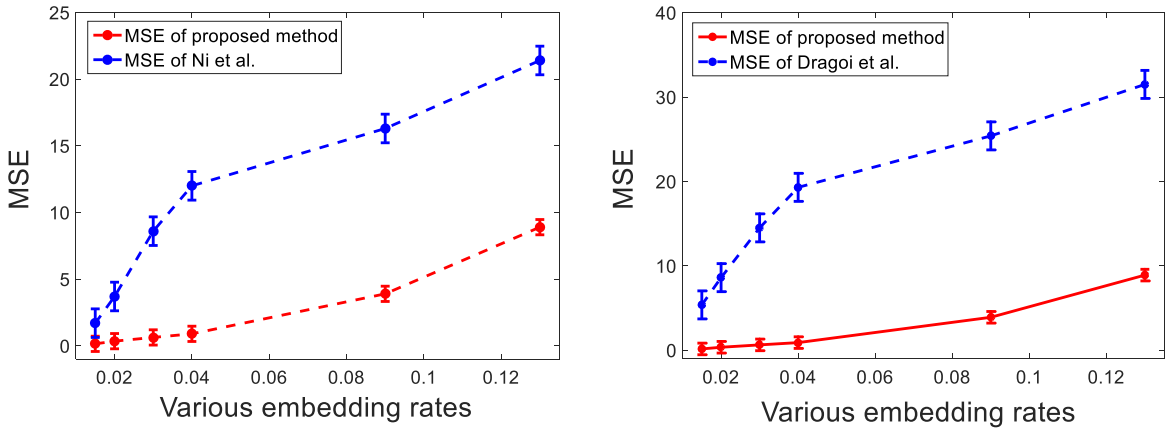
Figure 4.6: The average value of the Hamming Distance (HD) of the extracted bits for 7 graphs types with $N = 10000$ nodes after the additive noise for various values of σ^2.

graph Fourier transform, also by utilising the advantages of the histogram characteristics of the graph spectral coefficients which provides several peak points and zero points.

4.3.6 Comparison with existing methods

We compare the proposed method with two reversible data hiding methods namely, Ni et al. [3] and Dragoi et al. [4] in terms of the embedding distortion using the MSE for various embedding rates. We test the proposed method using various graph types (Sensor, Spiral, Swiss-roll, Sphere, Cube, Community, and Torus) with a different number of graph nodes ($N = \{5000, 10000\}$ nodes) and various graph signals. Figure 4.7 compares the average value of the MSE of the modified graph for the proposed method with that obtained by Ni et al. [3] and Dragoi et al. [4] from their methods, using the same graph signals for various embedding rates. The empirical simulations illustrate that the proposed method yields lower embedding distortion than the method used by Ni et al. [3], although it uses the same reversible data hiding algorithm based on histogram shifting for embedding the same secret bits $\mathbf{w} = \{0,\ 1\}$ for the same graph signals and the same embedding rates. This is due primarily to the advantage of using the graph Fourier domain. The embedding distortion is reduced more by using the proposed method than by using the methods of Ni et al. [3] and Dragoi et al. [4]. Moreover, the proposed method and Ni et al.'s [3] method can extract the embedding data accurately and without any error for any embedding rate, whereas the Dragoi et al. [4] method cannot extract the embedding data without error for embedding rates greater than 0.01. The results show that the proposed method outperforms the existing work by an average of 87% and 92% over Ni et al. [3] and Dragoi et al. [4] methods, respectively.

In addition, we test the capability of the embedded data to withstand attacks. We consider an additive noise using various values of noise variance (σ^2). In this experiment, we have calculated the average values of the Hamming Distance of the extracted secret bits $\mathbf{w} = \{0,\ 1\}$ for various σ^2 values and various signals, using the proposed method and those of Ni et al. [3] and Dragoi et al. [4]. Figure 4.8 compares the robustness of the proposed method and the existing methods, using the average values of the Hamming Distance of extracted secret data for different image data and various σ^2values. The secret data can withstand the additive noise when the noise variance σ^2 is less than 0.05. Our proposed method is more robust to the additive



(a) Ni et al. [3]

(b) Dragoi et al. [4]

Figure 4.7: Comparison the embedding distortion of the proposed method with existing work for various embedding rates. (a) Ni et al. [3]. (b) Dragoi et al. [4].

noise than the existing methods. The proposed method outperforms the existing work by an average of 54% and 86% over Ni et al. [3] and Dragoi et al. [4] methods, respectively.

We have compared the reversibility of the proposed method to the reversible data hiding algorithm [3] and the Dragoi et al. [4] method in the case of no attack. In these experiments, we have calculated the MSE between the recovered data and the original data using various graph signals at different embedding rates. Figure 4.9 (a) shows the average MSE values of the

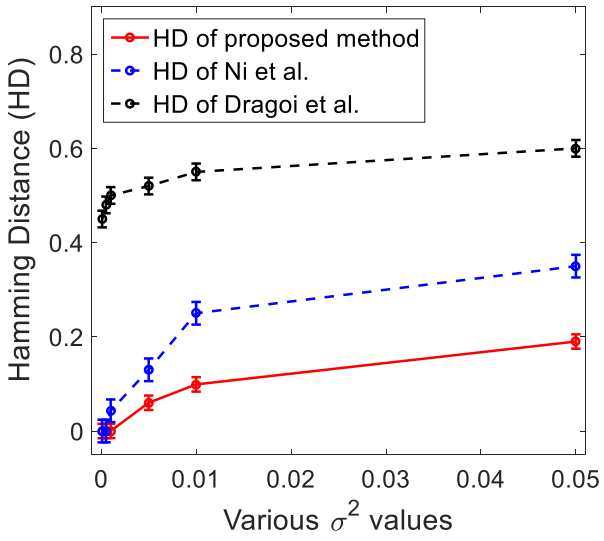
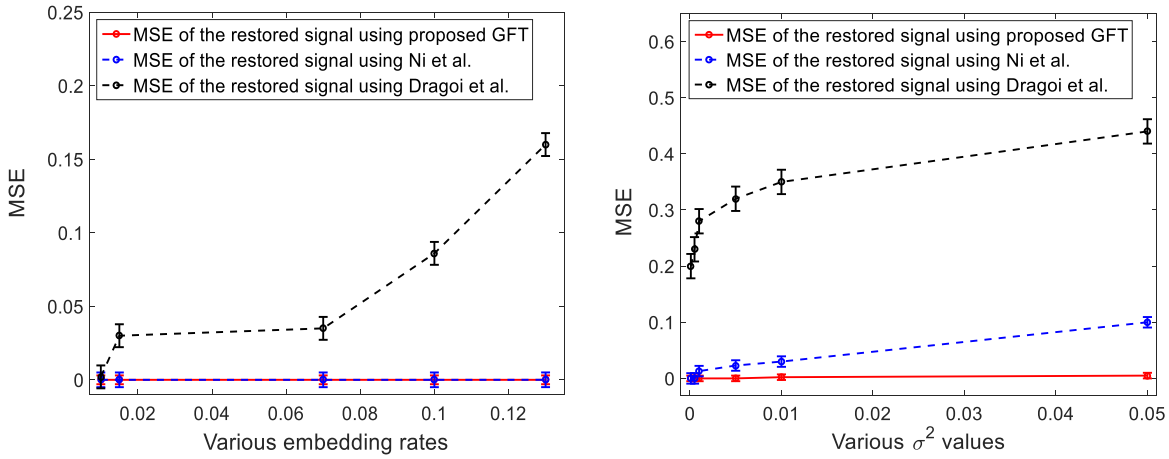


Figure 4.8: Comparison the Hamming Distance (HD) of the proposed method with the Ni et al. [3] and Dragoi et al. [4] after the additive noise for various values of σ^2.

proposed method and the existing methods in the case of no attack, using the same graph signals data at various payloads. The proposed method recovers the original signal without any error for any embedding rate. RDH using histogram shifting [3] can also recover the original data error-free when the embedding rate is low. In contrast, Dragoi et al.'s method can recover the original data with distortion when the embedding rate is greater than 0.01. Figure 4.9 (b) shows the average MSE value of the proposed method and the existing methods after the additive noise for various values of σ^2, using the same graph signals. The proposed method outperforms the existing work by an average of 97% and 99% over Ni et al. [3] and Dragoi et al. [4] methods, respectively.



(a) (b)

Figure 4.9: Comparison the performance of reversibility of the proposed reversible data hiding using GFT with Ni et al. [3] and Dragoi et al. [4]. (a) Without additive noise. (b) After the additive noise for various values of σ^2.

4.4 Concluding remarks

In this chapter, we have proposed a graph Fourier domain reversible data hiding approach based on the histogram shifting. Two new models are proposed to minimise the embedding distortion and to make the embedded data robust to additive noise. The embedding distortion is reduced based on selecting the smallest value of w. Moreover, the robustness of the proposed method is improved by selecting the GFT coefficents that satisfy spesific conditions. We have evaluated the proposed method in terms of embedding distortion, reversibility of the original graph

signal and the robustness against additive noise. The proposed method has compared with two previous reversible data hiding, namely, Ni et al. [3] and Dragoi et al. [4] methods. The empirical simulations illustrate that the proposed approach has improved the distortion by an average of 87% and 92% compared to [3] and Dragoi et al. [4] methods, respectively in terms of embedding distortion. The robustness of the proposed method is enhanced to additive noise by an average of 54% and 86% over the previous methods. The proposed method can restore the original graph signal and the hidden secret data without any error for any embedding rates compared to Dragoi et al. method [4] which extracts the embedded bits and the recovered images data with errors in the case of no attacks. The proposed method retrieves the original graph signal without any error, therefore, it does not need side information to correct the error in the recovered signal as in existing work. In addition, the proposed method has evaluated in terms of restoring the original host signal after the additive noise. The results show that the proposed method provides lower distortion by an average of 97% and 99% compared to Ni et al. [3] and Dragoi et al. [4] methods, respectively. The next chapter includes the graph wavelet domain data hiding for graph data.

Chapter 5

Graph wavelet domain data hiding for graph data

5.1 Introduction

The previous two chapters proposed irreversible data hiding and reversible data hiding approaches for graph data in the graph Fourier domain. This chapter proposes data hiding approaches including irreversible data hiding and reversible data hiding using graph wavelet transform. Discrete wavelet transform is considered as a powerful tool in signal processing due to its ability to localise the contents of the signal in the time and frequency domains. It provides multi-scale representations of the signal. These reasons make the discrete wavelet transform a typical choice for data hiding in multimedia [21]. This chapter exploits the wavelet transform advantages to propose new data hiding approaches for unstructured data in graph wavelet domain.

Recently, we have seen an increase in applications that represent their data as weighted graphs like sensor networks and social networks.This creates a strong need for protecting these data. The most common approaches of graph data hiding depend on modifying the graph topology as illustrated in chapter 3. These approaches are not robust and insecure because they are based on the vertex domain. In reversible data hiding, the existing methods of graph reversible data hiding relies on a mesh. In general, there are four groups based on the embedding domain: vertex domain, compressed domain, transform domain and encrypted domain as we explained

in chapter 4.

On the other hand, the graph Fourier domain data hiding has proven to be a very effective approach to protect the graph data for irreversible and reversible data hiding. This is due to advances in signal transforms as shown in chapter 3 and chapter 4, respectively. This chapter proposes two data hiding methods in graph wavelet domain with new models, namely, an embedding distortion minimisation model to minimise the embedding distortion in the modified graph and a robustness model to make the secret data robust to attacks. Finally, the conditions of the proposed models are combined for satisfying the basic requirements of the data hiding system. The main contributions of this chapter are:

1. Proposing a distortion minimisation model and a robustness model for graph wavelet domain irreversible data hiding.

2. Proposing a new graph wavelet domain histogram shifting algorithm for reversible data hiding on non-integer data.

3. Proposing new models to minimise the embedding distortion in host graph data after embedding and to make the embedding robust to additive noise.

The rest of this chapter is organised as the following: Section 5.2 introduces the proposed methodology including the proposed data hiding in graph wavelet domain, followed by the proposed models, embedding distortion minimisation and robustness enhancing, respectively. The performance of the proposed methods are evaluated using experimental results in Section 5.3. We finally present the concluding remarks in Section 5.4.

5.2 Proposed Methodology

This section presents the proposed data hiding methods using graph wavelet transform. We consider two data hiding methods, namely, irreversible and reversible data hiding.

5.2.1 Graph Wavelet Transform (GWT)

We suppose that $\mathcal{G}$ is an undirected graph without self-loops and multiple edges between nodes, the adjacency matrix with edge weights, $\mathbf{A}$, is defined as in Eq. (2.9). The combinatorial graph

Laplacian matrix, $\mathbf{L}$, is calculated as in Eq. (2.11). The Graph Fourier Transform (GFT) and its inverse are defined as in Eq. (2.17) and Eq. (2.18). The graph wavelet coefficients are calculated using Eq. (2.26).

5.2.2 Proposed methodology of irreversible data hiding

In this section, we present the proposed irreversible data hiding method using graph wavelet transform. We consider two embedding scenarios: non-blind and blind data hiding. Figure 5.1 shows the block diagram of the proposed method. The proposed methodology includes two new models, i.e. embedding distortion minimisation model and robustness model to minimise the embedding distortion and to enhance the robustness against the attacks, respectively.

5.2.2.1 GWT domain data hiding

5.2.2.1.1 Non-blind data hiding

We propose a non-blind algorithm using magnitude based multiplicative watermarking [13]. We firstly calculate the graph wavelet coefficients using Eq. (2.26), then the low-frequency GWT

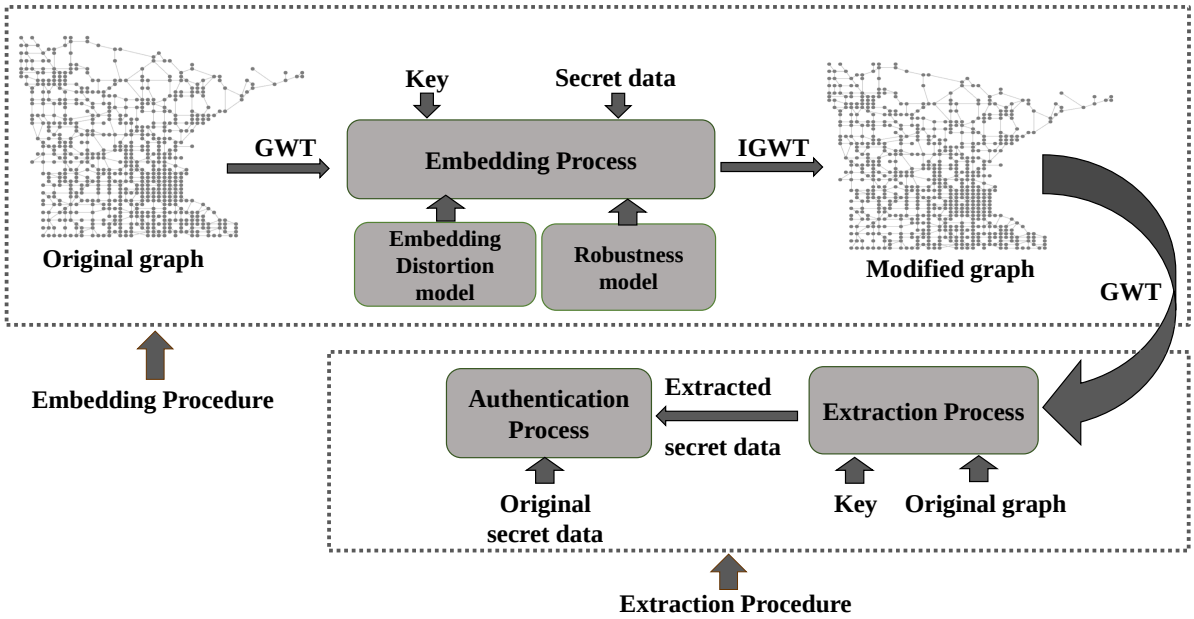


Figure 5.1: The block diagram of the proposed graph wavelet domain data hiding.

coefficients $\mathbf{Y}$ are selected to embed the secret bits as follows:

$$\mathbf{Y}_w = \mathbf{Y}(1 + \alpha w_b), \tag{5.1}$$

where $\mathbf{Y}_w$ is the modified graph wavelet coefficient, α is the data hiding parameter and w_b is the secret bit. The inverse GWT is performed on the modified GWT coefficients to obtain the modified graph.

The extraction process requires the original coefficients to extract the secret bits. The embedded bits are extracted based on the embedding key which is sent to the receiver in a separate file. The embedding key includes w_0, w_1, length of the secret bits and α. Firstly, the GWT is performed on the modified graph, then the secret bits w'_b are extracted from the low-frequency GWT coefficients as follows:

$$w'_b = \frac{\mathbf{Y}_w - \mathbf{Y}}{\alpha \mathbf{Y}}, \tag{5.2}$$

where $w'_b (b = \{0, 1\})$ is the extracted bit. Let w_0 and w_1 are the selected secret bits values for embedding a 0 and 1, respectively. The extracted secret bit b' is determined depend on a threshold T where $T = (w_0 + w_1)/2$, whereas:

$$b' = \begin{cases} 0 \quad , & \text{if } \ w'_b < T, \\ 1 \quad , & \text{if } w'_b > T. \end{cases} \tag{5.3}$$

5.2.2.1.2 Blind data hiding

A blind algorithm is proposed using a prediction-based graph data hiding. Firstly, the graph wavelet coefficients are calculated using Eq. (2.26). Then, the selected GWT coefficients Y are sorted in descending order, $\mathbf{Y}_s(m)$. After that, a non-overlapping 3×1 running window is passed through the selected GWT coefficients for hiding the secret bit in the median graph wavelet coefficient at each sliding position, as the following:

$$\mathbf{Y}_{s_w}(m) = \left\lfloor \frac{\mathbf{Y}_s(m-1) + \mathbf{Y}_s(m+1)}{2} \right\rfloor + w_b, \tag{5.4}$$

where $\mathbf{Y}_{s_w}$ is the modified graph wavelet coefficient, $\lfloor \mathbf{Y} \rfloor$ refers to rounding of $\mathbf{Y}$ to the largest integer value smaller than $\mathbf{Y}$ and $w_b > 0$ is the secret bit. For extraction the secret bits without error, the embedding is restricted for any three GWT coefficients, if and only if they satisfy following condition:

$$\mathbf{Y}_s(m-1) \geq \Big\lfloor \frac{\mathbf{Y}_s(m-1) + \mathbf{Y}_s(m+1)}{2} \Big\rfloor + w \geq \mathbf{Y}_s(m+1). \tag{5.5}$$

Then, the spectral coefficients are used in the embedding process; otherwise (if the condition does not satisfy) the first coefficient is skipped. We start from the second coefficient and check the three coefficients again and so on. We use a code (0) for no skip coefficient and (1) for skipping coefficients followed by the locations of the coefficients. The secret key includes this information and it is sent to the receiver separately. Finally, we perform the inverse GWT on the modified GWT coefficients to obtain the modified graph signal.

In the extraction process, the receiver extract the embedded bits based on the embedding key. The embedding key includes w_0, w_1, length of the secret bits, number of skipped coefficients and the positions of the skipped coefficients. The GWT is applied on the modified graph signal, followed by sorting in descending order, to get sorted modified GWT coefficients, $\mathbf{Y}_w(m)$. Then the secret bit from each 3×1 running window with coefficients, $\mathbf{Y}_w(m-1) \geq \mathbf{Y}_w(m) \geq \mathbf{Y}_w(m+1)$, is extracted based on the secret key as given:

$$w'_b = \mathbf{Y}_w(\ell) - \lfloor \frac{\mathbf{Y}_w(\ell-1) + \mathbf{Y}_w(\ell+1)}{2} \rfloor. \tag{5.6}$$

where $w'_b (b = \{0, 1\})$ is the extracted bit. Let w_0 and w_1 are the selected secret bits values for embedding a 0 and 1 , respectively. The extracted secret bit b' is determined according to a threshold T, where $T = (w_0 + w_1)/2$, as shown in Eq. (5.3).

5.2.2.2 Authentication Process

We perform the authentication based on comparing the extracted secret bits with the original secret bits using the Hamming Distance (HD) as defined as in Eq. (2.6).

5.2.2.3 Embedding distortion minimisation

A new model is proposed to minimise the embedding distortion in the graph wavelet domain. The proposed model establishes the relationship between the error distortion using mean square error (μ) and the chosen graph wavelet coefficients for data hiding. We consider two cases: orthogonal wavelet bases and non-orthogonal wavelet bases. The models are proposed based on following the models in [21].

5.2.2.3.1 Embedding distortion minimisation model for orthogonal graph wavelet filters

We propose a new model for reducing the embedding distortion in the GWT coefficients after hiding the secret bits. The performance of the embedding distortion is measured using the MSE (μ). We define mean square error (μ) in vertex domain between the original graph signal $\mathbf{x}$ and modified graph signal $\mathbf{x}_w$ is defined as the following:

$$\mu = \frac{1}{N}\sum_{i=0}^{N-1}(\mathbf{x}(i) - \mathbf{x}_w(i))^2, \tag{5.7}$$

where N is the number of graph nodes. Since the wavelet bases are orthogonal [2], the energy between an input graph signal and the graph wavelet coefficients is conserved according to the Parseval's Theorem which means:

$$\|\mathbf{x}\|^2 = \|\mathbf{Y}\|^2, \tag{5.8}$$

where $\mathbf{x}$ is the signal of graph in vertex domain and $\mathbf{Y}$ is the GWT coefficient. This can be extended to the sum of the error power in the input signal of graph, $\Delta\mathbf{x}$, and to the sum of the error power in the graph wavelet domain $\Delta\mathbf{Y}$ as the following:

$$\sum_i |\Delta\mathbf{x}(i)|^2 = \sum_\ell |\Delta\mathbf{Y}(\ell)|^2. \tag{5.9}$$

From Eq. (5.7) and Eq. (5.9), we get

$$\mu = \frac{1}{N}\sum_\ell |\Delta\mathbf{Y}(\ell)|^2. \tag{5.10}$$

We suggest two data hiding scenarios: non-blind and blind.

Proposition 5.1 (Non-blind)

For non-blind approach, the MSE (μ) of the modified graph is proportional to the energy sum of selected GWT coefficients:

$$\mu \propto \sum_{\ell=0}^{N-1} |\mathbf{Y}(\ell)|^2. \tag{5.11}$$

Proof. In non-blind approach, the modified coefficients $\mathbf{Y}_w(\ell)$ are calculated as follows:

$$\begin{aligned} \mathbf{Y}_w(\ell) &= \mathbf{Y}(\ell) + \mathbf{Y}(\ell)\alpha w_b, \\ \mathbf{Y}_w(\ell) - \mathbf{Y}(\ell) &= \mathbf{Y}(\ell)\alpha w_b, \\ \Delta\mathbf{Y}(\ell) &= \mathbf{Y}(\ell)\alpha w_b, \end{aligned}$$

where $\Delta\mathbf{Y}(\ell)$ is the value of modification owing to hiding the secret bits. From Eq. (5.10), thereby leading to the the relationship between the MSE of modified graph and the chosen GWT coefficients:

$$\mu \propto \sum_{\ell=0}^{N-1} |\mathbf{Y}(\ell)|^2. \tag{5.12}$$

□

Proposition 5.2 (Blind)

In a blind approach, for any embedding GWT coefficient triple $\mathbf{Y_s}(m-1) \geq \mathbf{Y_s}(m) \geq \mathbf{Y_s}(m+1)$, the MSE (μ) of the modified graph is proportional to the gradient difference of the embedding coefficient triple $[(\mathbf{Y_s}(m-1) - \mathbf{Y_s}(m)) - (\mathbf{Y_s}(m) - \mathbf{Y_s}(m+1))]$ as follows:

$$\mu \propto [(\mathbf{Y_s}(m-1) - \mathbf{Y_s}(m)) - (\mathbf{Y_s}(m) - \mathbf{Y_s}(m+1))]. \tag{5.13}$$

Proof. For any three sorted spectral coefficients, $\mathbf{Y_s}(m-1) \geq \mathbf{Y_s}(m) \geq \mathbf{Y_s}(m+1)$, the modification value due to embedding the secret bits $\Delta\mathbf{Y_s}(m)$ using the prediction algorithm is

estimated from Eq. (5.4) as follows:

$$\begin{aligned}\Delta\mathbf{Y_s}(m) &= \mathbf{Y}_{sw}(m) - \mathbf{Y}_s(m),\\ &= \Big\lfloor \frac{\mathbf{Y_s}(m-1) + \mathbf{Y_s}(m+1)}{2} \Big\rfloor + w - \mathbf{Y_s}(m),\\ \Delta\mathbf{Y_s}(m) &= \Big\lfloor \frac{\mathbf{Y_s}(m-1) + \mathbf{Y_s}(m+1)}{2} \Big\rfloor - \mathbf{Y_s}(m).\end{aligned}$$

By substituting $\mathbf{Y}_s(m-1)$ with $\mathbf{Y}_s(m) + \Delta_1$ and $\mathbf{Y}_s(m+1)$ with $\mathbf{Y}_s(m) - \Delta_2$, based on the sorted coefficients, $\mathbf{Y_s}(m) + \Delta_1 \geq \mathbf{Y_s}(m) \geq \mathbf{Y_s}(m) - \Delta_2$.

$$\Delta\mathbf{Y_s}(m) = \Big\lfloor \frac{\mathbf{Y_s}(m-1) + \mathbf{Y_s}(m+1)}{2} \Big\rfloor - \mathbf{Y_s}(m),$$

$$\Delta\mathbf{Y_s}(m) = \Big\lfloor \frac{\mathbf{Y_s}(m) + \Delta_1 + \mathbf{Y_s}(m) - \Delta_2}{2} \Big\rfloor - \mathbf{Y_s}(m).$$

The minimum error distortion is obtained when the difference between $\mathbf{Y}_{s_w}(m)$ and $\mathbf{Y}_s(m)$ is close to 0:

$$\Big\lfloor \frac{\mathbf{Y_s}(m-1) + \mathbf{Y_s}(m+1)}{2} \Big\rfloor - \mathbf{Y_s}(m) = 0,$$

$$\Big\lfloor \frac{\mathbf{Y_s}(m) + \Delta_1 + \mathbf{Y_s}(m) - \Delta_2}{2} \Big\rfloor - \mathbf{Y_s}(m) = 0,$$

$$\mathbf{Y_s}(m) + \Delta_1 + \mathbf{Y_s}(m) - \Delta_2 = 2\mathbf{Y_s}(m),$$

$$(\mathbf{Y_s}(m) + \Delta_1) - \mathbf{Y_s}(m) = \mathbf{Y_s}(m) - (\mathbf{Y_s}(m) - \Delta_2),$$

$$[(\mathbf{Y_s}(m) + \Delta_1) - \mathbf{Y_s}(m)] - [\mathbf{Y_s}(m) - (\mathbf{Y_s}(m) - \Delta_2)] = 0.$$

Since the wavelet bases are orthogonal and from Eq. (5.10) we obtain:

$$\mu \propto \sum |\Delta\mathbf{Y_s}(m)|^2. \tag{5.14}$$

Thereby leading to

$$\mu \propto \sum (\lfloor \frac{\mathbf{Y_s}(m-1) + \mathbf{Y_s}(m+1)}{2} \rfloor - \mathbf{Y_s}(m))^2.$$

Thereby leading to the relationship between the MSE of modified graph (μ) and the chosen GWT coefficient triple:

$$\mu \propto (\mathbf{Y_s}(m-1) - \mathbf{Y_s}(m)) - (\mathbf{Y_s}(m) - \mathbf{Y_s}(m+1)).$$

Therefore for minimizing μ, for each hiding GWT coefficient triple, $\lfloor 0.5(\mathbf{Y_s}(m-1) + \mathbf{Y_s}(m+1))\rfloor - \mathbf{Y_s}(m)$ should be close to 0 or in other words the gradient difference, $[(\mathbf{Y_s}(m-1) - \mathbf{Y_s}(m)) - (\mathbf{Y_s}(m) - \mathbf{Y_s}(m+1))]$ should be close to 0. □

5.2.2.3.2 Embedding distortion minimisation model for non-orthogonal graph wavelet filters

The energy between an input graph signal and the graph wavelet coefficients is not conserved when the wavelet bases are non-orthogonal as follows [25]:

$$c_1 \sum_{\ell} |\mathbf{Y}(\ell)|^2 \leq \|\mathbf{x}\|^2 \leq c_2 \sum_{\ell} |\mathbf{Y}(\ell)|^2, \tag{5.15}$$

where c_1 and c_2 are the orthonormality correction factor.

Since the wavelet filter is not orthogonal, in this case, the transform coefficients have to satisfy the Eq. (5.15). This leads to:

$$\|\mathbf{x}\|^2 = R\|\mathbf{Y}\|^2, \tag{5.16}$$

where $\mathbf{x}$ is the signal of graph in vertex domain, $\mathbf{Y}$ is the GWT coefficient and R is a weighting factor as given:

$$R = \frac{\|\mathbf{x}\|^2}{\sum_{\ell} |\mathbf{Y}(\ell)|^2}. \tag{5.17}$$

This can be extended to the sum of the error power in the input signal of graph, $\Delta\mathbf{x}$, and to the sum of the error power in the graph wavelet domain $\Delta\mathbf{Y}$ as the following:

$$\sum_{i} |\Delta\mathbf{x}(i)|^2 = R \sum_{\ell} |\Delta\mathbf{Y}(\ell)|^2. \tag{5.18}$$

From Eq. (5.7) and Eq. (5.18), we get

$$\mu = \frac{1}{N}(R^{\eta\gamma} \sum_{\ell} |\Delta \mathbf{Y}(\ell)|^2), \tag{5.19}$$

where $R^{\eta\gamma}$ is the weighting parameter at η sub-band and γ decomposition level.

Proposition 5.3 (Non-blind)

For non-blind approach, the MSE (μ) of the modified graph is proportional to the weighted energy sum of chosen GWT coefficients:

$$\mu \propto R^{\eta\gamma} \sum_{\ell=0}^{N-1} |\mathbf{Y}(\ell)|^2. \tag{5.20}$$

Proof. In non-blind approach, the modified coefficients $\mathbf{Y}_w(\ell)$ are calculated as follows:

$$\begin{aligned} \mathbf{Y}_w(\ell) &= \mathbf{Y}(\ell) + \mathbf{Y}(\ell)\alpha w_b, \\ \mathbf{Y}_w(\ell) - \mathbf{Y}(\ell) &= \mathbf{Y}(\ell)\alpha w_b, \\ \Delta \mathbf{Y}(\ell) &= \mathbf{Y}(\ell)\alpha w_b, \end{aligned}$$

where $\Delta \mathbf{Y}(\ell)$ is the value of modification owing to embedding the secret data. From Eq. (5.19), thereby leading to the the relationship between the MSE of the modified graph and the chosen GWT coefficients:

$$\mu \propto R^{\eta\gamma} \sum_{\ell=0}^{N-1} |\mathbf{Y}(\ell)|^2.$$

□

Proposition 5.4 (Blind)

In a blind approach, for any embedding GWT coefficient triple $\mathbf{Y_s}(m-1) \geq \mathbf{Y_s}(m) \geq \mathbf{Y_s}(m+1)$, the MSE ($\mu$) of the modified graph is proportional to the gradient difference of the embedding coefficient triple $[(\mathbf{Y_s}(m-1) - \mathbf{Y_s}(m)) - (\mathbf{Y_s}(m) - \mathbf{Y_s}(m+1))]$ as follows:

$$\mu \propto R^{\eta\iota}[(\mathbf{Y_s}(m-1) - \mathbf{Y_s}(m)) - (\mathbf{Y_s}(m) - \mathbf{Y_s}(m+1))]. \tag{5.21}$$

Proof. For any three sorted spectral coefficients, $\mathbf{Y_s}(m-1) \geq \mathbf{Y_s}(m) \geq \mathbf{Y_s}(m+1)$, the modification value due to embedding the secret data $\Delta\mathbf{Y_s}(m)$ using the prediction algorithm is estimated from Eq. (5.4) as follows:

$$\begin{aligned}\Delta\mathbf{Y_s}(m) &= \mathbf{Y}_{sw}(m) - \mathbf{Y}_s(m),\\ &= \Big\lfloor \frac{\mathbf{Y_s}(m-1) + \mathbf{Y_s}(m+1)}{2} \Big\rfloor + w - \mathbf{Y_s}(m),\\ \Delta\mathbf{Y_s}(m) &= \Big\lfloor \frac{\mathbf{Y_s}(m-1) + \mathbf{Y_s}(m+1)}{2} \Big\rfloor - \mathbf{Y_s}(m).\end{aligned}$$

By substituting $\mathbf{Y}_s(m-1)$ with $\mathbf{Y}_s(m) + \Delta_1$ and $\mathbf{Y}_s(m+1)$ with $\mathbf{Y}_s(m) - \Delta_2$, based on the sorted coefficients, $\mathbf{Y_s}(m) + \Delta_1 \geq \mathbf{Y_s}(m) \geq \mathbf{Y_s}(m) - \Delta_2$.

$$\Delta\mathbf{Y_s}(m) = \Big\lfloor \frac{\mathbf{Y_s}(m-1) + \mathbf{Y_s}(m+1)}{2} \Big\rfloor - \mathbf{Y_s}(m),$$

$$\Delta\mathbf{Y_s}(m) = \Big\lfloor \frac{\mathbf{Y_s}(m) + \Delta_1 + \mathbf{Y_s}(m) - \Delta_2}{2} \Big\rfloor - \mathbf{Y_s}(m).$$

The minimum error distortion is obtained when the difference between $\mathbf{Y}_{s_w}(m)$ and $\mathbf{Y}_s(m)$ is close to 0:

$$\Big\lfloor \frac{\mathbf{Y_s}(m-1) + \mathbf{Y_s}(m+1)}{2} \Big\rfloor - \mathbf{Y_s}(m) = 0,$$

$$\Big\lfloor \frac{\mathbf{Y_s}(m) + \Delta_1 + \mathbf{Y_s}(m) - \Delta_2}{2} \Big\rfloor - \mathbf{Y_s}(m) = 0,$$

$$\mathbf{Y_s}(m) + \Delta_1 + \mathbf{Y_s}(m) - \Delta_2 = 2\mathbf{Y_s}(m),$$

$$(\mathbf{Y_s}(m) + \Delta_1) - \mathbf{Y_s}(m) = \mathbf{Y_s}(m) - (\mathbf{Y_s}(m) - \Delta_2),$$

$$[(\mathbf{Y_s}(m) + \Delta_1) - \mathbf{Y_s}(m)] - [\mathbf{Y_s}(m) - (\mathbf{Y_s}(m) - \Delta_2)] = 0.$$

Since the wavelet bases are non-orthogonal bases and from Eq. (5.19) we obtain:

$$\mu \propto R^{\eta\gamma} \sum |\Delta\mathbf{Y_s}(m)|^2. \tag{5.22}$$

Thereby leading to

$$\mu \propto R^{\eta\gamma} \sum (\lfloor \frac{\mathbf{Y_s}(m-1) + \mathbf{Y_s}(m+1)}{2} \rfloor - \mathbf{Y_s}(m))^2.$$

Thereby leading to the relationship between the MSE of the modified (μ) and the chosen GWT coefficient triple:

$$\mu \propto R^{\eta\gamma}(\mathbf{Y_s}(m-1) - \mathbf{Y_s}(m)) - (\mathbf{Y_s}(m) - \mathbf{Y_s}(m+1)).$$

Therefore for minimizing μ, for each hiding GWT coefficient triple, $\lfloor 0.5(\mathbf{Y_s}(m-1) + \mathbf{Y_s}(m+1)) \rfloor - \mathbf{Y_s}(m)$ should be close to 0 or in other words the gradient difference, $[(\mathbf{Y_s}(m-1) - \mathbf{Y_s}(m)) - (\mathbf{Y_s}(m) - \mathbf{Y_s}(m+1))]$ should be close to 0. □

5.2.2.4 On enhancing robustness

The proposed model aims to identify the graph wavelet coefficients that are able to extract the secret data accurately after the attack in the graph wavelet domain. We consider two data hiding scenarios, i.e. non-blind and blind to analyse the robustness against the attacks, namely, noise addition and deleting nodes data. The modified GWT coefficients values, $\mathbf{Y}_w(\ell)$, are adjusted based on the modification value due to attack Δ_a as follows:

$$\mathbf{X}'_w(\ell) = \mathbf{Y}_w(\ell) + \Delta_a, \tag{5.23}$$

where $\mathbf{Y}'_w(\ell)$ are the modified graph wavelet coefficients values after the attack. The value of modification due to attack Δ_a can be in the range:

$$\Delta_{a_{min}} \leq \Delta_a \leq \Delta_{a_{max}}, \tag{5.24}$$

where $\Delta_{a_{min}}$ and $\Delta_{a_{max}}$ are the minimum and maximum modifications values. The modification value Δ_a depends on the type of attack. For instance, the value of modification due to additive noise depends on the noise variance σ^2 , while the value of modification of deleting nodes data depends on the number of the node data which are deleting.

5.2.2.4.1 Non-blind model

A model is derived to show the relationship between the selected coefficients to hide the secret bits and the robustness against attacks. The basic form of the data embedding in the non-blind approach is:

$$\mathbf{Y}_w(\ell) = \mathbf{Y}(\ell) + \Delta, \tag{5.25}$$

where $\mathbf{Y}(\ell)$ is the GWT coefficient to be modified, $\mathbf{Y}_w(\ell)$ is the modified coefficient and Δ is the modification value due to hiding the secret data.

$$\Delta = \mathbf{Y}_w(\ell) - \mathbf{Y}(\ell), \tag{5.26}$$

$$\Delta = \alpha \mathbf{Y}(\ell) w, \tag{5.27}$$

where α is the data hiding parameter and $w_b (b = \{0, 1\})$ is the secret bit. Based on substituting the Eq. (5.26) in Eq. (5.25), we get:

$$\mathbf{Y}_w(\ell) = \mathbf{Y}(\ell) + \alpha \mathbf{Y}(\ell) w_b, \tag{5.28}$$

$$= \mathbf{Y}(\ell)(1 + \alpha w_b).$$

The relationship between the original graph wavelet coefficient and modified graph Fourier coefficient is:

$$\mathbf{Y}(\ell) = \frac{\mathbf{Y}_w(\ell)}{1 + \alpha w_b}. \tag{5.29}$$

The secret bit w'_b is extracted as follows:

$$w'_b = \frac{\mathbf{Y}_w(\ell) - \mathbf{Y}(\ell)}{\alpha \mathbf{Y}(\ell)}. \tag{5.30}$$

Let $\mathbf{Y}'_w(\ell)$ be the reconstructed modified coefficient after the attack, Δ_a is the modification value due to attack, we get:

$$\mathbf{Y}'_w(\ell) = \mathbf{Y}_w(\ell) + \Delta_a, \tag{5.31}$$

$$\Delta_a = \mathbf{Y}'_w(\ell) - \mathbf{Y}_w(\ell). \tag{5.32}$$

Proposition 5.5

The original GWT coefficients for hiding a bit value $b = 1$ and retain intact after the attacks are in the range:

$$\frac{\mathbf{Y}'_w(\ell')}{1+\alpha w_1} \leq \mathbf{Y}(\ell) \leq \frac{\mathbf{Y}'_w(\ell')}{1+\alpha T}. \tag{5.33}$$

This proposition was proved in Chapter 3.

Proposition 5.6

The original GWT coefficients for hiding a bit value $b = 0$ and retain intact after the attacks are in the range:

$$\frac{\mathbf{Y}'_w(\ell')}{1+\alpha T} < \mathbf{Y}(\ell) < \frac{\mathbf{Y}'_w(\ell')}{1+\alpha w_0}. \tag{5.34}$$

We proved this proposition in Chapter 3.

Finally, we combine proposition 5.5 and proposition 5. 6 to find the region of coefficients that are capable of retaining both $b = 1$ and $b = 0$ after the attacks The original GWT coefficients which can retain the correct secret bit should be in the range:

$$\frac{\mathbf{Y}'_w(\ell')}{1+\alpha w_1} \leq \mathbf{Y}(\ell) \leq \frac{\mathbf{Y}'_w(\ell')}{1+\alpha w_0}. \tag{5.35}$$

In the case of no attack, the modified coefficient $\mathbf{Y}_w(\ell)$ after embedding the secret bit $b = 0 \; and \; b = 1$ will be in the range:

$$\mathbf{Y}(\ell)(1+\alpha w_0) \; \leq \mathbf{Y}_w(\ell) \; \leq \mathbf{Y}(\ell)(1+\alpha w_1). \tag{5.36}$$

And the secret bit can be extracted accurately when the graph wavelet coefficients in the range:

$$\frac{\mathbf{Y}_w(\ell)}{1+\alpha w_1} \leq \mathbf{Y}(\ell) \leq \frac{\mathbf{Y}_w(\ell)}{1+\alpha w_0}. \tag{5.37}$$

Figure 5.2 displays the range of the graph wavelet coefficients that is able to retain the secret bits after the attacks.

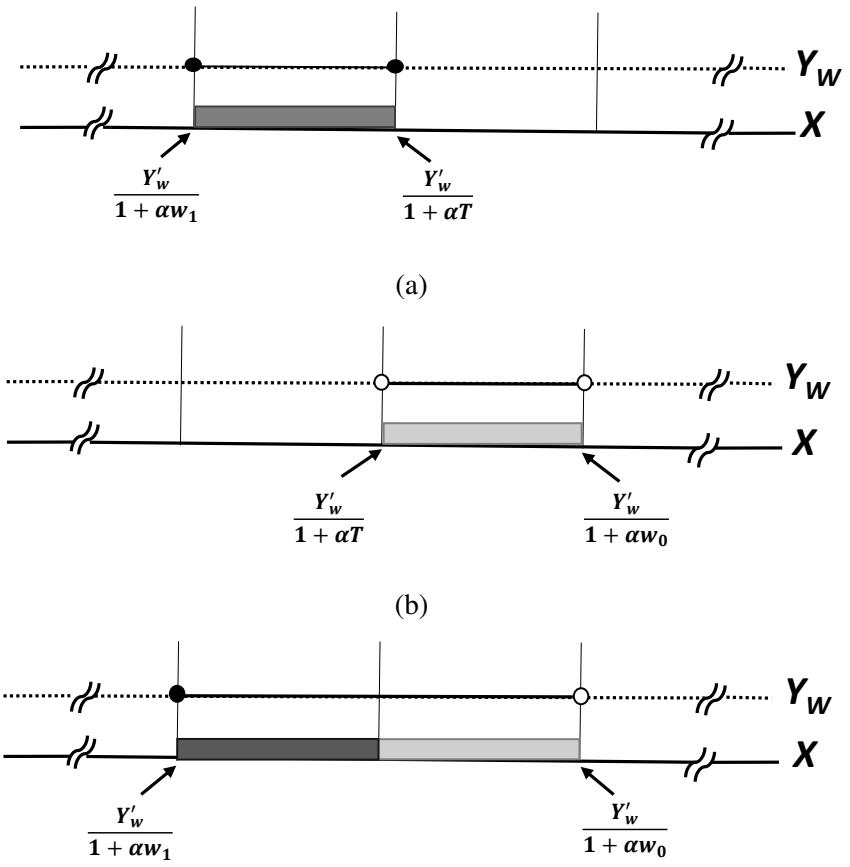


Figure 5.2: The range of the graph wavelet coefficients that is able to extract the secret bits correctly. (a) Hiding $b = 1$. (b) Hiding $b = 0$. (c) Hiding $b = 0$ and 1.

5.2.2.4.2 Blind model

We propose a new model to identify the graph wavelet coefficients that are able to extract the secret bits accurately after the attack in the GWT domain for a blind approach using a prediction-based graph data hiding. The modified coefficients are given as:

$$\mathbf{Y}_w(\ell) = \left\lfloor \frac{\mathbf{Y}(\ell-1) + \mathbf{Y}(\ell+1)}{2} \right\rfloor + w_b. \tag{5.38}$$

After the attack, to extract the secret bits w', we have new values of graph wavelet coefficients values $\mathbf{Y}'_w(\ell'-1)$, $\mathbf{Y}'_w(\ell')$ and $\mathbf{Y}'_w(\ell'+1)$:

$$w'_b = \mathbf{Y}'_w(\ell') - \left\lfloor \frac{\mathbf{Y}'_w(\ell'-1) + \mathbf{Y}'_w(\ell'+1)}{2} \right\rfloor. \tag{5.39}$$

We consider three embedding cases: hiding only $b = 0$ bit, hiding only $b = 1$ bit and hiding $b = 0, 1$ bit.

Proposition 5.7

For embedding a secret bit b=1 and to extract the correct bit after the attack, the modified coefficients should be in the range:

$$\left\lfloor \frac{Y'_w(\ell'-1) + Y'_w(\ell'+1)}{2} \right\rfloor + T \leq Y'_w(\ell') < \left\lfloor \frac{Y'_w(\ell'-1) + Y'_w(\ell'+1)}{2} \right\rfloor + w_1. \quad (5.40)$$

This proposition was proved in Chapter 3.

Proposition 5.8

For embedding b=0 bit, we can extract the correct bits when the modified graph wavelet coefficients are in the range:

$$\left\lfloor \frac{Y'_w(\ell'-1) + Y'_w(\ell'+1)}{2} \right\rfloor + w_0 \leq Y'_w(\ell') < \left\lfloor \frac{Y'_w(\ell'-1) + Y'_w(\ell'+1)}{2} \right\rfloor + T. \quad (5.41)$$

We proved this proposition in Chapter 3.

We combine proposition 5.7 and proposition 5.8 to identify the condition of correct extraction of the secret bits when hiding $b = 0$ and $b = 1$. The range of the graph wavelet coefficients that is able to extract the secret bits correctly is:

$$\left\lfloor \frac{Y'_w(\ell'-1) + Y'_w(\ell'+1)}{2} \right\rfloor + w_0 \leq Y'_w(\ell') < \left\lfloor \frac{Y'_w(\ell'-1) + Y'_w(\ell'+1)}{2} \right\rfloor + w_1. \quad (5.42)$$

Figure 5.3 displays the range of the graph wavelet coefficients capable of retaining the secret bits after the attacks.

5.2.2.5 Joint robust-low distortion data hiding

The proposed models are combined for satisfying the main requirements of the graph data hiding. We consider two embedding algorithms: non-blind and blind. For the non-blind algorithm, in order to combine the model of the embedding distortion minimisation with the robustness

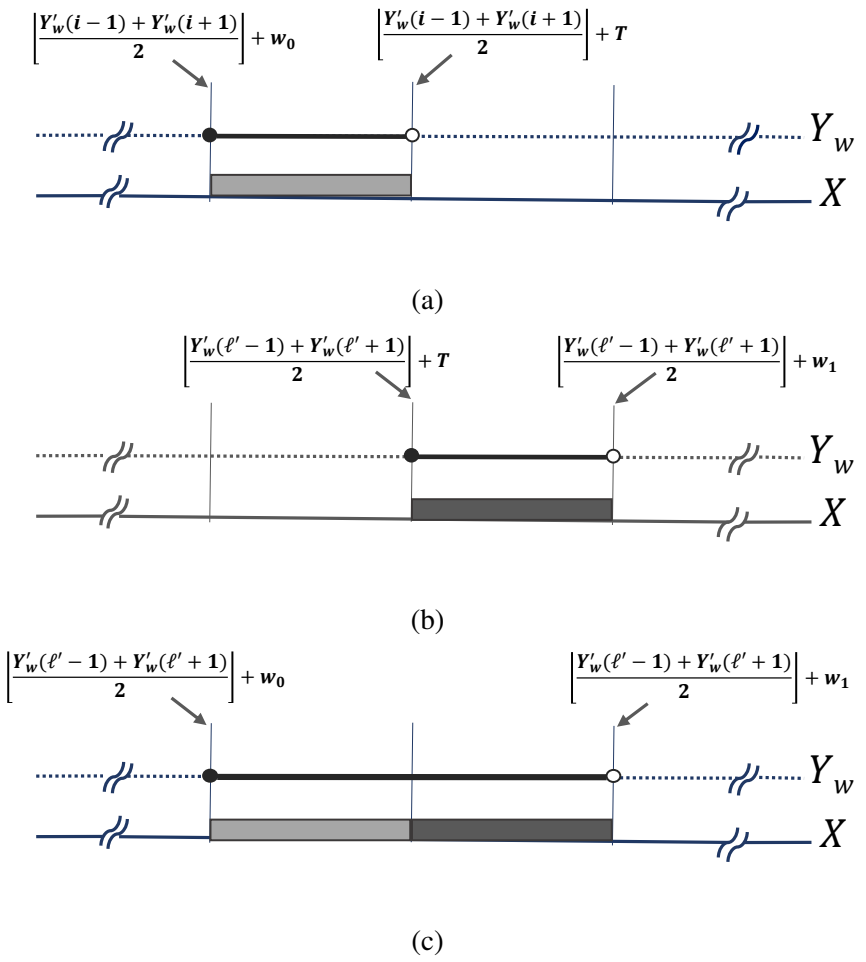


Figure 5.3: The graph wavelet coefficients range which is able to extract the secret bits correctly. (a) Hiding only $b = 0$. (b) Hiding only $b = 1$. (c) Hiding $b = 0$ and $b = 1$.

model, we select the GWT coefficients that satisfy the conditions in Eq. (5.33), Eq. (5.34), Eq. (5.35) for satisfying the robustness conditions. Then, we select the GWT coefficients (from the above GWT coefficients) that satisfy the condition in Eq. (5.11) for for orthogonal graph wavelet filters and Eq. (5.20) for for non-orthogonal graph wavelet filters (in other words, the GWT coefficients have the the lowest values) to minimise the embedding distortion i.e., the GWT coefficients which satisfy the above conditions are selected to embed the secret data. For the blind algorithm, we combine the proposed models of the embedding distortion minimisation and the robustness based on selecting the GWT coefficients that satisfy the conditions in Eq. (5.40), Eq. (5.41), Eq. (5.42) to meet the robustness, then, we select the GWT coefficients which satisfy the condition in Eq. (5.13) for for orthogonal graph wavelet filters and Eq. (5.21) for for non-orthogonal graph wavelet filters (in our model the GWT coefficient triple which has

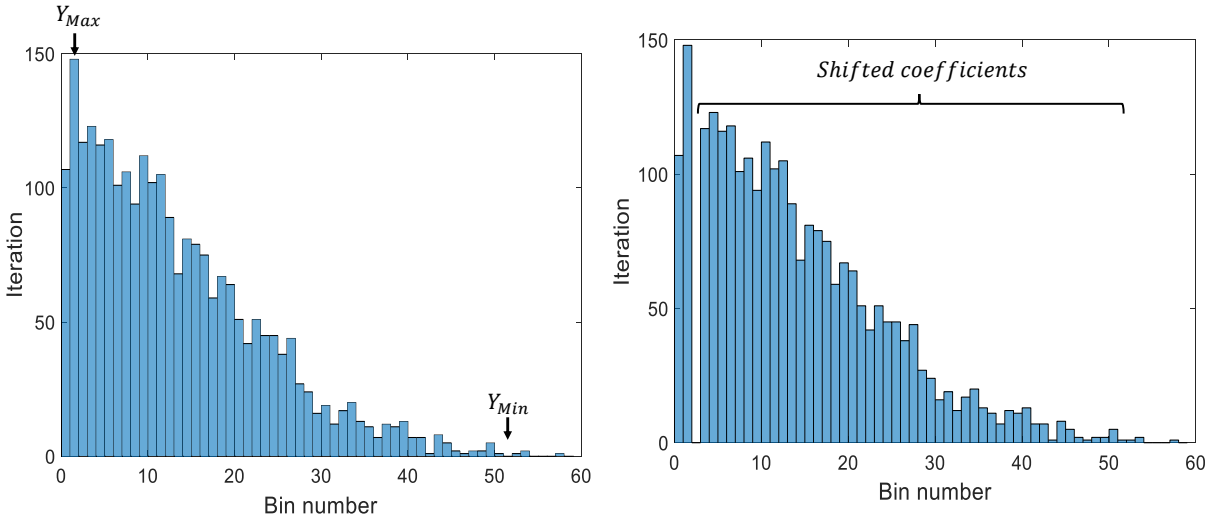
the gradient difference close to 0) in order to minimise the embedding distortion i.e., the GWT coefficients that satisfy the above conditions are chosen for embedding the secret bits.

5.2.3 Proposed methodology of reversible data hiding

5.2.3.1 Reversible data hiding algorithm

We propose a reversible data hiding algorithm based on shifting the graph wavelet coefficients. The algorithm starts with the GWT to decompose the graph signal for n levels of decompositions to obtain the GWT coefficients using Eq. (2.26). Then, the histogram of the magnitudes of the GWT coefficients is generated. The peak point $h(Y_{Max})$ and the zero point $(h(Y_{Min})$ are determined in the histogram, where Y_{Max} and Y_{Min} are the GWT coefficients magnitudes that have the largest and lowest repetition, respectively. Next, all the magnitudes of the GWT coefficients in the range $[Y_{Max}+q, Y_{Min}-q]$ are shifted towards the zero points based on the value of the shifting bin q as illustrated in Figure 5.4. Then, the secret data are hidden in the peak points of the histograms of the low-frequency and high-frequency of the GWT coefficients' magnitudes Y_{Max}. All the GWT coefficients' magnitudes that are greater than Y_{Min} and less than Y_{Max} remain without any change. Finally, the IGWT is applied on the GWT coefficients to obtain the modified graph.

In the extraction process, the receiver extracts the secret bits based on the embedding key. The embedding key includes a number of shifting times, peak points, X_{Max}, zero points, X_{Min}, the shifting bin, q, the level of decomposition, sub-band which is used for embedding, w_0, w_1 and length of the secret bits. The extraction process starts with calculating the GWT coefficients using Eq. (2.26), then the embedded bits are extracted from the magnitudes of the low-frequency and high-frequency of the GWT coefficients in the range $[Y_{Max}, Y_{Max}+q]$. Let's consider for example $w_b(b=\{0,1\})$ and $q=1$, the extracted bit is 0 when the GWT coefficient's magnitude is $\mathbf{Y}_{Max}$, whereas, the extracted bit is 1 when the GWT coefficient's magnitude is $\mathbf{Y}_{Max}+q$. To recover the original coefficients magnitudes perfectly, all embedded data are subtracted from the modified coefficients magnitudes in the range $[\mathbf{Y}_{Max}, \mathbf{Y}_{Max}+q]$. Then, all the magnitudes of the coefficients in the range $[\mathbf{Y}_{Max}+2q, \mathbf{Y}_{Min}]$ are shifted back by $q=1$ unit. Finally, the IGWT is applied on the restored GWT coefficients to get the recovered graph signal. The embedding



(a) (b)

Figure 5.4: Histogram of the high-frequency GWT coefficients of Torus graph with 10000 nodes after two levels wavelet decompositions. (a) Before the shifting process. (b) After the shifting process.

and extracting algorithms are the same embedding and extracting algorithms in chapter 4.

The embedding capacity depends on many parameters, the value of the shifting bin q, the graph connectivity, the level of decomposition and the number of peak points in the low-frequency and high-frequency GWT coefficients which are used in the embedding process. For example, when the shifting bin value is large, it causes the peak point to be high. In other words, it increases the number of the GWT coefficient $\mathbf{Y}_{Max}$. The graph connectivity has an effect on the embedding capacity, for instance some graphs have high peak points which leads to increase the embedding capacity. Moreover, we can increase the number of the GWT coefficients in each sub-band by decreasing the number of graph decomposition levels that leads to increase the embedding capacity. In addition, the embedding capacity can be increased by using several peak points for embedding the data. This case is referred to as a multiple embedding.

In the reversible data hiding, the embedding distortion depends on many parameters, the number of the secret data to be hidden, the decomposition level of GWT coefficients and the sub-band which is used to embed. This is called the embedding distortion due to embedded secret data. Another type of the distortion depends on the type of the process which is used to restore the original host data. This type of distortion is called the reversibility distortion. In the proposed method, the reversibility distortion comes from the shifting of the GWT coeffi-

cients. This distortion depends on the number of the GWT coefficients which are shifted and the number of the shifting times.

To satisfy the main requirements of the reversible data hiding, we have to balance between the distortion and the robustness of the method based on selecting the GWT coefficients. The embedding distortion of the proposed method is decreased when the secret bits are hidden in the high-frequency GWT coefficients. We can improve the robustness of the proposed method by embedding the secret bits in the low-frequency GWT coefficients and using multi levels of decomposition.

5.2.3.2 Embedding distortion minimisation

For minimising the embedding distortion in the proposed method, we established the relationship between the error distortion using mean square error (μ) of the modified graph and the value of the embedding bit $\mathbf{w}$ in graph wavelet domain. The proposed proposition is the MSE (μ) is less than the squared value of embedded bits $\mathbf{w}^2$ as follows:

$$\mu < \mathbf{w}^2 \tag{5.43}$$

For minimising the embedding distortion, we have to minimise the value of $\mathbf{w}$ and this leads to decreasing the MSE value and decreasing the embedding capacity. While increasing the value of $\mathbf{w}$ will increase the MSE value and increase the embedding capacity. Therefore, we have to do a balance between the embedding distortion and the embedding capacity based on choosing a suitable value of $\mathbf{w}$. This proposition has proved in Chapter 4.

5.2.3.3 On enhancing robustness

For improving the robustness of the proposed method, we propose a robustness model to identify the GWT coefficients magnitudes that are able to extract the secret bits after the attack in graph wavelet domain. We establish the relationship between the GWT coefficients magnitudes and the type of attack, namely, additive noise.

We consider three cases of the secret bits: hiding only $b = 0$ bits, hiding only $b = 1$ bits and hiding $b = 0, 1$ bits, where $w_0 < w_1$, $w_0 = 0$, $w_1 = q$ and $q = 1$.

Proposition 5.9

To obtain the correct secret bit after embedding $b = 1$ bits, the modified coefficients have to be in the range:

$$\mathbf{Y}'_{Max}(\ell') + q \leq Y'_w(\ell') < \mathbf{Y}'_{Max}(\ell') + 2q. \tag{5.44}$$

where $w_b = q$ and $q = 1$. This proposition has been proved in Chapter 4.

Proposition 5.10

To embed $b = 0$ bits, we can extract the correct secret bits when the modified coefficients magnitudes are in the range:

$$\mathbf{Y}'_{Max}(\ell') + w_0 \leq Y'_w(\ell') < \mathbf{Y}'_{Max}(\ell') + q. \tag{5.45}$$

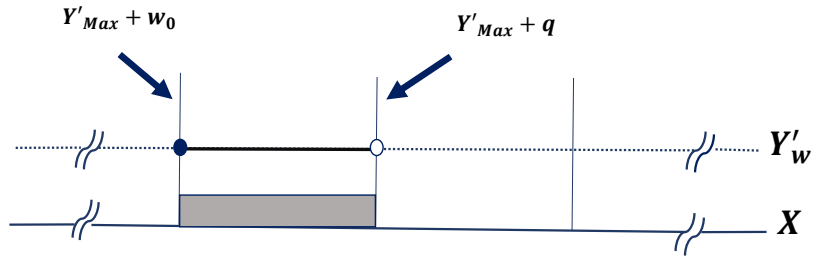
where $w_0 = 0$ and $q = 1$. We have proved this proposition in chapter 4.

Proposition 5.11

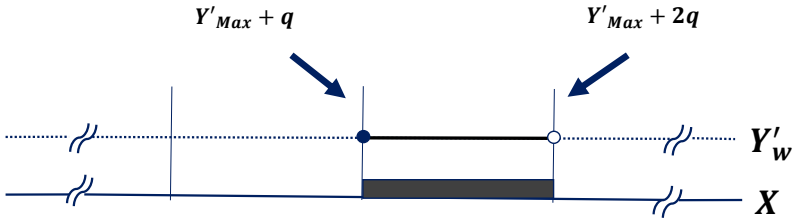
We combine the above propositions to identify the condition of correct detection of the secret bits when hiding $b = 0$ and $b = 1$. The range of the GWT coefficients which retain the secret bits correctly is:

$$\mathbf{Y}'_{Max}(\ell') + w_0 \leq Y'_w(\ell') < \mathbf{Y}'_{Max}(\ell') + 2q. \tag{5.46}$$

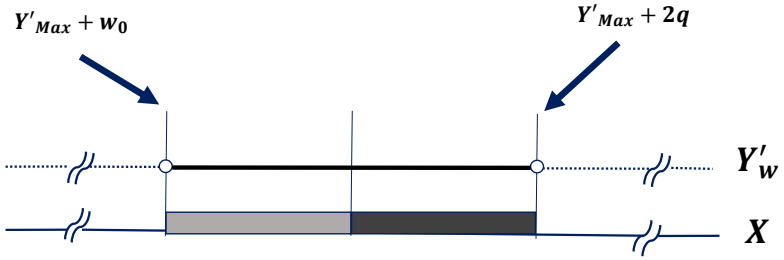
where $w_0 = 0$ and $q = 1$. Figure 5.5 demonstrates the range of the graph wavelet coefficients which able to retain the secret bits after the attacks.



(a)



(b)



(c)

Figure 5.5: The range of the graph wavelet coefficients that able to extract the secret bits correctly. (a) Hiding only $b = 0$. (b) Hiding only $b = 1$. (c) Hiding $b = 0$ and $b = 1$.

5.2.3.4 Joint robust-low distortion reversible data hiding

For satisfying the basic requirements of the graph data hiding, the proposed embedding distortion minimisation model and robustness model are combined. We select the GWT coefficients that satisfy the condition in Eq. (5.46) to satisfy the robustness condition. Then, we select a small value for w (according to the Eq. (5.43) for minimising the embedding distortion) to em-

bed in the selected GWT coefficients for minimising the embedding distortion and enhance the robustness.

5.3 Performance evaluation

The performance of the proposed methods are evaluated using the graph watermarking dataset [181]. We have two parts of the evaluations: performance of irreversible data hiding and performance of reversible data hiding. We would like to indicate that we have identified many limitations regarding the comparison of the proposed irreversible data hiding methods with previous methods. We could not find a method where a framework similar to the proposed method was considered. None of the existing graph data hiding methods embed the secret bits into the graph signal; instead, they embed the secret data in the mesh coordinates or the graph topology. Due to lack of any other comparable work, it is not possible to compare our experimental results with other works. In that context, we consider the results without the proposed models as the baseline. Therefore, we calculate the results by using the data hiding algorithms without using the proposed models (embedding distortion minimisation and robustness) to show improvements when the proposed models were applied using the same data hiding algorithms. For reversible data hiding, we have selected the same RDH methods which are chosen in chapter 4 for the comparison (for the same reasons as shown in chapter 4).

5.3.1 Experimental set up

The proposed GWT data hiding algorithms with the proposed models: embedding distortion minimisation and robustness are tested using the dataset of graph watermarking [181]. This dataset includes 160 various types of graphs with a different number of nodes and five graph signals as described in Section 3.3.1.

5.3.2 Performance evaluation of the irreversible data hiding using GWT

The proposed models: embedding distortion minimisation model and robustness model are verified by the experimental simulations for the non-blind and blind data hiding. The experimental simulations are divided into two types: evaluation of the embedding distortion performance and

evaluation of the performance of the robustness. The proposed models are evaluated based on comparing the performance of the data hiding methods with and without using the proposed models.

5.3.2.1 Evaluation of the embedding distortion Performance

In this section, we present two types of empirical results: verification of the embedding distortion minimisation model and evaluation of the performance of the embedding distortion. We consider two data hiding scenarios: non-blind and blind.

5.3.2.1.1 Verification of embedding distortion minimisation model for non-blind data hiding

The propositions 5.1 and 5.3 are verified in the experimental simulations. We consider two types of graph wavelet filters: orthogonal Meyer filter and bi-orthogonal $9/7$ filter. For the orthogonal Meyer filter, the energy sum of the chosen GWT coefficients and the MSE (μ) of the modified graphs are calculated for the test graphs. For the bi-orthogonal 9/7 filter, the weighted energy sum of the selected GWT coefficients and the MSE (μ) of the modified graphs are calculated for the test graphs. The graph wavelet filter decomposes the graph signal into n levels of decompositions, at each level there is two sub-bands, Low-frequency L and high-frequency H, after two levels of decompositions, four sub-bands are generated, $L1$ and $H1$ at the first level, $L2$ and $H2$ at the second level. The low-frequency GWT coefficients for second level decomposition $L2$ are considered in these experiments. Four sets of graphs with a number of nodes $N = \{500, 2500, 5000, 10000\}$ for 7 graphs types with 5 kinds of graph signals are used to verify the effects of embedding three scenarios of the secret bits. We consider the pseudo-random binary sequences as the secret bits with three scenarios: $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$ to embed in the low frequency GWT coefficients. The low-frequency GWT coefficients are categorised into five groups according to their values. Then, we embed the same number of the secret bits in each group separately using 5 types of graph signals, where blue colour for graph signal 1, red colour for graph signal 2 , magenta colour for graph signal 3 , green colour for graph signal 4 and cyan colour for graph signal 5. We obtain two sets of empirical results to verify the effects of embedding three scenarios of the secret bits as given:

In the Set 1 of experiments, the low frequency of GWT coefficients at the second level of decomposition $L2$ are divided into five groups based on their values by taking into consideration all the low frequency of GWT coefficients using two types of graph wavelet filters, i.e. orthogonal Meyer filter and bi-orthogonal $9/7$ filter. Then, the energy sum (or weighted energy sum according to the type of wavelet filter) of the chosen graph wavelet coefficients to be modified and MSE of the modified graphs are calculated using the same $\alpha = 0.1$ for all groups separately. In these experiments, we consider the case when the secret bits equal to $\mathbf{w} = \{1\}$ as shown in Figure 5.6 and Figure 5.7 respectively.

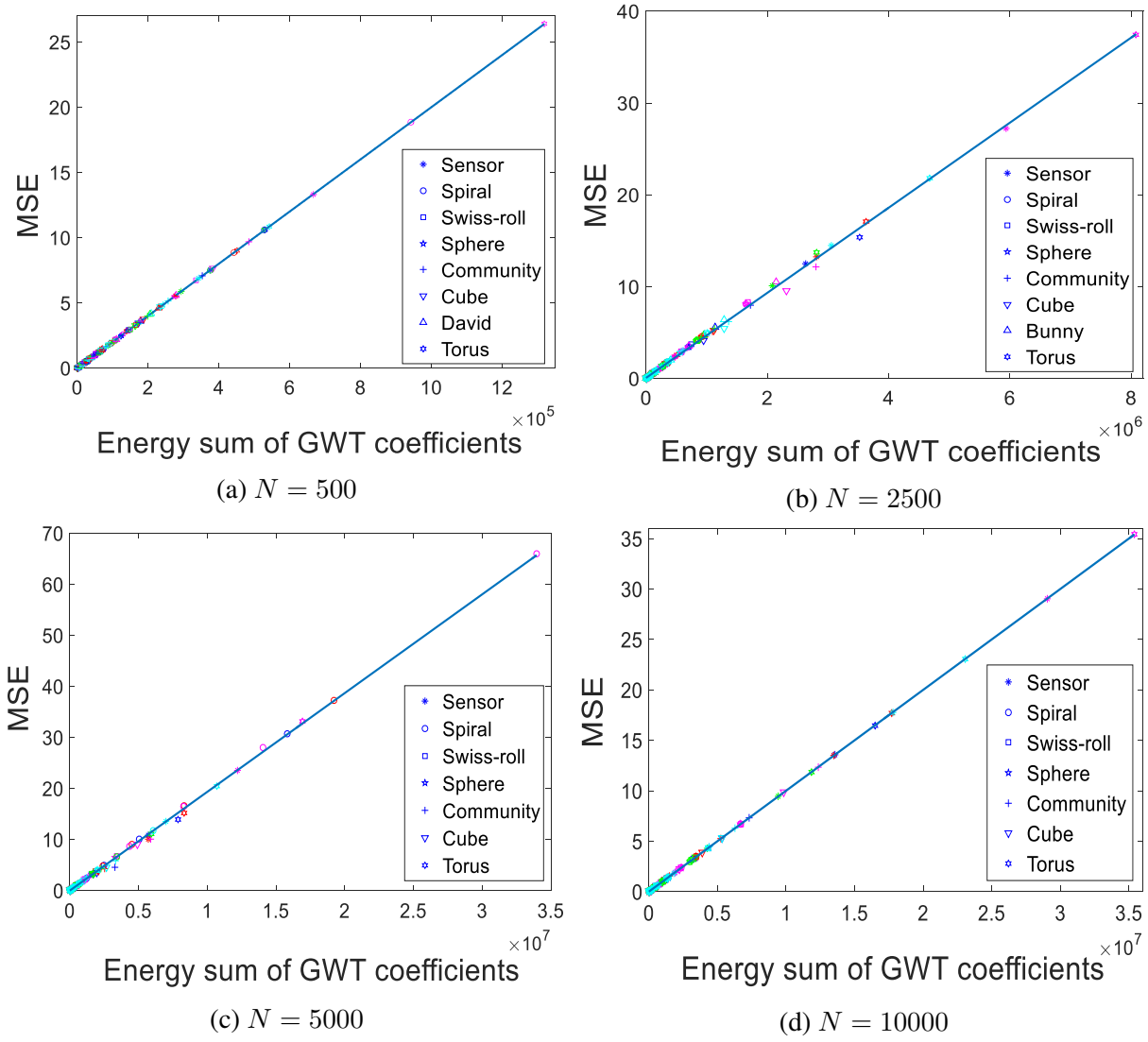
In the experiment Set 2, we consider the case when the secret bits are zero and one $\mathbf{w} = \{0,\ 1\}$, where the number of 0s and 1s are equally distributed in the binary sequence using two types of graph wavelet filters, i.e. orthogonal Meyer filter and bi-orthogonal $9/7$ filter. Embedding performance is calculated in a similar way to that mentioned in Set 1 of experiments to notice the trend as illustrated in Figure 5.8 and Figure 5.9, respectively.

The results demonstrate that the distortion in the experiment Set 1 is higher than the distortion in the experiment Set 2 because the number of 1s which is embedded in the experiment Set 1 is double than the embedded numbers in the experiment Set 2. While there is no distortion in case of embedding $\mathbf{w} = \{0\}$ only.

The simulation results demonstrate a strong correlation between the energy sum of the graph wavelet coefficients which are selected for embedding and the MSE of the modified graph. We can see that the MSE of the modified graph is a linear proportional to the energy sum of the selected GWT coefficients using orthogonal Meyer filter and to the weighted sum of the energy of the chosen graph wavelet coefficients using bi-orthogonal $9/7$ filter (where $y = m_1 x + \beta$, m_1 is the slope of the graph and β is the y-intercept). The proposed model is supported by simulation results using the graph dataset.

5.3.2.1.2 Verification of the embedding distortion minimisation model for blind data hiding

The propositions 5.2 and 5.4 are verified in the simulation results. We consider two types of graph wavelet filters: orthogonal Meyer filter and bi-orthogonal $9/7$ filter. For the orthogonal Meyer filter, the MSE (μ) of the modified graph and the gradient difference of the se-

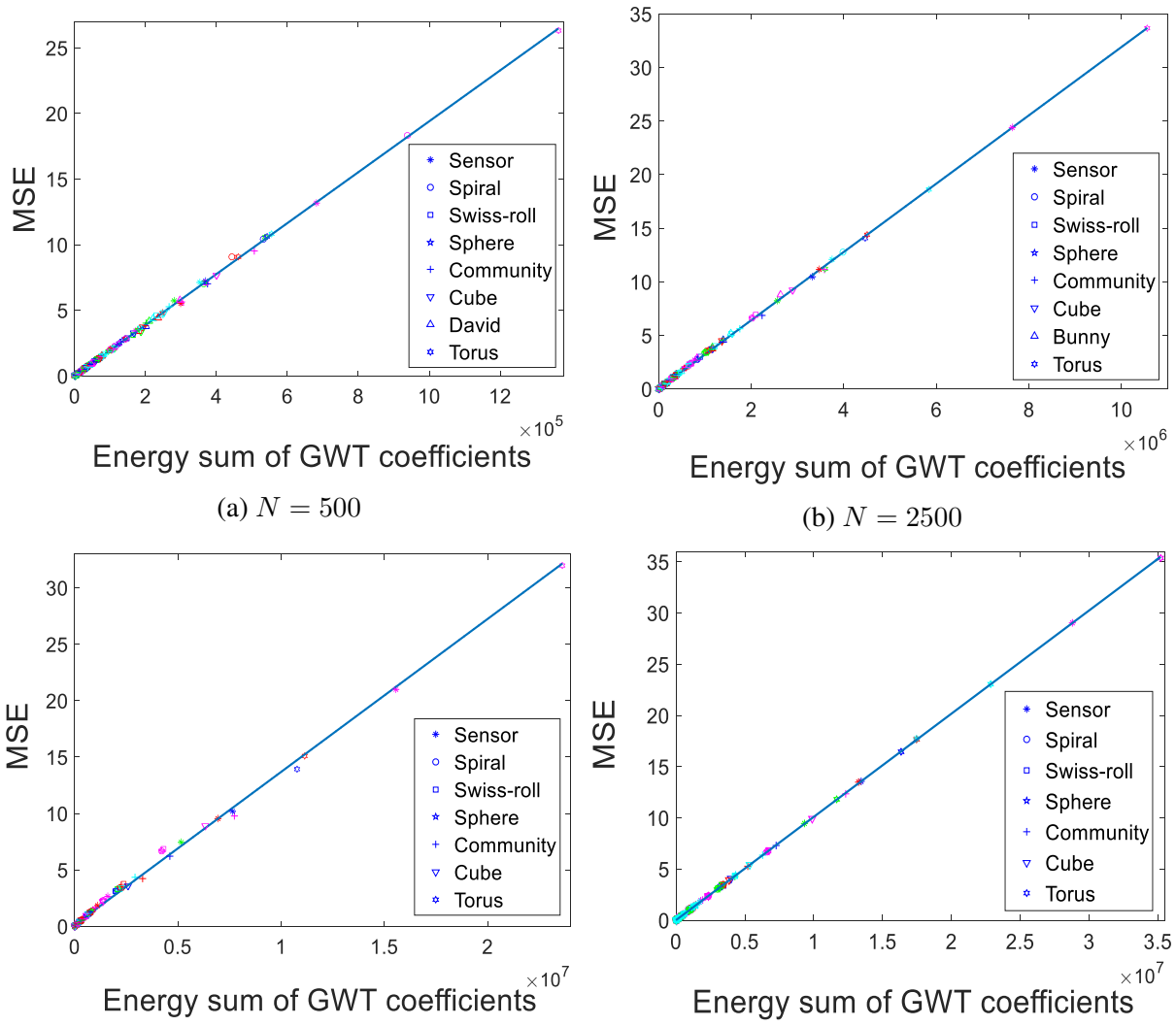


(a) $N = 500$ (b) $N = 2500$

(c) $N = 5000$ (d) $N = 10000$

Figure 5.6: Verification of embedding distortion of non-blind algorithm using orthogonal Meyer filter: MSE of the modified graph vs. energy sum of GWT coefficients when $w = \{1\}$, for individual graphs with different number of nodes $N = 500, 2500, 5000$ and 10000, respectively for 5 graph signals where the colours, blue, red, magenta, green and cyan represent the graph signal $1,\ 2,\ 3,\ 4\ and\ 5$, respectively and the blue line demonstrate the MSE is linearly proportional to the energy sum (where $y = m_1 x + \beta$).

lected GWT coefficients to be modified have been calculated for the test graphs. For the bi-orthogonal 9/7 filter, the MSE (μ) of the modified graph and the weighted gradient difference of the selected GWT coefficients have been calculated. In these experiments, four sets of graphs $N = \{500, 2500, 5000, 10000\}$, respectively for 4 graphs types with 5 kinds of graph signals are used to verify the effects of embedding five scenarios of the secret data. We consider the pseudo-random number sequences as the secret data with five scenarios: $\mathbf{w} = \{0, 0.1, 0.2, 0.3, 0.4\}$ to



(a) $N = 500$

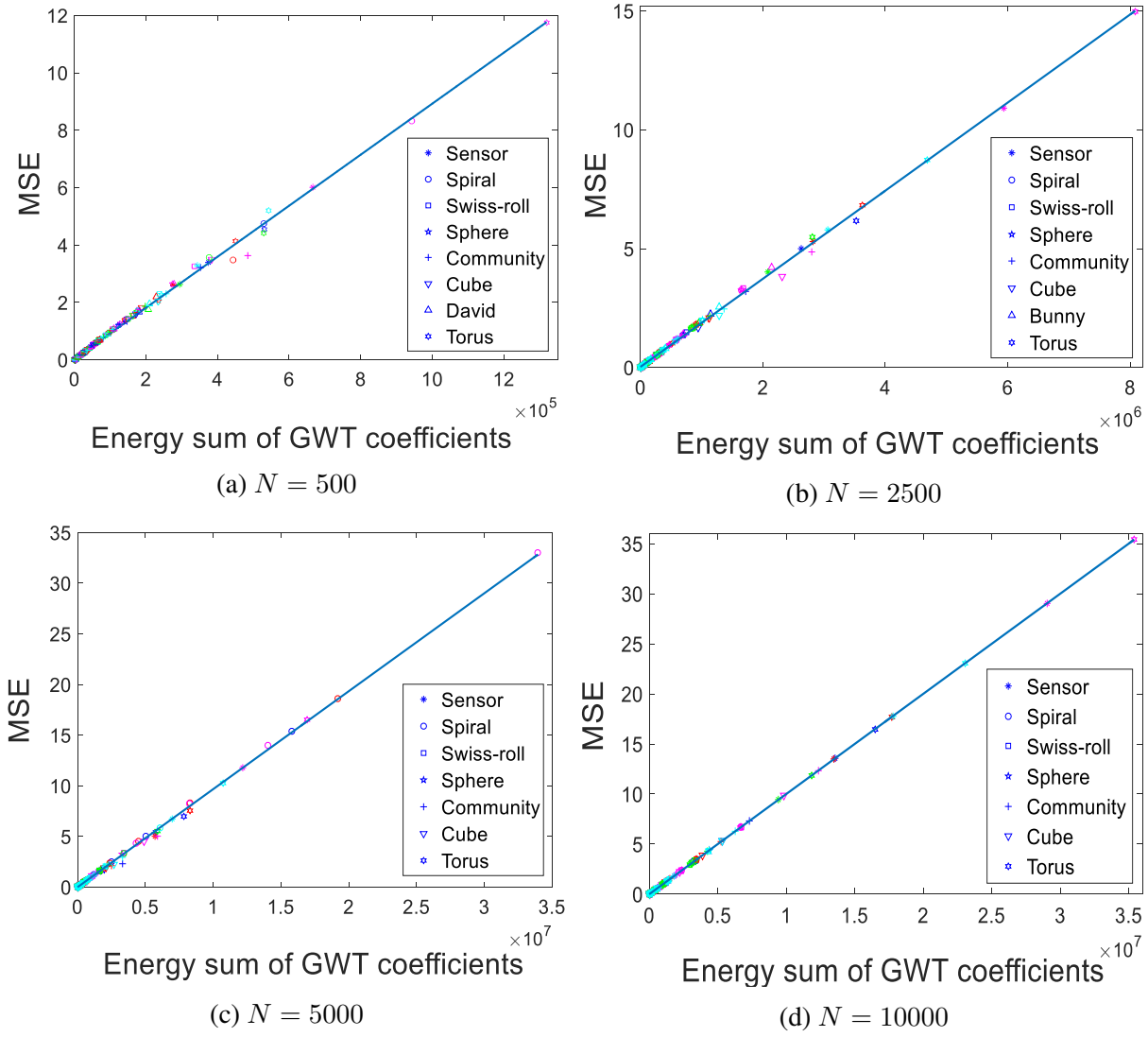
(b) $N = 2500$

(c) $N = 5000$

(d) $N = 10000$

Figure 5.7: Verification of embedding distortion of non-blind algorithm using bi-orthogonal $9/7$ filter: MSE of the modified graph vs. weighted energy sum of GWT coefficients when $\mathbf{w} = \{1\}$, for individual graphs with different number of nodes $N = 500, 2500, 5000$ and 10000, respectively for 5 graph signals where the colours, blue, red, magenta, green and cyan represent the graph signal $1,\ 2,\ 3,\ 4\ and\ 5$, respectively and the blue line demonstrate the MSE is linearly proportional to the energy sum (where $y = m_1 x + \beta$).

embed in the low frequency of GWT coefficients at the second level $L2$ of the graph dataset. The sorted GWT coefficients are divided into four groups based on their gradient difference values by taking into consideration all the low-frequency GWT coefficients $L2$. Then, the MSE of the modified graph has been calculated for all groups separately using different embedding scenarios. We calculate two sets of empirical results to demonstrate the effects of embedding five scenarios of the secret data as given below:



(a) $N = 500$

(b) $N = 2500$

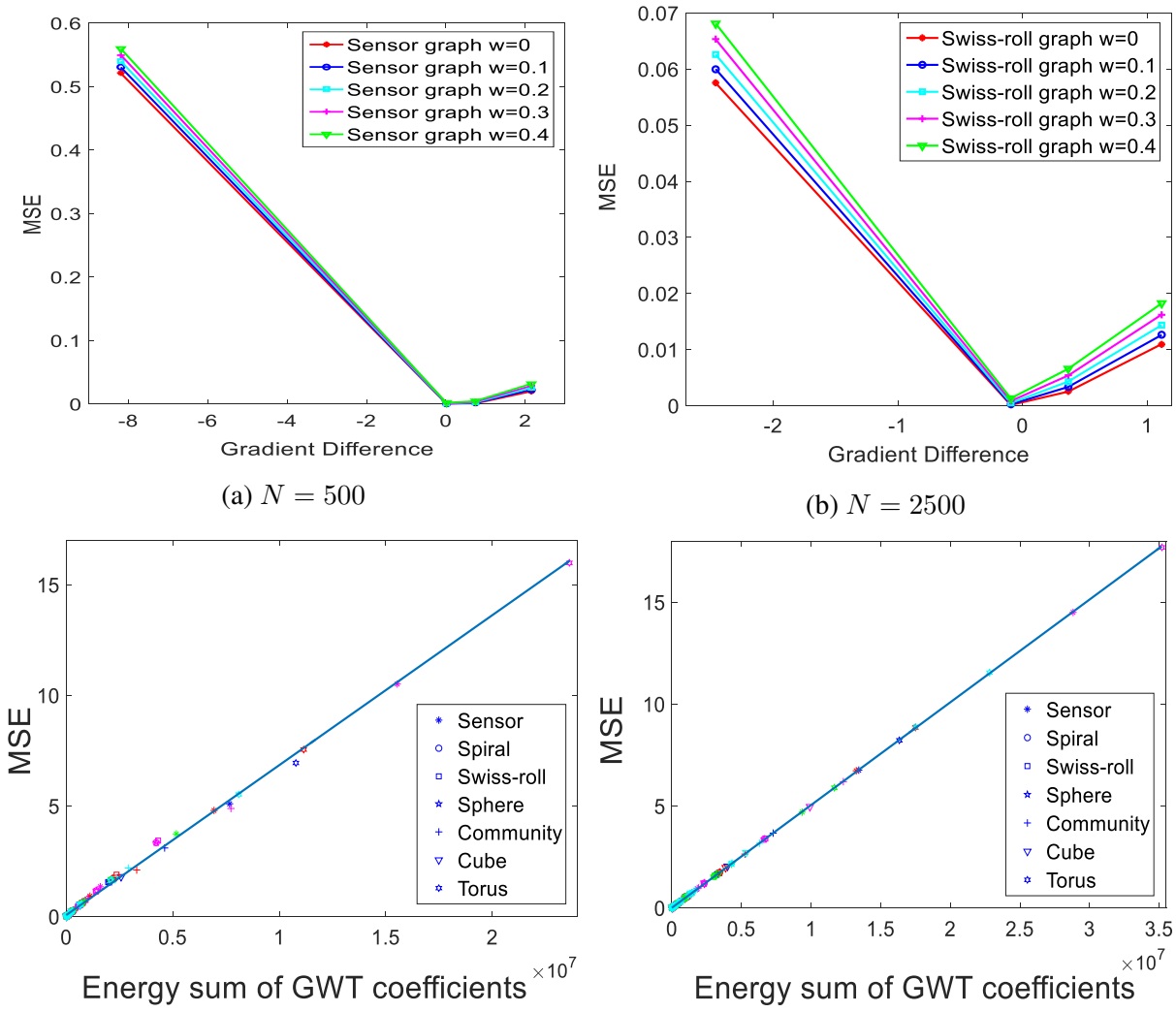
(c) $N = 5000$

(d) $N = 10000$

Figure 5.8: Verification of embedding distortion of non-blind algorithm using orthogonal Meyer filter: MSE of the modified graph vs. energy sum of GWT coefficients when $\mathbf{w} = \{0, 1\}$, for individual graphs with different number of nodes $N = 500, 2500, 5000$ and 10000, respectively for 5 graph signals where the colours, blue, red, magenta, green and cyan represent the graph signal $1,\ 2,\ 3,\ 4\ and\ 5$, respectively and the blue line demonstrate the MSE is linearly proportional to the energy sum (where $y = m_1 x + \beta$).

In the Set 1 of experiments, proposition 5.2 is verified using the orthogonal Meyer filter for four sets of graphs $N = \{500, 2500, 5000, 10000\}$ and for 4 graphs types as shown in Figure 5.10, Figure 5.11, Figure 5.12 and Figure 5.13, respectively.

In experiment Set 2, proposition 5.4 is verified using the bi-orthogonal $9/7$ filter. In these experiments, four sets of graphs $N = \{500, 2500, 5000, 10000\}$ and for 4 graphs types as shown in Figure 5.14, Figure 5.15, Figure 5.16 and Figure 5.17, respectively.



(a) $N = 500$ (b) $N = 2500$

(c) $N = 5000$ (d) $N = 10000$

Figure 5.9: Verification of embedding distortion of non-blind algorithm using bi-orthogonal $9/7$ filter: MSE of the modified graph vs. weighted energy sum of GWT coefficients when $\mathbf{w} = \{0, 1\}$, for individual graphs with different number of nodes $N = 500, 2500, 5000$ and $N = 10000$, respectively for 5 graph signals where the colours, blue, red, magenta, green and cyan represent the graph signal $1,\ 2,\ 3,\ 4\ and\ 5$, respectively and the blue line demonstrate the MSE is linearly proportional to the energy sum (where $y = m_1 x + \beta$).

The empirical simulations demonstrate that the MSE of the modified graph has a strong correlation with the gradient difference of a GWT coefficient triple. It is observed that the minimum distortion is obtained (low MSE) when the gradient difference of GWT coefficient triple is close to zero. The proposed model is supported based on the extensive simulation results using a graph dataset and various embedding scenarios.

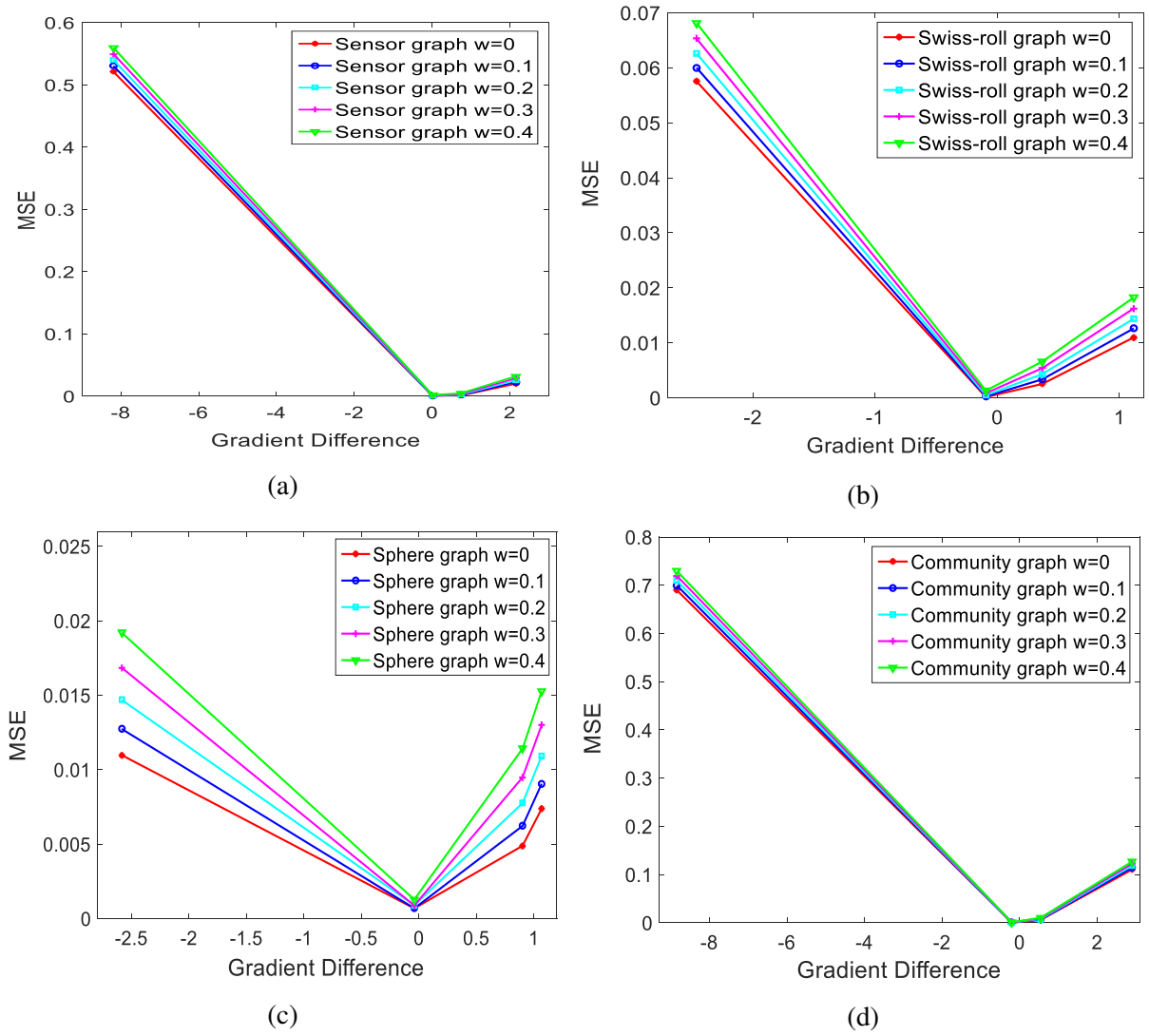


Figure 5.10: Verification of embedding distortion of blind algorithm using orthogonal Meyer filter: MSE of the modified graph vs. gradient difference, for individual graphs with nodes $N = 500$. (a) Sensor graph. (b) Swiss-roll graph. (c) Sphere graph. (d) Community graph.

5.3.2.1.3 Performance evaluation of the embedding distortion of non-blind data hiding

The performance of the embedding distortion of non-blind data hiding is evaluated for various embedding capacities using graph dataset. In these experiments, we consider the low-frequency GWT coefficients at the second level decomposition $L2$ to embed the secret bits $\mathbf{w} = \{0, 1\}$. A set of 35 graphs with $N = 2500$ nodes are used for evaluation of the method performance. We calculate the MSE of the modified graphs using the original algorithm with using the proposed embedding distortion minimisation model (by embedding the secret bits in the GWT coef-

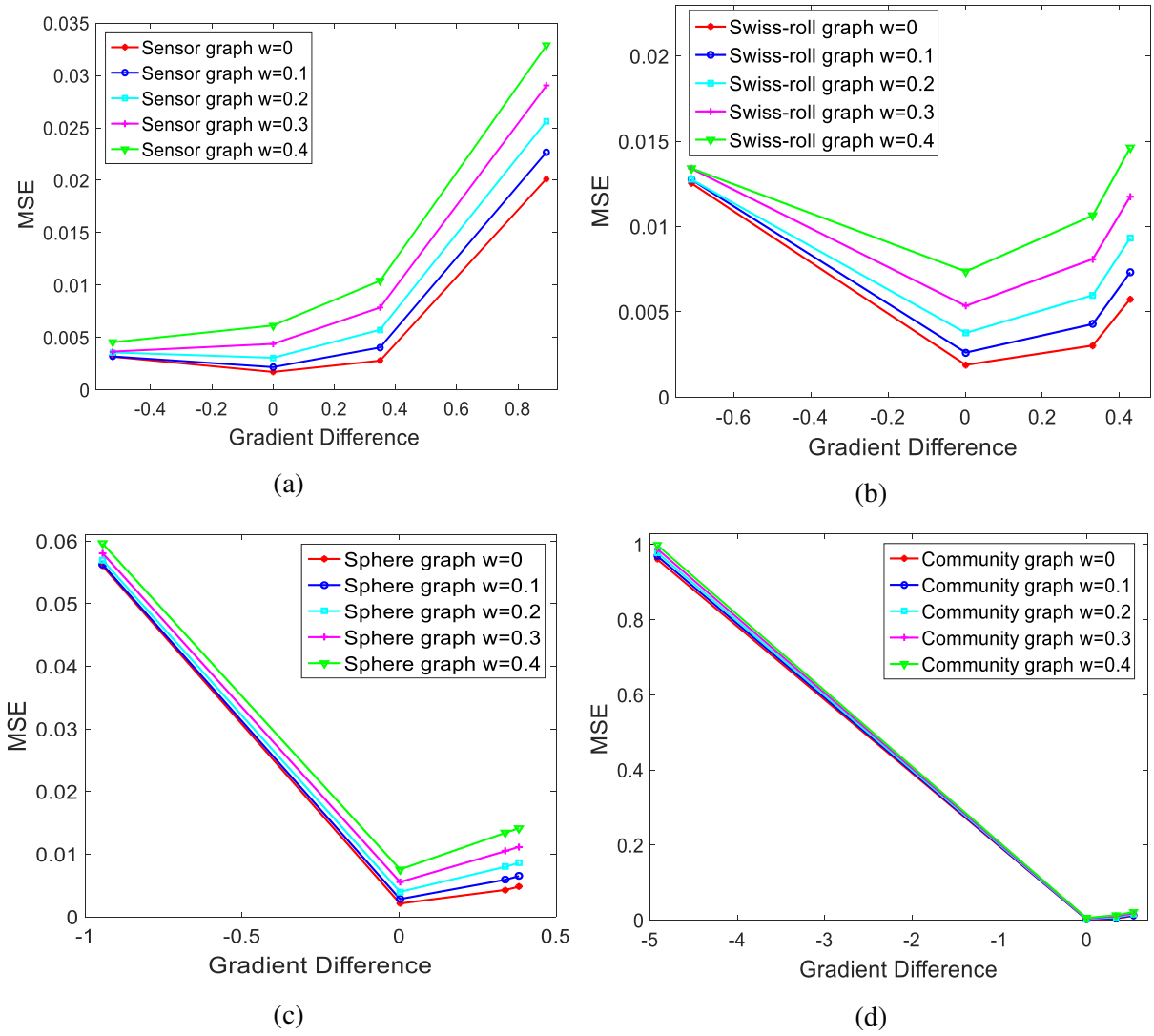


Figure 5.11: Verification of embedding distortion of blind algorithm using orthogonal Meyer filter: MSEof the modified graph vs. gradient difference, for individual graphs with nodes $N = 2500$. (a) Sensor graph. (b) Swiss-roll graph. (c) Sphere graph. (d) Community graph.

ficients which have low values) and MSE of the modified graphs using the original algorithm without using the proposed model (by embedding the secret bits in any GWT coefficients which are selected randomly). We use the same data hiding parameter $\alpha = 0.1$ and bi-orthogonal 9/7 filter. Figure 5.19 shows the community graph with $N = 2500$ nodes before and after embedding the secret bits with length 350 bits.

The empirical results show that the proposed method provides lower distortion over the original algorithm without the model. As shown in Figure 5.18, the embedding distortion is improved by an average of 99% compared to the original algorithm. We can notice that the

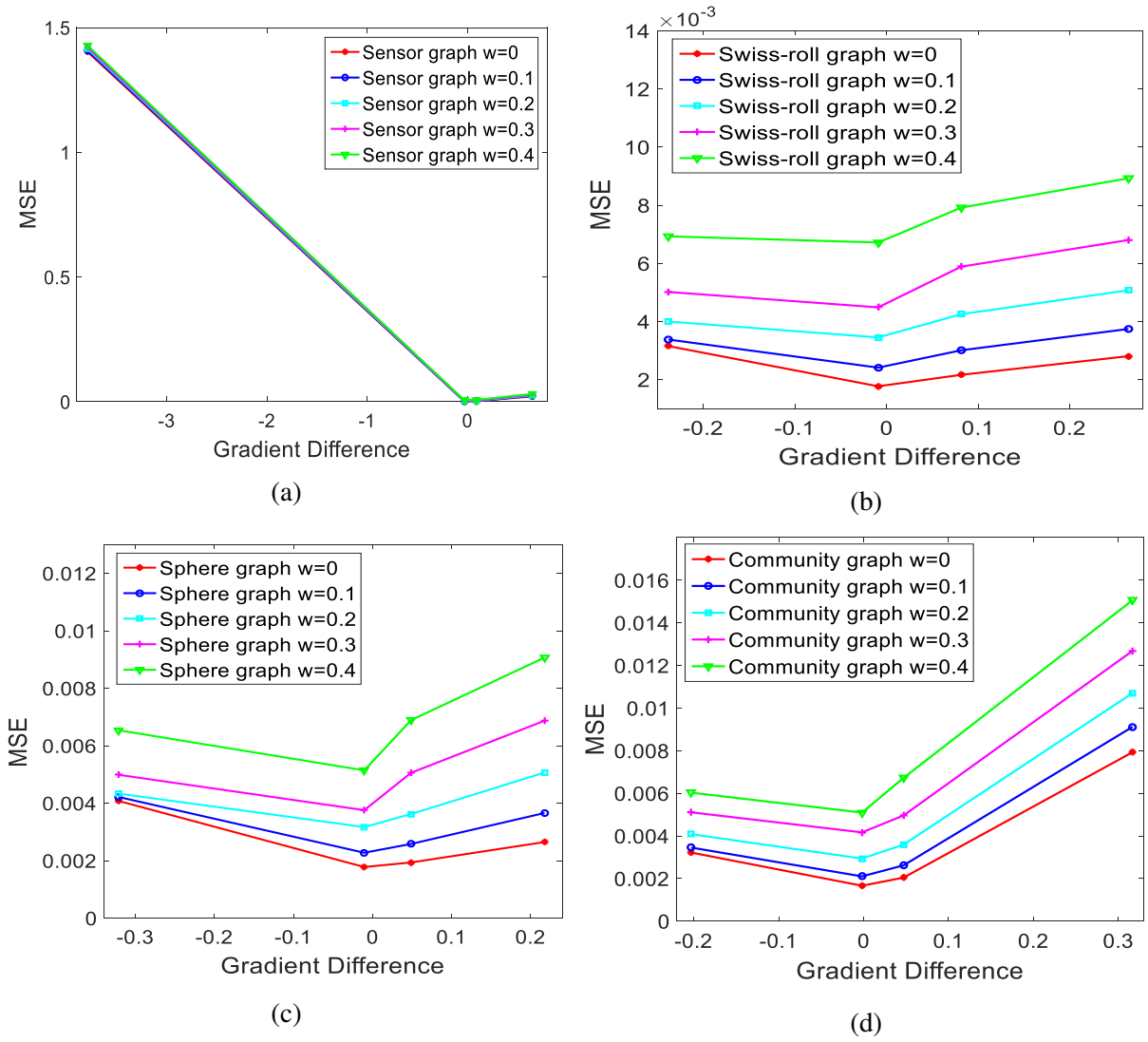


Figure 5.12: Verification of embedding distortion of blind algorithm using orthogonal Meyer filter: MSE of the modified graph vs. gradient difference, for individual graphs with nodes $N = 5000$. (a) Sensor graph. (b) Swiss-roll graph. (c) Sphere graph. (d) Community graph.

embedding distortion using the MSE (μ) is increased when the embedding capacity is increased.

5.3.2.1.4 Performance evaluation of the embedding distortion of blind data hiding

The performance of the embedding distortion model of blind data hiding is evaluated at various embedding capacities using graph dataset. In these experiments, we use two sets of graphs with $N = 2500$ and $N = 5000$ nodes, respectively for 5 types of graph signals for evaluation of the method performance. We consider the low-frequency GWT coefficients at the second

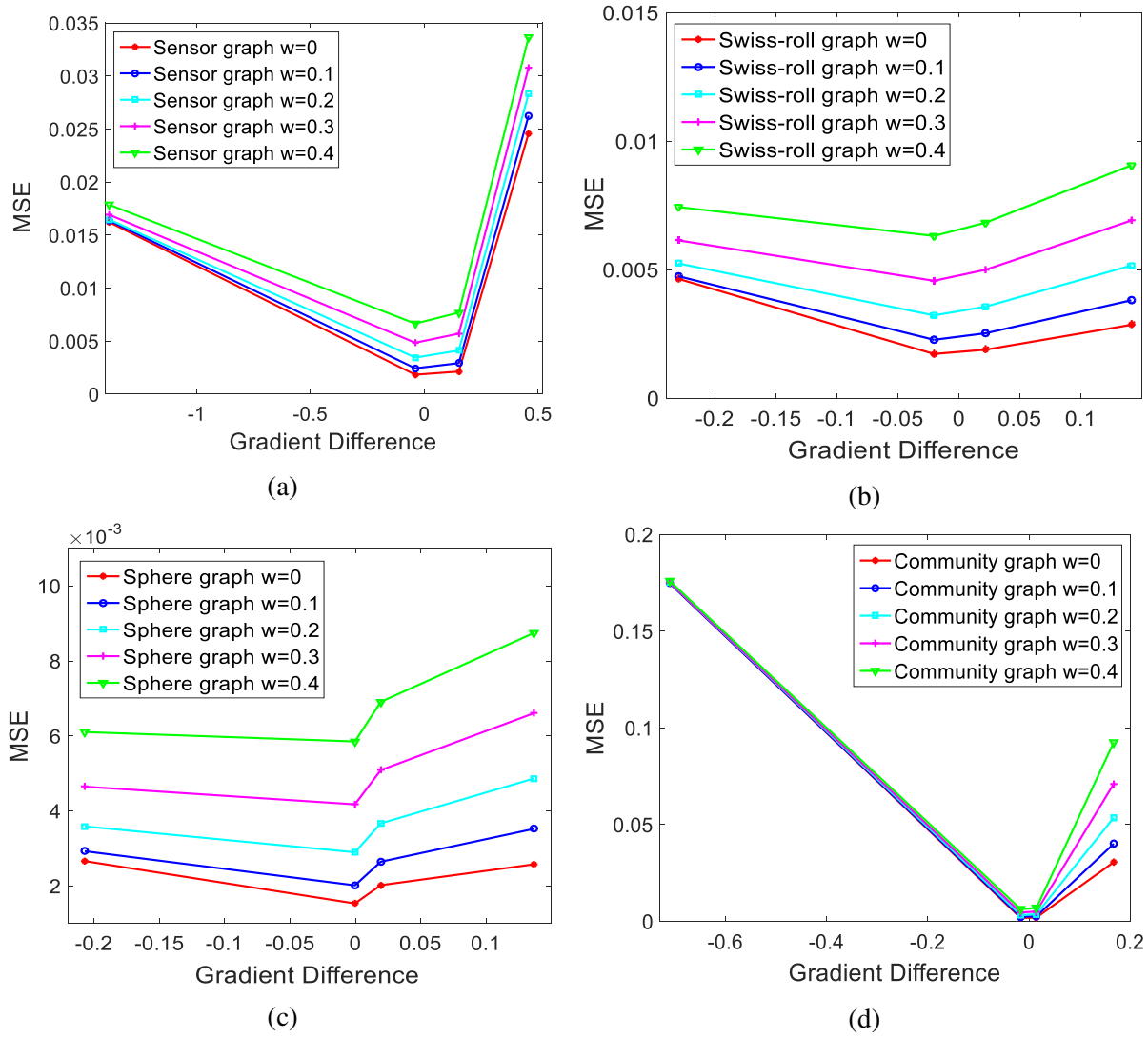


Figure 5.13: Verification of embedding distortion of blind algorithm using orthogonal Meyer filter: MSE of the modified graph vs. gradient difference, for individual graphs with nodes $N = 10000$. (a) Sensor graph. (b) Swiss-roll graph. (c) Sphere graph. (d) Community graph.

level decomposition $L2$ and bi-orthogonal wavelet filter 9/7 to embed the secret data $0.1,\ 0.3$ to represent $0,\ 1$. We calculate the MSE of the modified graphs using the original algorithm with the proposed embedding distortion minimisation model (by embedding the secret data in the GWT coefficients triple which have gradient difference close to 0) and MSE of the modified graphs using the same algorithm without using the proposed model (by embedding the same secret data in the GWT coefficients triple which have any gradient differences). Figure 5.20 shows the spiral graph with $N = 5000$ nodes before and after embedding the secret bits with length 200 bits.

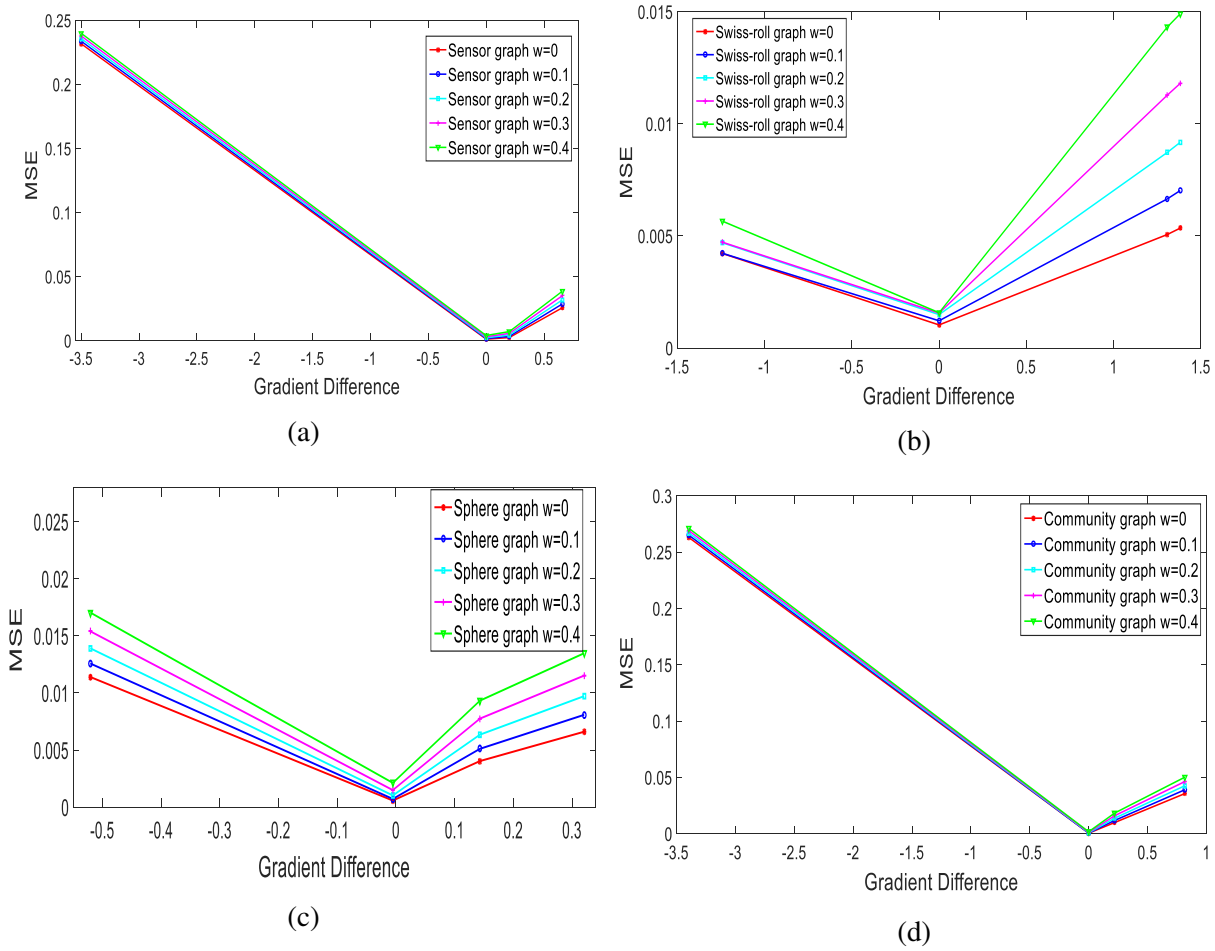


Figure 5.14: Verification of embedding distortion of blind algorithm using bi-orthogonal 9/7: MSE of the modified graph vs. gradient difference, for individual graphs with nodes $N = 500$. (a) Sensor graph. (b) Swiss-roll graph. (c) Sphere graph. (d) Community graph.

The experimental results demonstrate that the proposed method provides lower distortion over the original algorithm without the model. As illustrated in Figure 5.21, the embedding distortion is improved by an average of 99.4% compared to the original algorithm. We can notice that the embedding distortion using the MSE (μ) is increased when the embedding capacity is increased.

5.3.2.2 Evaluation of the performance of the robustness model

In this section, we present the evaluation of the robustness performance of non-blind and blind data hiding.

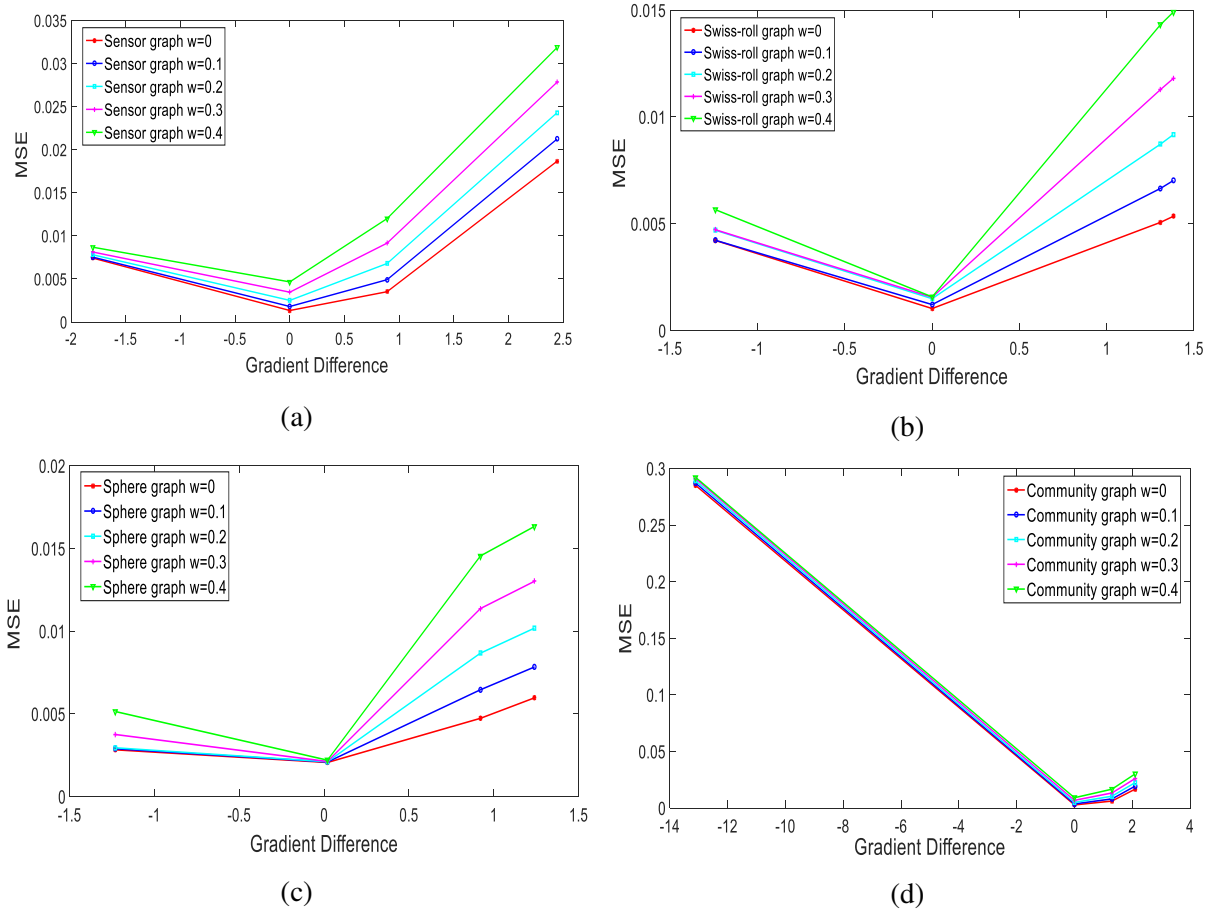


Figure 5.15: Verification of embedding distortion of blind algorithm using bi-orthogonal 9/7: MSE of the modified graph vs. gradient difference, for individual graphs with nodes $N = 2500$. (a) Sensor graph. (b) Swiss-roll graph. (c) Sphere graph. (d) Community graph.

5.3.2.2.1 Performance evaluation of the robustness model of non-blind data hiding

The robustness model of non-blind algorithm is evaluated in the experimental simulations. Two sets of the experiments are obtained for verifying the robustness model.

In the experiments Set 1, we calculate the Hamming Distance (HD) of the extracted secret bits after the additive noise and deleting nodes data. The HD is calculated using the non-blind algorithm with the robustness model (based on selecting the graph wavelet coefficients that satisfy the specific conditions in Eq. (5.33), Eq. (5.34), and Eq. (5.35) to embed the secret bits) and the HD of the extracted bits after the attack using the same algorithm without the robustness model (by embedding the same secret bits in the GWT coefficients randomly). The pseudo-random binary sequences are considered as the secret data to hide in the low frequency

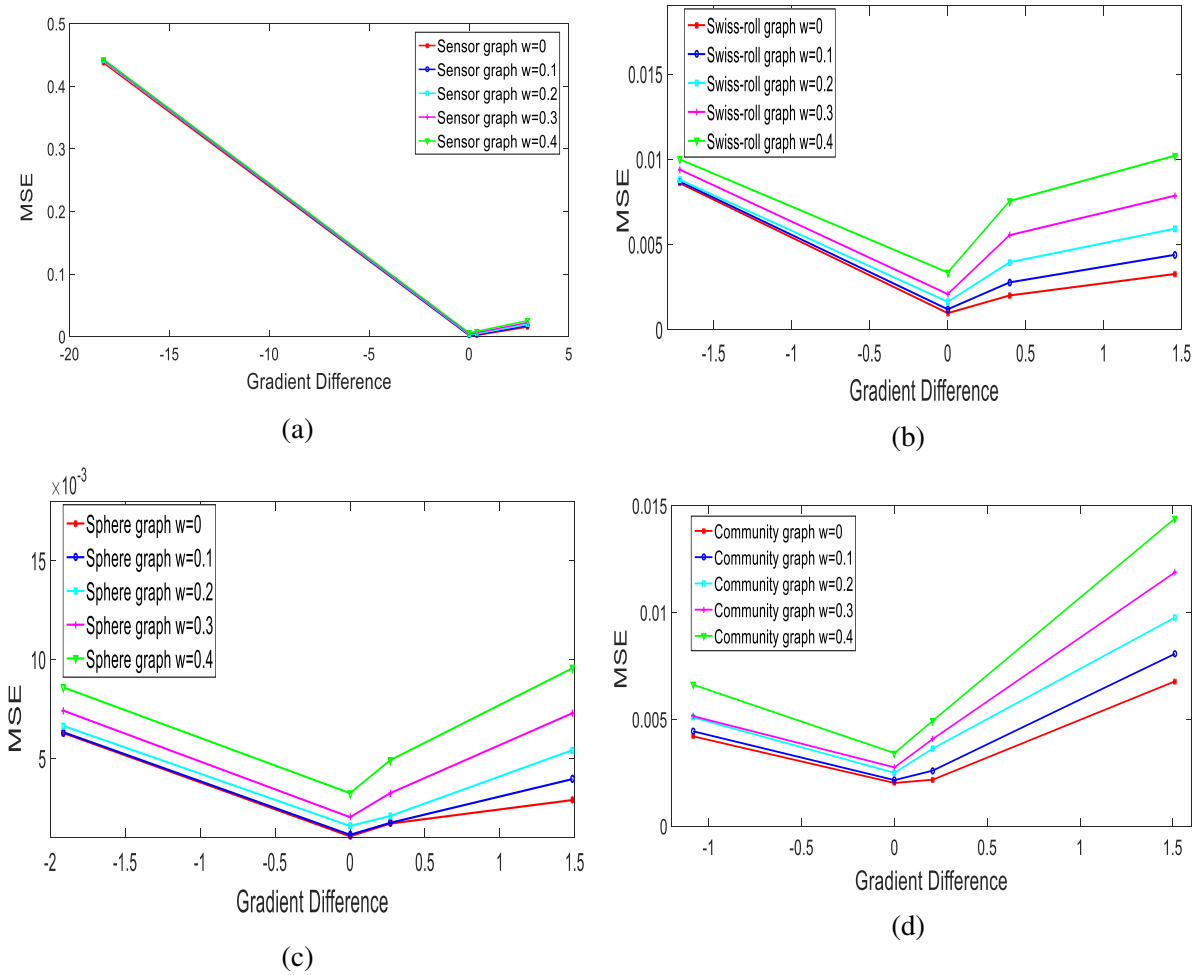


Figure 5.16: Verification of embedding distortion of blind algorithm using bi-orthogonal 9/7: MSE of the modified graph vs. gradient difference, for individual graphs with nodes $N = 5000$. (a) Sensor graph. (b) Swiss-roll graph. (c) Sphere graph. (d) Community graph.

coefficients at the second level of decomposition $L2$ for three hiding scenarios: $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$. In these experiments, a sets of graphs with $N = 500$ nodes for 7 graphs types is used to evaluate the robustness model using bi-orthogonal 9/7 filter and data hiding parameter $\alpha = 0.5$.

The empirical results demonstrate that the proposed method provides higher robustness over the original algorithm without the robustness model. As shown in Figure 5.22, the robustness against the additive noise is improved by an average of 99.9%, 99.9% and 33% for three embedding scenarios, $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$, respectively. Figure 5.23 illustrates that the proposed method outperforms the original algorithm by an average of 60%, 99.9% and 55% for three embedding scenarios, $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$, respectively after

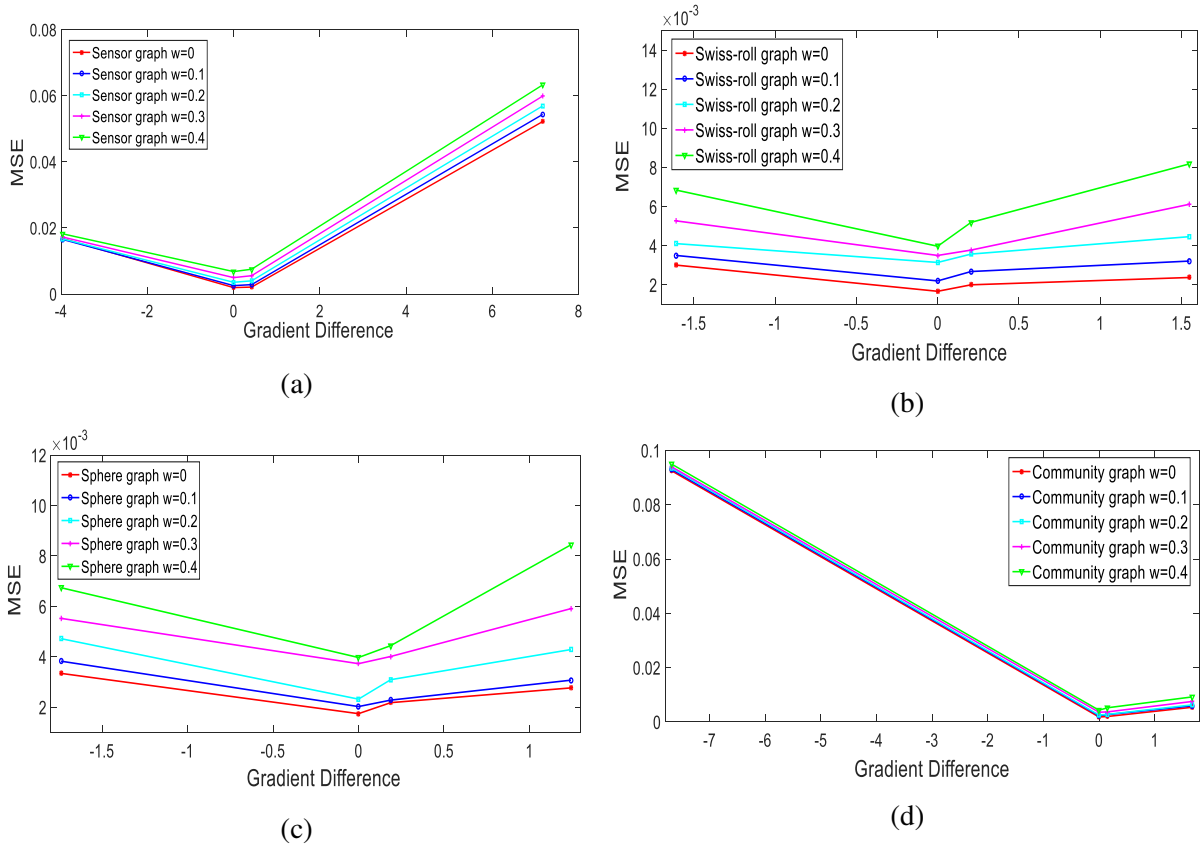


Figure 5.17: Verification of embedding distortion of blind algorithm using bi-orthogonal 9/7 : MSE of the modified graph vs. gradient difference, for individual graphs with nodes $N = 10000$. (a) Sensor graph. (b) Swiss-roll graph. (c) Sphere graph. (d) Community graph.

deleting different number of nodes data randomly.

In the experiments Set 2, we evaluate the robustness model based on calculating the Ham-

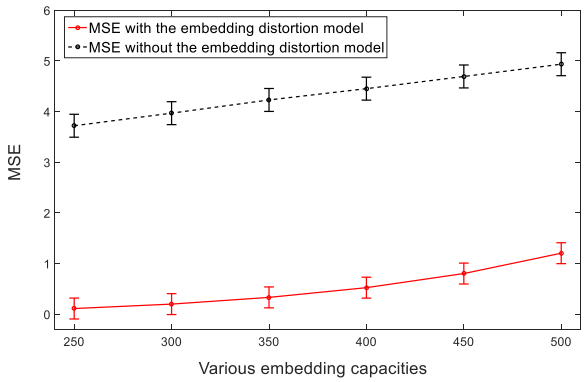


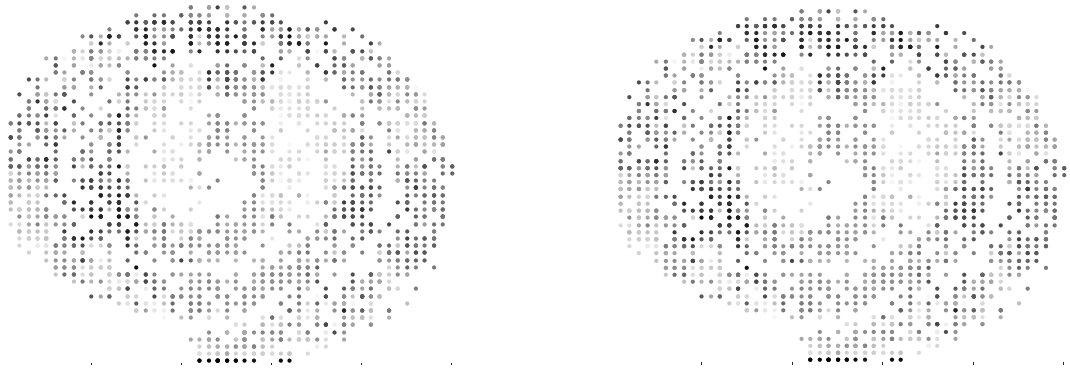
Figure 5.18: Embedding distortion performance of the non-blind algorithm for various embedding capacities using 35 graphs with $N = 2500$ nodes.

(a) (b)

Figure 5.19: Community graph. (a) Original Community graph. (b) Modified Community graph.

ming Distance (HD) of the extracted secret bits after the additive noise and deleting nodes data. In these experiments, we embed the binary logos with (25×25) bits as the secret data $\mathbf{w} = \{0, 1\}$ in the GWT coefficients using graph dataset with $N = 5000$ graph nodes. We consider the low frequency coefficients at the second level of decomposition L2 in the experiments. The HD is calculated using non-blind algorithm with the robustness model (based on embedding the secret bits in the GWT coefficients that satisfy the specific condition in Eq. (5.35)). Also, we calculate the HD using non-blind algorithm without using the proposed model (by embedding the same secret bits in the GWT coefficients which are selected randomly).

The experiments results illustrate that the robustness of the non-blind algorithm is improved by using the robustness model. Figure 5.24 shows the robustness is enhanced against the addi-



(a) (b)

Figure 5.20: Spiral graph. (a) Original Spiral graph. (b) Modified Spiral graph.

tive noise and deleting nodes data by an average of 91% and 99.9% respectively.

5.3.2.2.2 Performance evaluation of the robustness model for blind data hiding

The robustness performance of the blind data hiding is evaluated based on calculating the Hamming distance (HD) of the extracted bits after noise addition and deletion nodes data. The HD of the extracted bits after the attack is computed using the blind algorithm with using the robustness model (based on selecting the graph wavelet coefficients that satisfy the specific conditions in Eq. (5.40), Eq. (5.41), and Eq. (5.42)), and the HD of the extracted bits after the attack is computed using the same blind algorithm without using the robustness model (based on embedding the same secret bits in the GWT randomly). We consider the pseudo-random number sequences as the secret data to hide in the low frequency GWT coefficients at the second level decomposition $L2$ using bi-orthogonal wavelet filter 9/7. We select the value 0.1 for the case 0 and 0.3 for the case 1 and 0.1 and 0.3 for the case 0 and 1. Based on the proposed method, the secret bit should be a real value because the extraction process depends on the difference between the modified and the predicted coefficients. In addition, the secret bit value should be a small value in order to not change the order of the GWT coefficients after the embedding process. In these

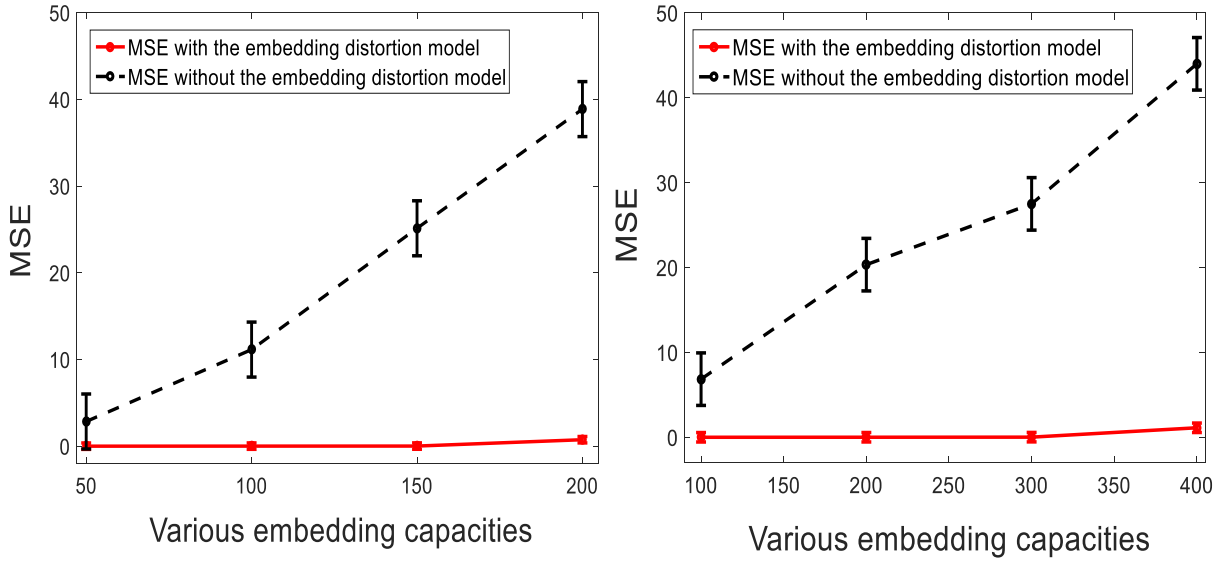
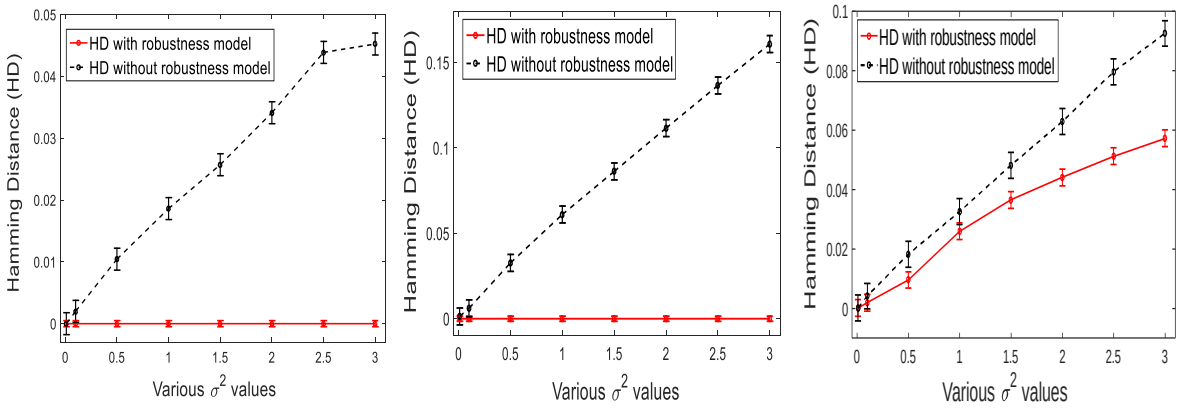
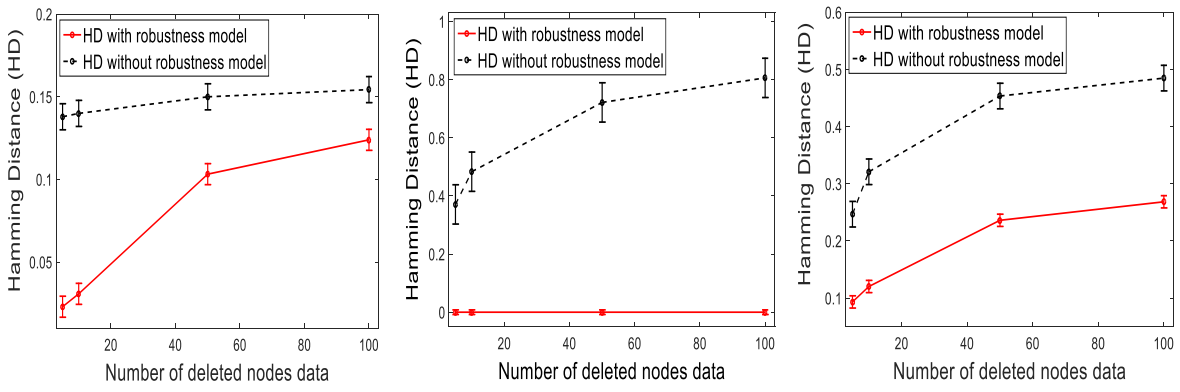


Figure 5.21: Embedding distortion performance of blind algorithm using graphs with different number of nodes for various embedding capacities. (a) $N = 2500$. (b) $N = 5000$.



(a) Hiding $b = 0$ (b) Hiding $b = 1$ (c) Hiding $b = 0, 1$

Figure 5.22: Robustness performance of non-blind algorithm to additive noise for various σ^2 values using 7 graphs types with $N = 500$. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.



(a) Hiding $b = 0$ (b) Hiding $b = 1$ (c) Hiding $b = 0, 1$

Figure 5.23: Robustness performance of non-blind algorithm after deleting various number of nodes data randomly using 7 graphs types with $N = 500$. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.

experiments, a set of graphs with 5000 nodes are used to evaluate the robustness model.

We can notice that the Hamming Distance (HD) of the extracted bits is decreased when using the proposed robustness model. This means the robustness is enhanced by using the proposed model for different values of σ^2 and after deleting various numbers of nodes data randomly. Figure 5.25 illustrates that the robustness against the additive noise is improved by an average of 36%, 99.9% and 44% for three embedding scenarios, $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$, respectively. It can be observed that the proposed method outperforms the original algorithm by an average of 99.9 % for three embedding scenarios, $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$,

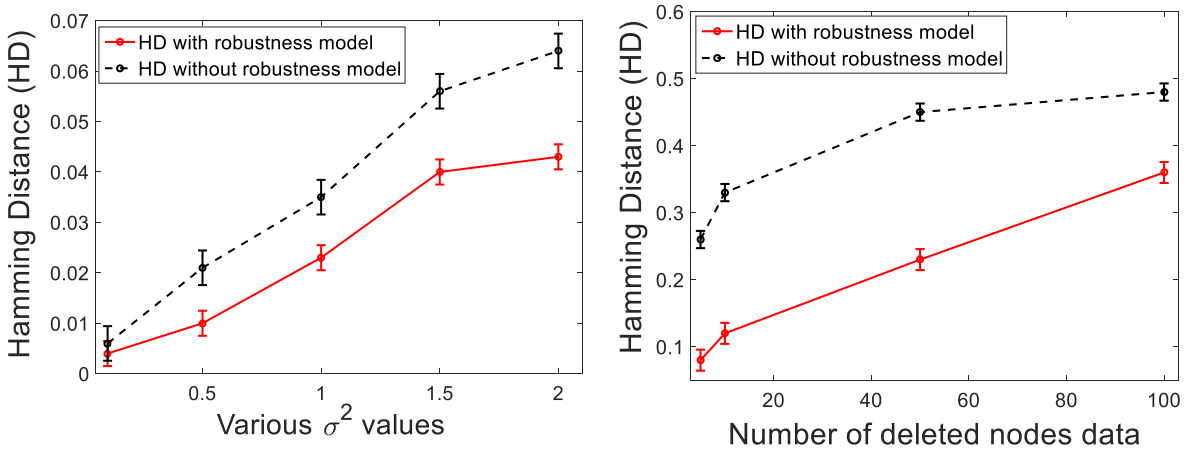


Figure 5.24: Robustness performance of non-blind data hiding after the attacks. (a) Additive noise. (b) Deleting nodes data.

respectively after deleting different number of nodes data randomly as shown in Figure 5.26

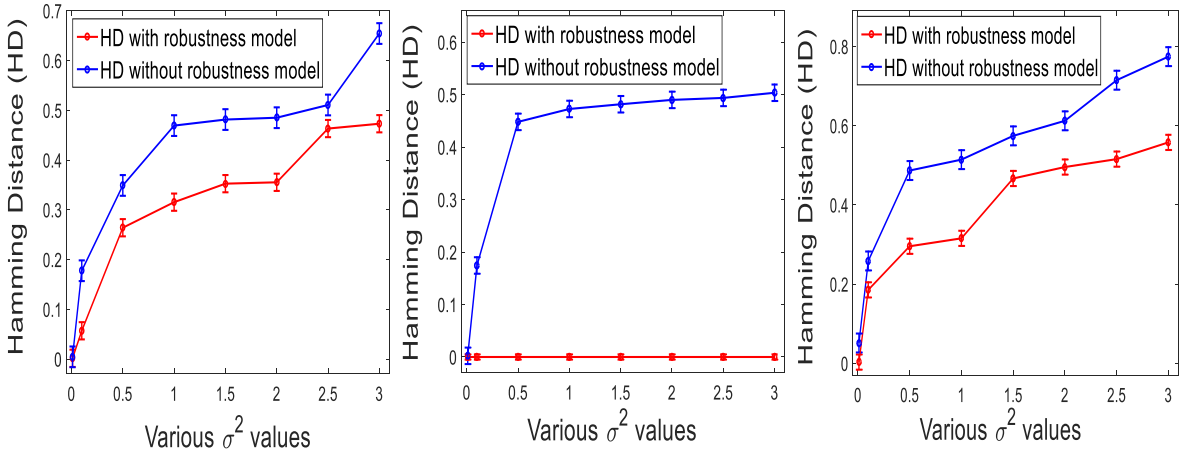
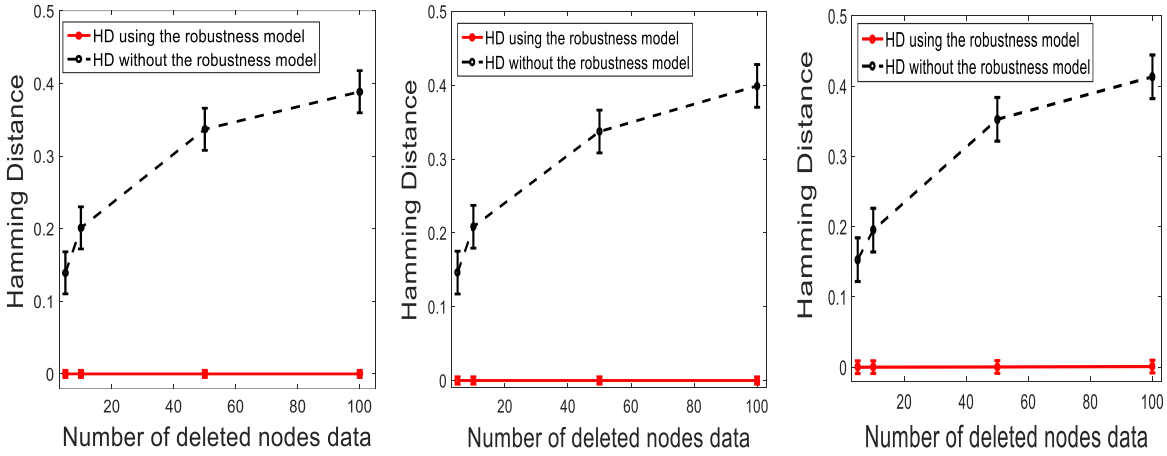


Figure 5.25: Hamming distance (HD) of the extracted bits after noise addition for different values of σ^2 using the robustness models. (a) Hiding $\mathbf{w} = \{0\}$. (b) Hiding $\mathbf{w} = \{1\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.

5.3.2.2.3 Robustness performance of the non-blind data hiding

The performance of the robustness of the non-blind algorithm against the noise additive and deletion nodes data is evaluated at various embedding capacities using graph dataset. We use 14 types of graphs with $N = 2500$ nodes and $\alpha = 0.5$ to evaluate the method using the Hamming



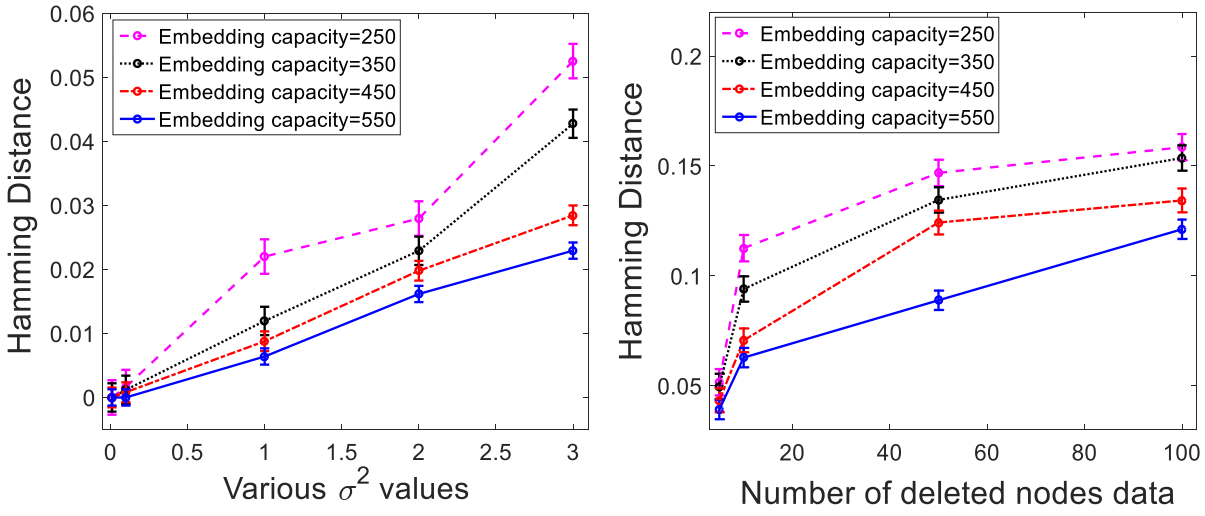
(a) Hiding $\mathbf{w} = \{0\}$ (b) Hiding $\mathbf{w} = \{1\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 5.26: Hamming distance (HD) of the extracted bits after deletion various number of random nodes data using the robustness models. (a) Hiding $\mathbf{w} = \{0\}$. (b) Hiding $\mathbf{w} = \{1\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.

Distance (HD) of the extracted bits for the algorithm with using the proposed robustness model (by embedding the secret bits in the GWT which satisfy the robustness conditions) after the additive noise for various σ^2 values and after deleting a different number of nodes data randomly at various embedding capacities. We consider the low frequency GWT coefficients at the second level decomposition $L2$ using bi-orthogonal wavelet filter 9/7 to embed the secret bits, $\mathbf{w} = \{0,\ ,1\}$. We can observe that the robustness performance is improved when increasing the embedding capacity as illustrated in Figure 5.27.

5.3.2.2.4 Robustness performance for blind data hiding

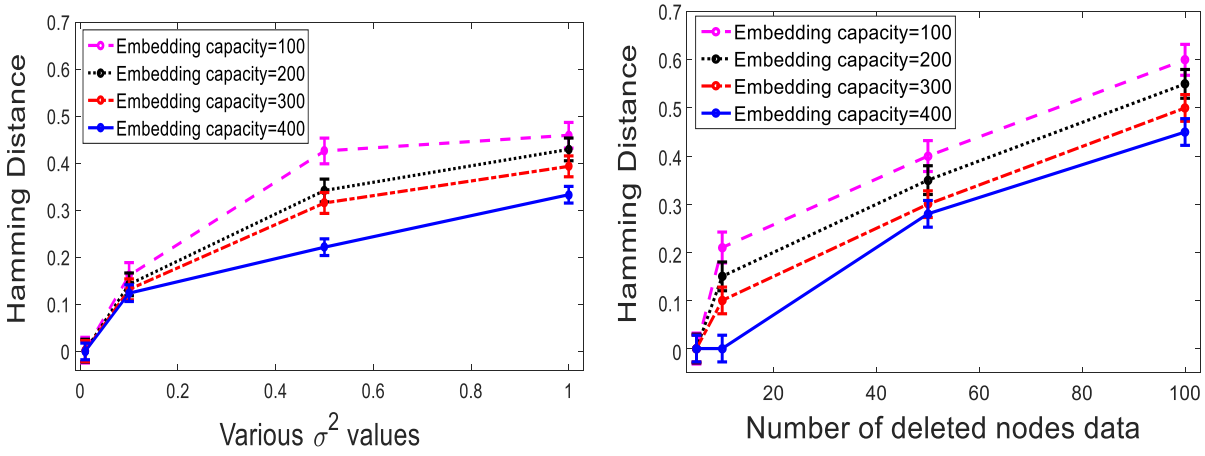
The performance of the robustness of the blind algorithm against the noise addition and deletion of node data is tested at various embedding capacities using graph dataset. In these experiments, we consider the low frequency GWT coefficients at the second level decomposition $L2$ using bi-orthogonal wavelet filter 9/7 for 14 graphs with $N = 5000$ nodes to embed the secret data ($\mathbf{w} = \{0.1,\ 0.3\}$ to represent $\mathbf{w} = \{0,\ 1\}$). The Hamming Distance (HD) of the extracted bits are calculated using the blind algorithm with using the proposed model (by embedding the secret data in the GWT which satisfy the robustness conditions) after the additive noise for various σ^2 values and deleting a different number of nodes data randomly at various embedding



(a) (b)

Figure 5.27: Robustness performance of non-blind algorithm using the proposed model after attacks for various embedding capacities using 14 graphs with $N = 2500$ nodes. (a) Additive noise. (b) Deletion nodes data.

capacities. The experimental results show that the robustness using the Hamming Distance is improved when the embedding capacity is increased as shown in Figure 5.28.



(a) (b)

Figure 5.28: Robustness performance of blind data hiding using the proposed model for various embedding capacities $N = 5000$. (a) Additive noise. (b) Deletion nodes data.

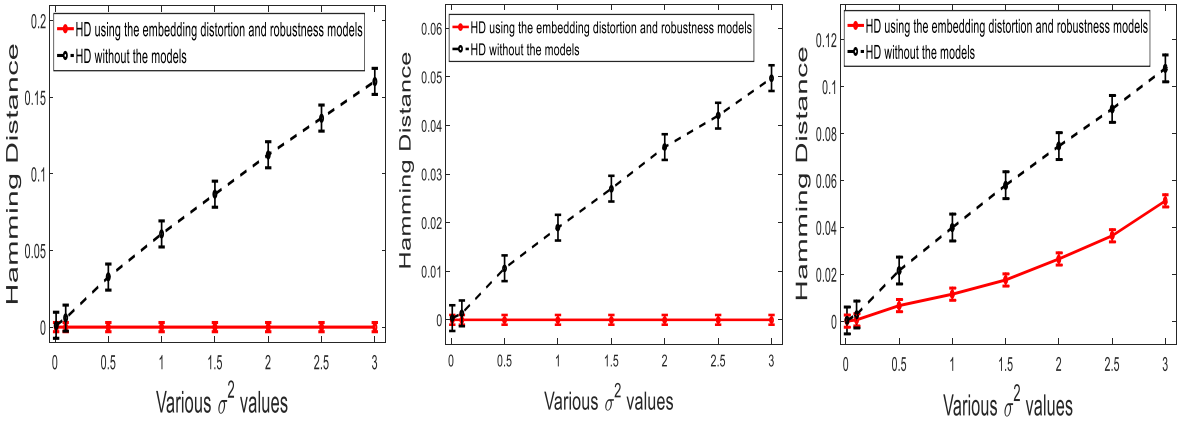
5.3.2.3 Joint robust-low distortion data hiding

For obtaining a data hiding method with low distortion and high robustness to attacks, we combine the embedding distortion minimisation model with the robustness model for non-blind and blind data hiding. In these experiments, we use 14 graphs with a number of nodes $N = 500$ and 5000, respectively and with $\alpha = 0.5$. We consider the low frequency GWT coefficients at the second level decomposition $L2$ using bi-orthogonal wavelet filter 9/7 to embed the secret data. We calculate the Hamming Distance of the extracted bits after the attacks using the data hiding algorithms with the two proposed models, i.e., embedding distortion minimisation and robustness models (based on selecting the GWT coefficients which satisfy the conditions of the two proposed models to embed the secret bits) and the Hamming Distance of the extracted bits after the attacks using the data hiding algorithms without the two proposed models (by embedding the same secret bits in any GWT coefficients randomly).

We can notice that the performance of the data hiding methods is enhanced by combining the two proposed models. The empirical results show that the robustness of non-blind algorithm is improved by an average of 99.9%, 99.9% and 67% after the additive noise and by an average of 99.9%, 69% and 50% after deleting nodes data for three embedding scenarios, $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$, respectively as shown in Figure 5.29 and Figure 5.30. The results (Figure 5.31 and Figure 5.32) demonstrate that robustness of blind algorithm is enhanced by an average of 99.9%, 48% and 37% after the additive noise and by an average of 99.9%, 99.9% and 99.9% after deleting nodes data for three embedding scenarios, $\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$, respectively.

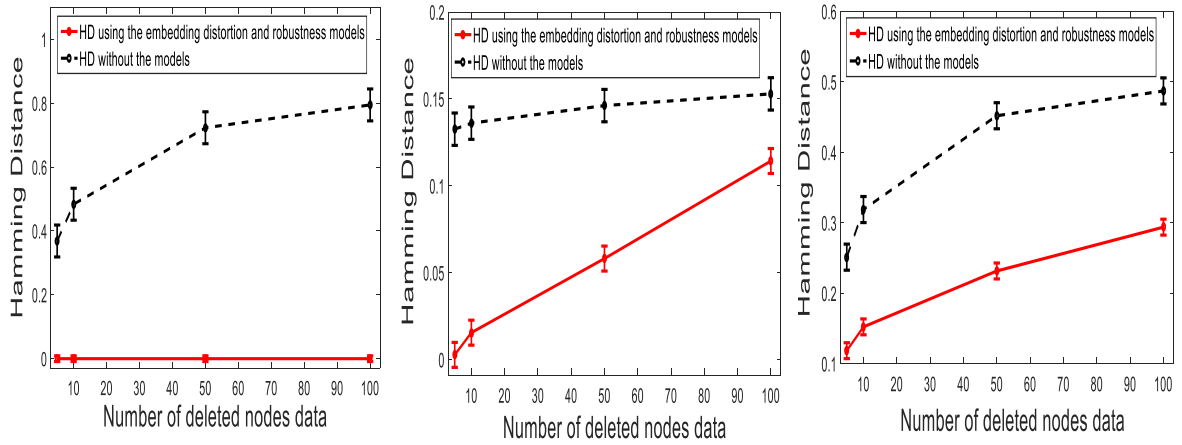
5.3.3 Performance evaluation of GWT reversible data hiding

The proposed method of reversible data hiding with two models have been evaluated by the experimental results. Four types of the experiments are presented: performance of embedding distortion, robustness performance, performance of reversibility of the original graph signal, and comparison the proposed method with two reversible data hiding methods, namely, Ni et al. [3] and Dragoi et al. [4] in terms of embedding distortion, robustness and reversibility.



(a) Hiding $\mathbf{w} = \{0\}$ (b) Hiding $\mathbf{w} = \{1\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 5.29: Hamming distance (HD) of extracted secret bits using the non-blind algorithm with the two models after noise addition for different values of σ^2 using 14 graphs with $N = 500$. (a) Hiding $\mathbf{w} = \{0\}$. (b) Hiding $\mathbf{w} = \{1\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.

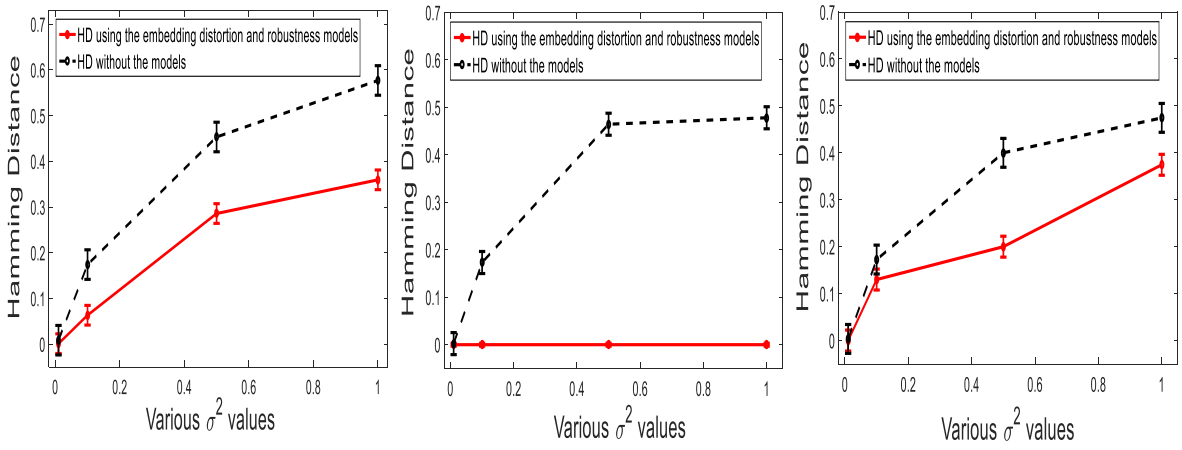


(a) Hiding $\mathbf{w} = \{0\}$ (b) Hiding $\mathbf{w} = \{1\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 5.30: Hamming distance (HD) of extracted secret bits using the non-blind algorithm with the two models after deleting a different number of random nodes data using 14 graphs with $N = 500$. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.

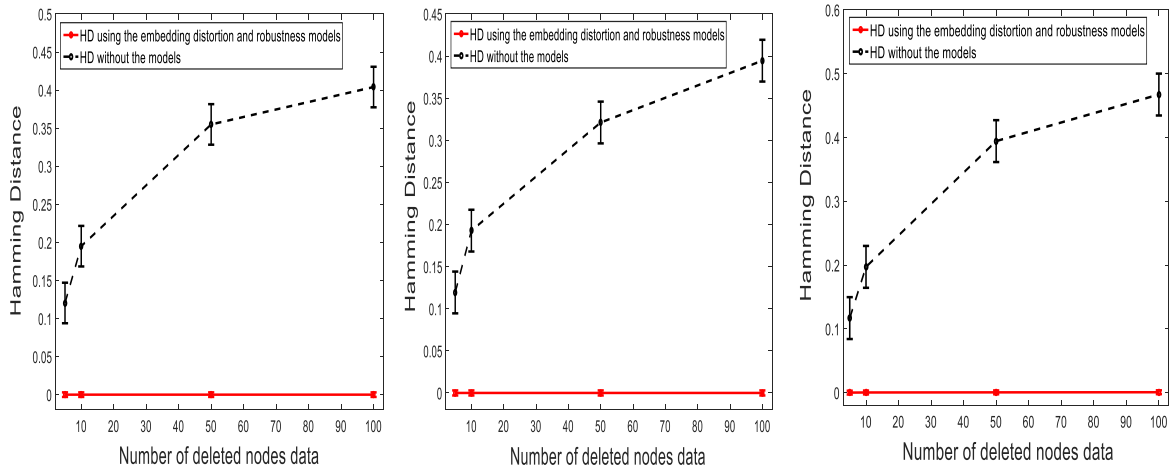
5.3.3.1 Evaluation of the embedding distortion performance

The experimental simulations verify the relationship between the MSE of the modified graph and the squared value of the embedded bits $\mathbf{w}^2$. The MSE of the modified graphs are calculated for different values of $\mathbf{w}$ using the graph dataset. We have considered the pseudo-random number sequences as the embedding data with nine scenarios: $\mathbf{w} = \{0, 0.1, 0.2, 0.4, 0.6, 0.8, 1, 2, 3\}$ to verify the relationship. In these experiments, 14 graphs types with a number of graph nodes



(a) Hiding $\mathbf{w} = \{0\}$ (b) Hiding $\mathbf{w} = \{1\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 5.31: Hamming distance (HD) of extracted bits using the blind algorithm with the two models after noise addition for different values of σ^2 using 14 graphs with $N = 5000$. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.



(a) Hiding $\mathbf{w} = \{0\}$ (b) Hiding $\mathbf{w} = \{1\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 5.32: Hamming distance (HD) of the extracted bits using the blind algorithm with the two models after deleting a different number of random nodes data using 14 graphs with $N = 5000$. (a) Hiding $\mathbf{w} = \{1\}$. (b) Hiding $\mathbf{w} = \{0\}$. (c) Hiding $\mathbf{w} = \{0, 1\}$.

$N = 5000$ and two graph wavelet filters, i.e, orthogonal Meyer filter and bi-orthogonal 9/7 filter are used to obtain the GWT coefficients for embedding various values of $\mathbf{w}$. Two sets of empirical results are calculated to verify the effects of embedding nine scenarios of the embedded data $\mathbf{w}$ as follows:

In the experiments, we embed various values of $\mathbf{w}$ in a peak point of the GWT coefficients magnitudes. We consider the low frequency and high-frequency GWT at the third level of de-

composition $L3$ and $H3$. The MSE of the modified graph has been calculated for each value of $\mathbf{w}$ separately. Figure 5.33 illustrates the relationship between the average value of the MSE of the modified different types of graphs for different values of $\mathbf{w} = \{0, 0.1, 0.2, 0.4, 0.6, 0.8, 1, 2, 3\}$ using two filters: orthogonal Meyer filter and bi-orthogonal 9/7 filter. We consider nine scenarios of embedding in order to demonstrate the proposed embedding distortion model instead of using only two cases, $\mathbf{w} = \{0, 1\}$, which are not enough to verify the proposed model. We can notice that the MSE of the modified graph is equal to zero when the value $\mathbf{w} = \{0\}$. For minimising the MSE of the modified graph, we have to select a small value of $\mathbf{w}$.

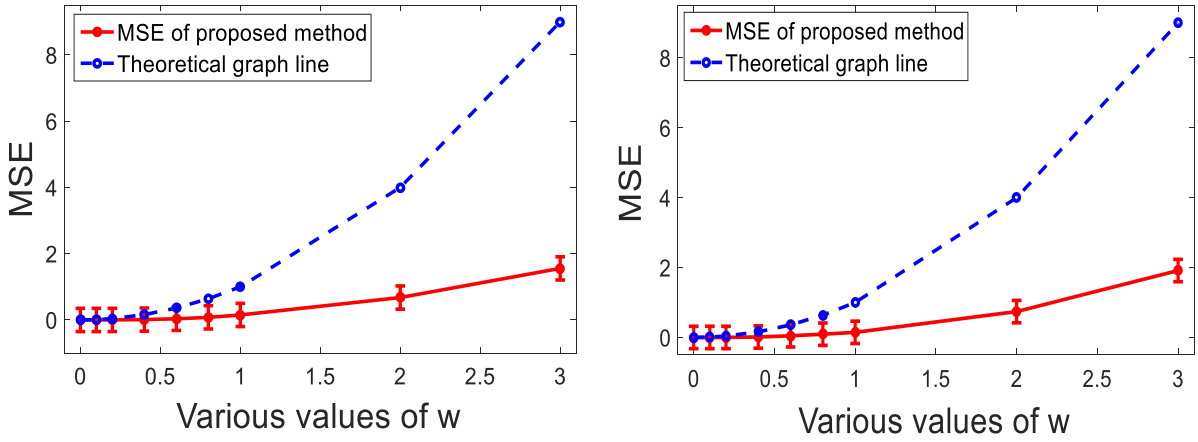


Figure 5.33: The average value of the MSE of modified graphs using various values of $\mathbf{w} = \{0, 0.1, 0.2, 0.4, 0.6, 0.8, 1, 2, 3\}$ and the theoretical graph line . (a) Orthogonal Meyer filter. (b) Bi-orthogonal 9/7 filter.

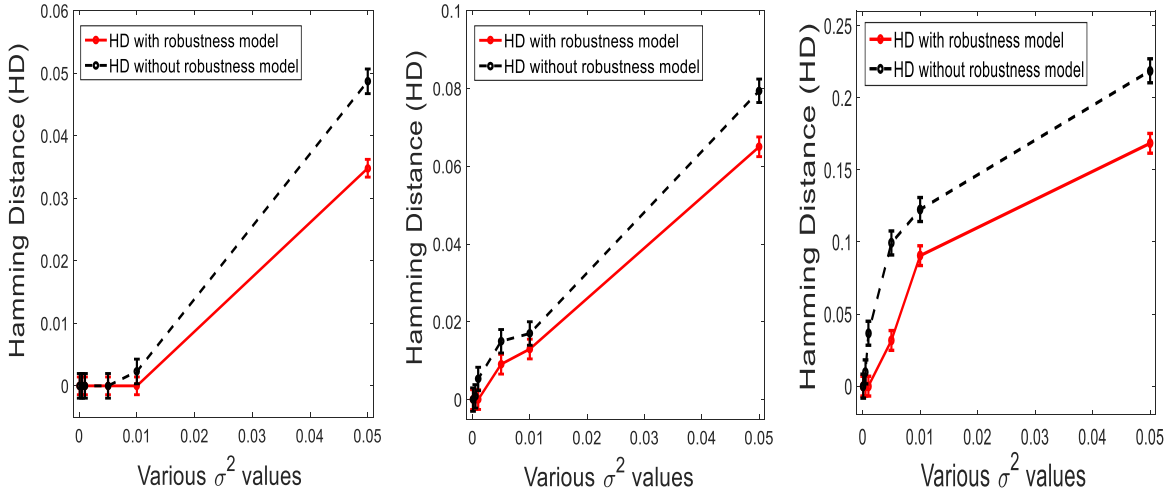
In order to verify the proposed model, we have plotted the theoretical graph line by assuming that the $x - axis$ represents value $\mathbf{w}$ and the $y - axis$ is the corresponding value $\mathbf{w}^2$ by considering various values of $\mathbf{w}$. The experimental results demonstrate the correlation between the proposed model and the theoretical graph line which support the proposed model.

5.3.3.2 Robustness performance

The robustness performance of the proposed method has been evaluated by using the Hamming Distance (HD) of the extracted secret data after the noise addition for various values of σ^2, $\sigma^2 = \{0.0001,\ 0.0005,\ 0.001,\ 0.005,\ 0.01,\ 0.05\}$. We calculate the HD of the extracted bits after the additive noise using the reversible data hiding algorithm with using the robustness model (by

embedding the secret bits in the GWT coefficients that satisfy the robustness conditions) and the HD of the extracted bits after the additive noise using the reversible data hiding algorithm without using the robustness model (by embedding the secret bits in any GWT coefficients). In these experiments, 14 graphs with $N = 5000$ nodes are utilised for evaluating the proposed method. We consider the GWT coefficients at the third level of decomposition $L3$ and $H3$ to hide the secret bits, $\mathbf{w} = \{0,\ 1\}$, for two types of wavelet filters: orthogonal Meyer filter and bi-orthogonal 9/7 filter.

The results demonstrate that the proposed method has achieved higher robustness over the original algorithm without the model. As shown in Figure 5.34 and Figure 5.35, the robustness against the additive noise is improved by an average of 63%, 48% and 51% using orthogonal Meyer filter and by an average of 58%, 33% and 54% using bi-orthogonal 9/7 filter for three embedding scenarios, $\mathbf{w} = \{0\}$, $\mathbf{w} = \{1\}$ and $\mathbf{w} = \{0,\ 1\}$, respectively.

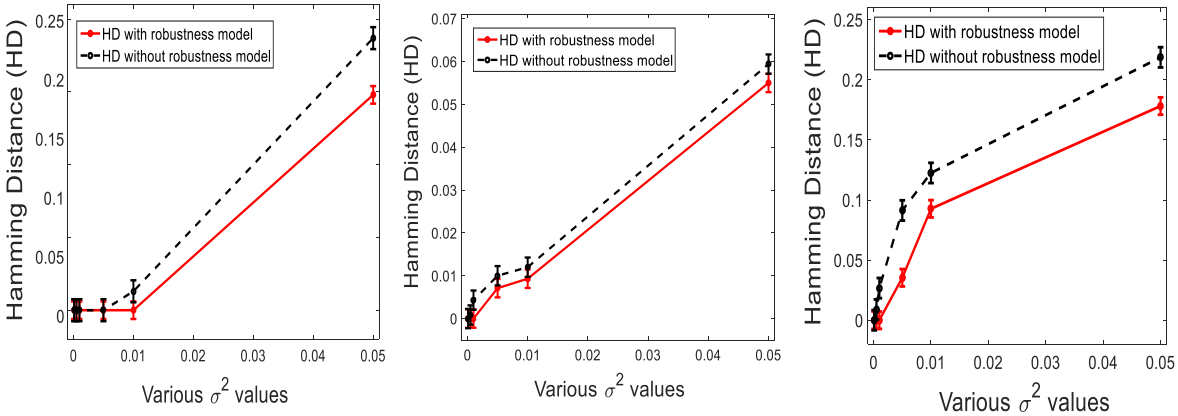


(a) Hiding $\mathbf{w} = \{0\}$ (b) Hiding $\mathbf{w} = \{1\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

Figure 5.34: Hamming distance (HD) of extracted secret bits after noise addition for different values of σ^2 using orthogonal Meyer filter. (a) Hiding $\mathbf{w} = \{0\}$. (b) Hiding $\mathbf{w} = \{1\}$. (c) Hiding $\mathbf{w} = \{0, 1\}$.

5.3.3.3 Reversibility performance

We have evaluated the proposed method in terms of reversibility of the original graph signal based on calculating the MSE of the restored graph signal after the embedded data have been extracted for different embedding rates. In these experiments, 14 graphs with $N = 5000$ nodes



(a) Hiding $\mathbf{w} = \{0\}$ (b) Hiding $\mathbf{w} = \{1\}$ (c) Hiding $\mathbf{w} = \{0,\ 1\}$

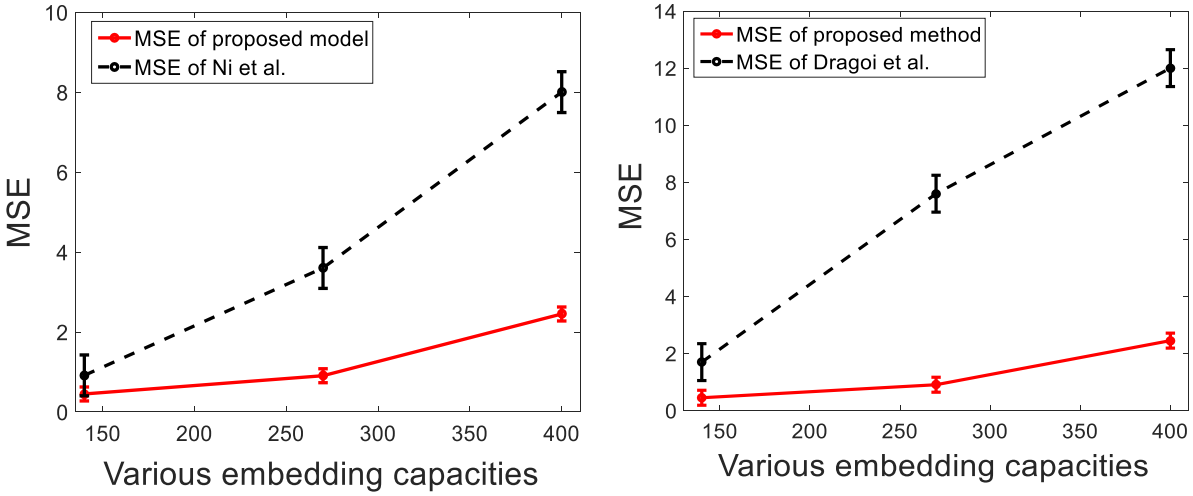
Figure 5.35: Hamming distance (HD) of extracted secret bits after noise addition for different values of σ^2 using bi-orthogonal 9/7 filter. (a) Hiding $\mathbf{w} = \{0\}$. (b) Hiding $\mathbf{w} = \{1\}$. (c) Hiding $\mathbf{w} = \{0,\ 1\}$.

are utilised for evaluating the proposed method. We consider the GWT coefficients at the third level of decomposition $L3$ and $H3$ to embed the secret bits $\mathbf{w} = \{0,\ 1\}$ using orthogonal Meyer filter. The proposed method has proved that it is able to restore the original graph signal with a free distortion for any payload in the case when no attack. This is mainly due to the shifting process which provides a lossless recovery of the original signal without requiring any side information except for one case if the histogram of the coefficients does not have zero points. In this case, the coefficients with the minimum number of repetition are used as zero points. This leads to loss of these coefficients due to the shifting process. In order to restore the original data without any error, these few numbers of the coefficients are added to the embedded data as a part of payload. Usually, this happens when the payload is big and this needs to shift the coefficients many times and to use several peaks and zero points. The proposed method overcomes this problem due to using Graph Wavelet Transform (GWT) and by using the advantage of the histogram characteristics of the graph spectral coefficients which provides several peak points and zero points.

5.3.3.4 Comparison with existing work

The proposed method has compared in terms of the embedding distortion, robustness and reversibility with two reversible data hiding methods, i.e., Ni et al. [3] and Dragoi et al. [4]. We

selected the same RDH methods for comparison in order to demonstrate the effect of using the graph wavelet domain compared to graph Fourier domain on the performance of the proposed method.. In these experiments, various graphs for various signals with $N = 10000$ nodes are used. We consider the GWT coefficients at the second level of wavelet decomposition $L2$ and $H2$ to embed the secret bits $\mathbf{w} = \{0,\ 1\}$ using orthogonal Meyer filter. The MSE of the modified graph has been calculated at various embedding capacities using graph dataset. Figure 5.36 shows that the proposed method outperforms the previous methods by an average of 68% and 82% for Ni et al. [3] and Dragoi et al. [4], respectively.



(a) (b)

Figure 5.36: Comparison the embedding distortion of the proposed method with Ni et al. [3] and Dragoi et al. [4] methods using MSE of the modified graphs for various embedding capacities. (a) Ni et al. [3] method. (b) Dragoi et al. [4] method.

The robustness performance of the proposed method has compared with two reversible data hiding methods, i.e., Ni et al. [3] and Dragoi et al. [4] based on comparing the Hamming Distance (HD) of the extracted secret data after the noise addition for various values of σ^2 using the proposed model and the comparable methods. In these experiments, various graphs for various signals with $N = 5000$ nodes are used. We consider the GWT coefficients at the second level of decomposition $L2$ and $H2$ to embed the secret bits $\mathbf{w} = \{0,\ 1\}$ using orthogonal Meyer filter. Figure 5.37 shows that the proposed method has achieved higher robustness to the noise addition over the Ni et al. [3] and Dragoi et al. [4] by an average of 78% and 92%, respectively.

We compare the reversibility of the proposed method with Ni et al. [3] and Dragoi et al. [4] methods by calculating the MSE of the recovered data after extracting the embedded bits for different embedding rates in the case of no attack. In these experiments, various graphs for various signals with $N = 5000$ nodes are used. We consider the GWT coefficients at the second level of decomposition $L2$ and $H2$ to hide the secret bits $\mathbf{w} = \{0,\ 1\}$ using orthogonal Meyer filter. The experimental results show that the proposed method and Ni et al. [3] method are able to restore the original data with error free in the case when no attack compared to Dragoi et al. [4]. We also compare the reversibility of the methods after the additive noise for various σ^2 values. Figure 5.38 shows that the proposed method outperforms the previous methods by an average of 95% and 99% for Ni et al. [3] and Dragoi et al. [4], respectively after the additive noise using the same graph signals.

5.3.4 Comparison the proposed GWT data hiding with the proposed GFT data hiding

This section presents a comparison of the proposed data hiding methods in two graph spectral domains: graph Fourier and graph wavelet. We use the graph dataset and the same embedding parameters such as data hiding parameter α, the number of graph nodes N and the length of the

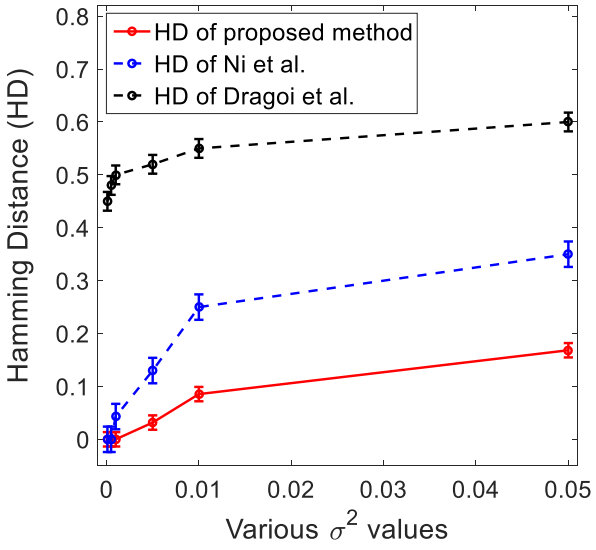
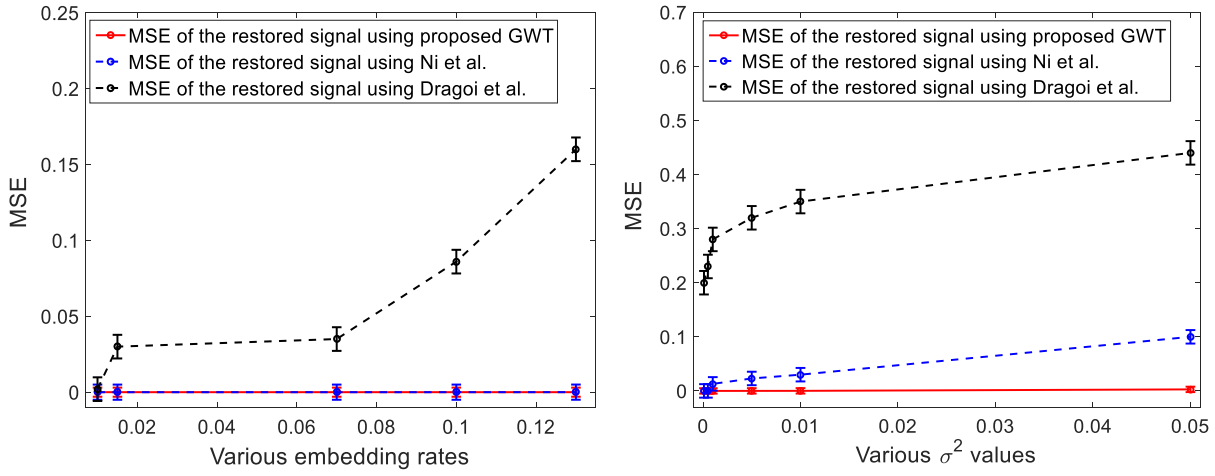


Figure 5.37: Comparison the robustness performance of the proposed method with Ni et al. [3] and Dragoi et al. [4] to additive noise for various σ^2 values.



(a) (b)

Figure 5.38: Comparison the reversibility performance of the proposed method with Ni et al. [3] and Dragoi et al. [4]. (a) Without additive noise. (b) After additive noise for various σ^2 values.

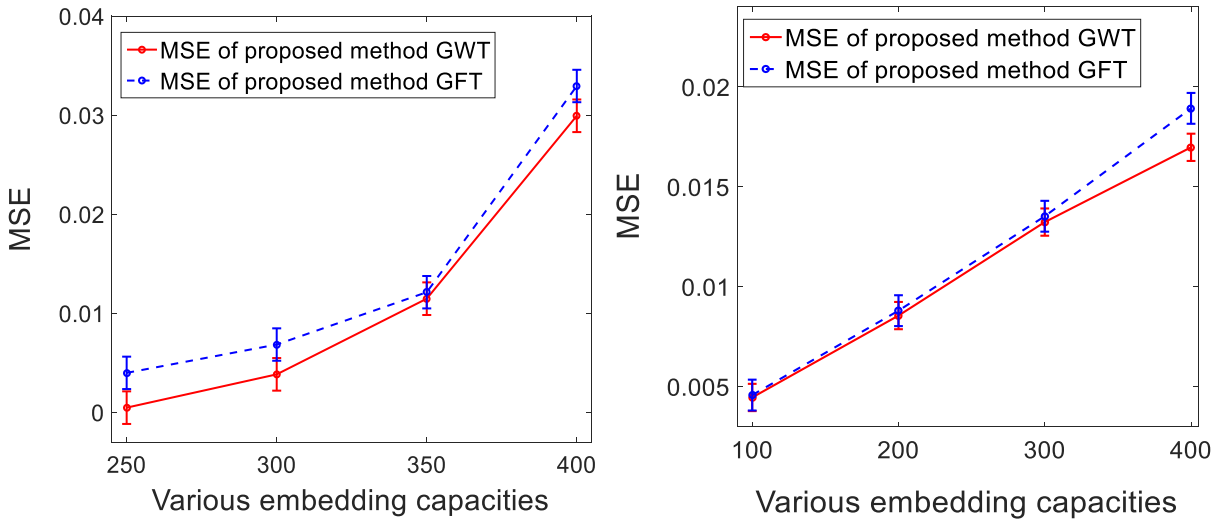
secret bits.

The performance of the embedding distortion of GWT data hiding with GFT data hiding has compared using the graph dataset. The MSE of the modified graphs for non-blind data hiding using GFT and non-blind data hiding using GWT have been calculated at various embedding capacities. In these experiments, we use 14 graphs with ($N = 2500$) nodes and the pseudo-random binary sequence is considered as the secret data, ($\mathbf{w} = \{0, 1\}$), to hide in the spectral coefficients using data hiding parameter ($\alpha = 0.1$). The GFT coefficients are selected to hide the secret bits based on the proposed embedding distortion minimisation model. For the GWT coefficients, we have selected the low frequency coefficients at third level of wavelet decompositions $L3$ using bi-orthogonal 9/7 filter to embed the secret bits based on using the embedding distortion minimisation model.

The embedding distortion of the proposed blind data hiding methods has evaluated for different embedding capacities. We calculate the MSE of the modified graphs for blind data hiding using GFT and blind data hiding using GWT for 14 graphs with ($N = 5000$) nodes. We consider the pseudo-random number sequences as the secret data to embed in the spectral coefficients, the value 0.1 is considered to represent the bit 0 and the 0.3 represents the bit 1. The GFT coefficients are chosen based on the proposed embedding distortion model. The low frequency GWT coefficients at the third level of wavelet decompositions $L3$ using bi-orthogonal 9/7 filter

are selected to hide the secret data.

The experimental results demonstrate that the proposed GWT data hiding methods provides lower distortion over the GFT data hiding methods. Figure 5.39 shows that the distortion is improved by an average of 39% and 5% for non-blind and blind algorithms in spite of using the same proposed models and the same data hiding parameters.



(a) (b)

Figure 5.39: Comparison the embedding distortion performance using MSE of the modified graphs with number of nodes $N = 2500$ and $N = 5000$. (a) Non-blind data hiding. (b) Blind data hiding.

The performance of the proposed non-blind data hiding methods have evaluated in terms of robustness against the attacks. Two attacks types are considered, namely, noise addition and deleting random nodes data. The Hamming Distance (HD) of the extracted secret data using the proposed GFT non-blind data hiding and GWT non-blind data hiding have been calculated. We use (14) graphs with ($N = 5000$) nodes, the pseudo-random binary sequences ($\mathbf{w} = \{0, 1\}$) are hidden in the spectral coefficients using a data hiding parameter ($\alpha = 0.5$). The low frequency GWT coefficients at the third level of wavelet decompositions $L3$ using bi-orthogonal 9/7 filter and GFT coefficients are selected based on specific conditions according to the proposed robustness models.

The robustness performance of the proposed blind data hiding has evaluated based on comparing the Hamming Distance (HD) of the extracted secret bits using the GFT blind data hiding and GWT blind data hiding after two attacks types, noise addition and deletion random nodes

data. In these experiments, we use 14 graphs with ($N = 5000$) nodes. We consider the pseudo-random number sequences as secret data (by using the values ($\{0.1,\ 0.3\}$ to represent the $\{0, 1\}$) to embed in the spectral coefficients. The low frequency GWT coefficients at the third level of wavelet decomposition $L3$ using bi-orthogonal 9/7 filter and GFT coefficients are selected based on specific conditions according to the proposed robustness models.

The empirical results show that the GWT data hiding methods have achieved higher robustness compared to the GFT data hiding methods by an average of 22% and 25% for non-blind algorithm and by an average of 21% and 87% for blind algorithm as illustrated in Figure 5.40 and Figure 5.41. This is mainly due to embedding the secret bits in the low-frequency GWT coefficients and using multi levels of decomposition (the third level of decomposition was used).

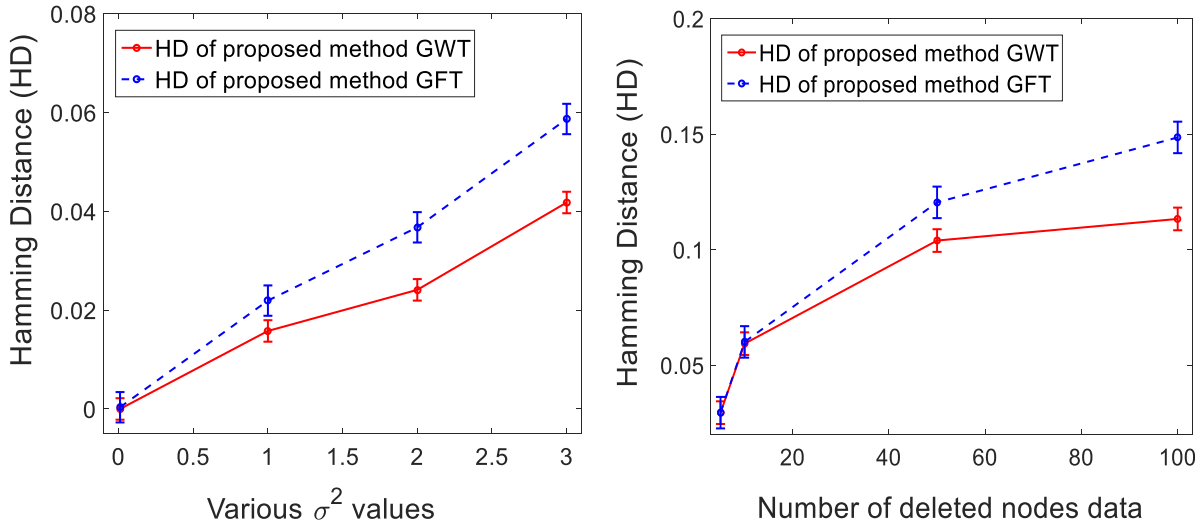
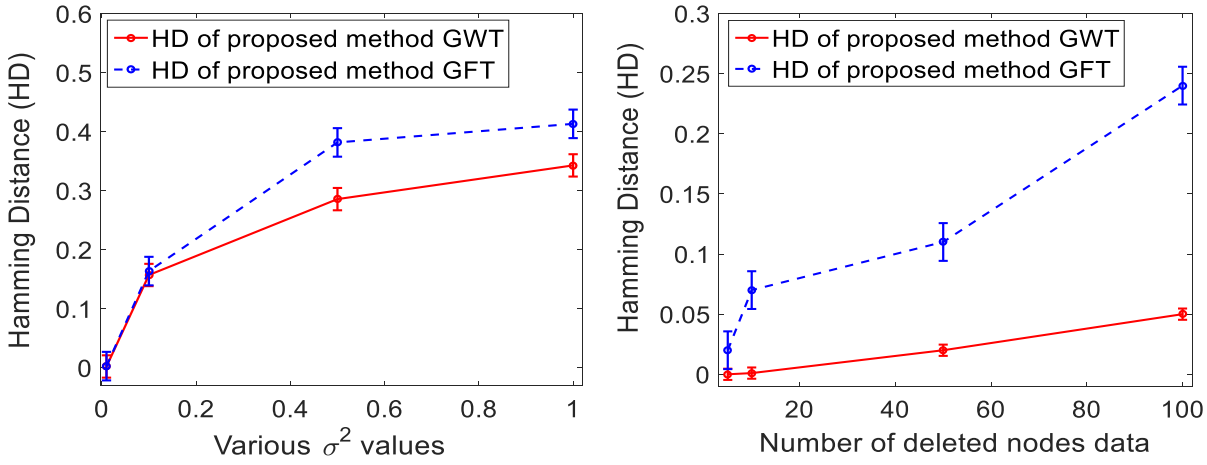


Figure 5.40: Comparison the robustness performance of non-blind data hiding using Hamming Distance (HD) of the extracted secret bits using 14 graphs with number of nodes $N = 5000$. (a) Noise addition. (b) Deleting random nodes data.

The empirical simulation shows that the performance of the proposed methods using graph wavelet domain is better than using the graph Fourier domain data hiding in terms of the embedding distortion and the robustness to the additive noise and deleting nodes data. This is mainly due its ability for representing the graph signal in time-frequency domain with a multi-resolution.



(a) (b)

Figure 5.41: Comparison the robustness performance of blind data hiding using Hamming Distance (HD) of the extracted secret bits using 14 graphs with number of nodes $N = 5000$. (a) Noise addition. (b) Deleting random nodes data.

5.3.5 Comparison the proposed reversible data hiding using GWT with the proposed GFT reversible data hiding

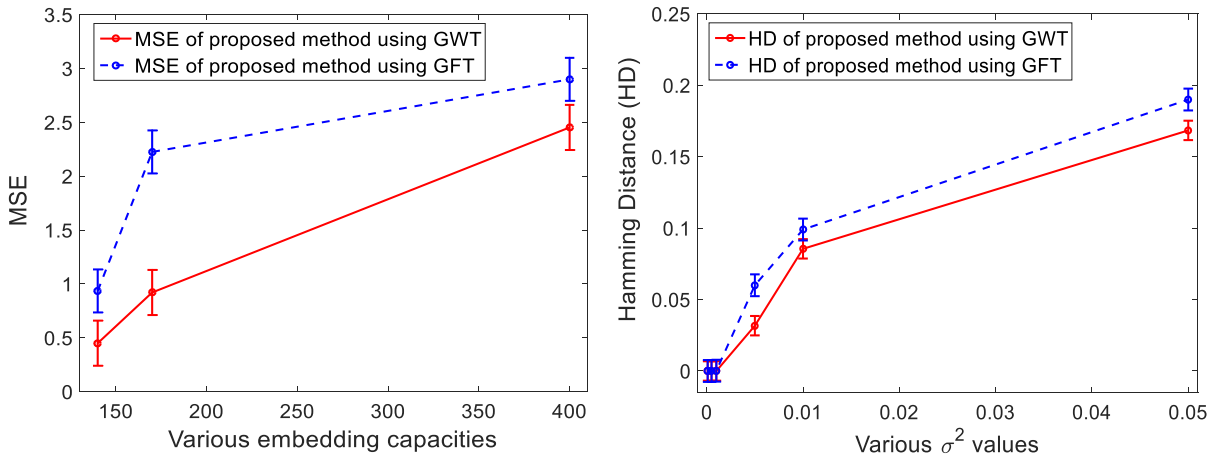
The performance of the proposed reversible data hiding methods is evaluated in two graph spectral domains: the graph Fourier domain and the graph wavelet domain. We use a graph dataset and the same embedding parameters, such as the length of hidden bits and the number of graph nodes.

Using a graph dataset, we compare the embedding distortion of the reversible data hiding that uses GWT and the reversible data hiding that uses GFT. We have calculated the MSE of the modified graphs for reversible data hiding using GWT and reversible data hiding using GFT for various embedding capacities. In these experiments, we used 14 graphs with $N = 10000$ nodes; we considered the pseudo-random binary sequences $\mathbf{w} = \{0, 1\}$ as embedded in the spectral coefficients. The secret data are hidden in the peak points of the GFT coefficient magnitudes. For the GWT coefficients, the low-frequency and high-frequency coefficients at the second level of wavelet decomposition ($L2$ and $H2$, using orthogonal filters) are used to select the peak points for hiding the secret bits.

We have evaluated the robustness performance of the proposed reversible data hiding methods by comparing the Hamming Distances (HDs) of the extracted secret bits, using reversible

data hiding in GFT and GWT after the additive noise . In these experiments, we used 14 graphs with $N = 10000$ nodes. The pseudo-random binary sequences $\mathbf{w} = \{0, 1\}$ are embedded in the spectral coefficients. We used the GFT coefficients and low-frequency and high-frequency GWT coefficients at the second level of decomposition ($L2$ and $H2$, using orthogonal filters) to select the peak points to use for hiding the secret bits.

The empirical results show that the proposed reversible data hiding method using GWT has achieved lower distortion and higher robustness compared to the GFT reversible data hiding by an average of 44% and 30%, respectively as shown in Figure 5.42. This is essentially due to using a graph wavelet transform, which represents the signal in a time-frequency domain with a multi-resolution decomposition. We hide the secret bits in the low-frequency and high-frequency GWT coefficients to create a balance between the embedding distortion and the robustness. Embedding in the high-frequency coefficients decreases embedding distortion, while hiding the secret data in the low-frequency coefficients increases the robustness to noise addition.



(a) (b)

Figure 5.42: Comparison the performance the proposed reversible data hiding using GWT with reversible data hiding using GFT for graphs with $N = 10000$ nodes. (a) Embedding distortion performance. (b) Robustness performance to additive noise.

We also evaluated the reversibility performance of the proposed reversible data hiding method in recovering the original graph signal, using both GWT and GFT RDH. The graph dataset was used to evaluate the proposed methods. The proposed methods using the GFT and GWT domains have proven that they can recover the original graph signal error-free for any payload,

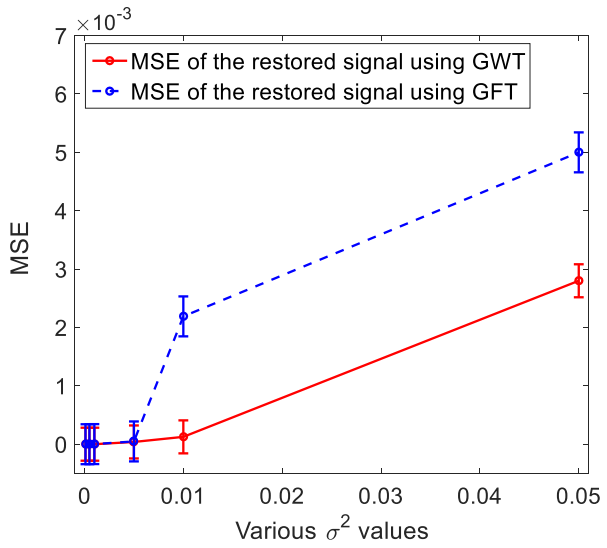


Figure 5.43: Comparison the reversibility performance of the proposed reversible data hiding using GWT with the proposed reversible data hiding using GFT after the additive noise using graph dataset with $N = 10000$ nodes.

in the case of no attack. This is primarily due to the shifting process, which provides a lossless recovery of the original signal without requiring any side information. In addition, we assessed the reversibility performance of the proposed methods after the noise addition. Notably, reversible data hiding using GWT can restore the original graph signal with less error than reversible data hiding using GFT. As shown in Figure 5.43, the proposed method using GWT has improved the reversibility of the original signal by an average of 72%. This is largely because GWT reversible data hiding is more robust, because it uses low-frequency coefficients and multiple levels of decompositions.

5.4 Concluding remarks

In this chapter, we have proposed data hiding methods in graph wavelet domain including irreversible and reversible data hiding. For irreversible data hiding, two scenarios are considered: non-blind and blind data hiding. The proposed methods involve novel models to minimise the embedding distortion on the graph signal and to make the secret data robust for attacks. The experimental results show that the embedding distortion is improved by an average of 99% and 99.4% for non-blind and blind data hiding respectively. In addition, the the robustness of non-

blind algorithm is improved by an average of 99.9%, 99.9% and 67% after the additive noise and by an average of 99.9%, 69% and 50% after deleting nodes data for three embedding scenarios, ($\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$), respectively. The robustness of blind algorithm is enhanced by an average of 99.9%, 48% and 37% after the additive noise and by an average of 99.9%, 99.9% and 99.9% after deleting nodes data for three embedding scenarios, ($\mathbf{w} = \{1\}$, $\mathbf{w} = \{0\}$ and $\mathbf{w} = \{0,\ 1\}$), respectively. The proposed GWT reversible data hiding method has compared with two reversible data hiding methods, namely, Ni et al. [3] and Dragoi et al. [4] in terms of embedding distortion, robustness to additive noise and the reversibility of the original signal. The empirical results illustrate that the proposed method outperforms the existing methods by an average of 68% and 82% in terms of the distortion. The proposed method has improved the robustness against the additive noise and reversibility of the original data by an average of 78%, 92%, 95% and 99% respectively compared to the existing methods. In addition, we have compared the performance of the proposed data hiding methods in graph wavelet domain with the proposed methods in graph Fourier domain. The experimental simulations have demonstrated that the graph wavelet domain data hiding methods superior the data hiding methods in graph Fourier domain.

Chapter 6

Conclusions

6.1 Summary of achievements

In this thesis, we have explored the benefit of the graph spectral domain for data hiding. We have utilised two graph spectral domains to embed the secret data. A new dataset has also been generated for data hiding. We have proposed three methods to embed the secret data using the graph spectral domain.

The first problem is protecting the data recorded on non-Cartesian grids. Unfortunately, traditional signal processing techniques cannot be applied to the irregular structure data. We have explored recently developed graph signal processing for protecting these data. The proposed methodology includes two new models: the embedding distortion minimisation model for minimising the embedding distortion resulting from embedding the secret data, and the robustness model for enhancing the robustness of the embedded data against attacks, namely, noise addition and deletion of nodes data. Finally, we have combined the proposed models to obtain a robust data hiding method with low embedding distortion. Two data hiding scenarios were considered: non-blind data hiding and blind data hiding in the graph Fourier domain. The original data hiding algorithms are compared both with and without using the proposed models. The experimental simulations demonstrate that the data hiding algorithms using the proposed models have achieved better performance (in terms of embedding distortion and robustness against attacks) than the same algorithms without using the proposed models. The results illustrate that the proposed methods using the embedding distortion minimisation model have achieved lower

distortion over the original methods by more than 94% and 80% for non-blind and blind algorithms, respectively. The proposed methods have compared in terms of the robustness against the attacks. We can see that the robustness of the proposed methods are improved by an average of 93% and 99.8% for non-blind and by an average of 60% and 71% for blind after the additive noise and deletion nodes data.

The second problem is proposing reversible data hiding algorithms for graph data that have irregular structure, for use when the traditional data hiding algorithms cannot recover the original host data. We have proposed the graph Fourier domain RDH method based on a shifting process for non-integer data. This proposed method includes two new models: the embedding distortion model to minimise distortion (resulting from the embedding process) and the robustness model to improve robustness against the attack – namely, additive noise. Finally, we have combined the proposed models to achieve maximum robustness with the lowest embedding distortion. The experimental results demonstrate that the proposed method outperforms the previous methods by an average of 87% and 92% in terms of the embedding distortion and by an average of 54% and 86% in terms of the robustness against the additive noise and by an average 97% and 99% in terms of reversibility of the original graph signal compared to Ni et al. [3] and Dragoi et al. [4] methods, respectively.

The third problem is exploring graph wavelet domain data hiding for unstructured data. Due to the properties of discrete wavelet transform (DWT) representing the signal in the time-frequency domain and with multi-resolution decomposition. DWT is considered an ideal option for data hiding. We exploit the properties of graph wavelet transforms to propose data hiding approaches, including irreversible and reversible data hiding. The proposed methodology of data hiding involves models for minimising distortion resulting from embedding the secret bits and for making the embedded data robust against attack. Finally, to obtain a robust approach with low distortion, we combine the proposed models. The proposed approaches were evaluated by comparing the original algorithms both with and without using the proposed models. The experimental simulations show that the proposed GWT data hiding method outperforms the original methods by an average of 99% and 99.4% for non-blind and blind data hiding in terms of embedding distortion. The robustness of the non-blind is enhanced by an average of 77%, 71%, 60% and 99% for non-blind after and blind after the additive noise and deleting nodes

data, respectively. The results show that the proposed GWT reversible data hiding method outperforms the previous methods by an average of 68%, 82%, 78%, 92%, 95% and 99% in terms of distortion, robustness and reversibility, respectively.

Finally, we have evaluated the technical methods used in this thesis by comparing the performance of the proposed methods in two spectral domains: graph Fourier and graph wavelet. The experimental evaluation shows that the proposed methods, including irreversible and reversible data hiding in the graph wavelet domain, are superior to the same data hiding methods in the graph Fourier domain. This is due primarily to the properties of a wavelet transform, which provides a multi-resolution decomposition and representation of the graph signal in the spatial-frequency domain.

6.2 Future directions

In this section, we have expanded the proposed contributions to this thesis to include two future directions as given:

1. Recently, the models of 3D point clouds have received a great attention in many applications such as urban centres and historical monuments. Therefore, the proposed graph data hiding methods can be extended to 3D point clouds models.

2. The proposed approach of reversible data hiding using the histogram shifting can be merged with the prediction error approach to enhance the performance of the proposed approach in terms of the embedding distortion.

Bibliography

[1] D. Bhowmik, "Robust watermarking techniques for scalable coded image and video," *Ph.D. dissertation, Dept. Elect. Eng., University of Sheffield, Sheffield, UK*, 2010.

[2] A. Sakiyama, K. Watanabe, Y. Tanaka, and A. Ortega, "Two-channel critically sampled graph filter banks with spectral domain sampling," *IEEE Transactions on Signal Processing*, vol. 67, no. 6, pp. 1447–1460, 2019.

[3] 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, 2006.

[4] 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 *Proc. of IEEE International Conference on European Signal Processing Conference (EUSIPCO)*, 2017, pp. 2186–2190.

[5] A. Shaik, V. Thanikaiselvan, and R. Amitharajan, "Data security through data hiding in images: a review," *J. Artificial Intelligence*, vol. 10, pp. 1–21, 2017.

[6] G. Qu and M. Potkonjak, "Analysis of watermarking techniques for graph coloring problem," in *Proc. of the IEEE/ACM International Conference on Computer-aided design*, 1998, pp. 190–193.

[7] A. Saha, D. Bhaumik, and S. Pathak, "Signature hiding and recovery in a graph coloring solutions using modified genetic algorithm," in *Proc. of IEEE International Conference on Computer and Information Technology (ICCIT)*, 2011, pp. 50–55.

[8] X. Zhao, Q. Liu, H. Zheng, and B. Y. Zhao, "Towards graph watermarks," in *Proc. of ACM on Conference on Online Social Networks*, 2015, pp. 101–112.

[9] D. Bhowmik and C. Abhayaratne, "A framework for evaluating wavelet based watermarking for scalable coded digital item adaptation attacks," in *Wavelet Applications in Industrial Processing VI*, vol. 7248, 2009, p. 72480M.

[10] ——, "The effect of quality scalable image compression on robust watermarking," in *Proc. of the International Conference on Digital Signal Processing*. IEEE, 2009, pp. 1–8.

[11] ——, "Video watermarking using motion compensated 2D+ t+ 2D filtering," in *Proc. of the 12th ACM Workshop on Multimedia and Security*, 2010, pp. 127–136.

[12] C. Abhayaratne and D. Bhowmik, "Scalable watermark extraction for real-time authentication of jpeg 2000 images," *Journal of Real-Time Image Processing*, vol. 6, no. 4, pp. 307–325, 2011.

[13] D. Bhowmik and C. Abhayaratne, "2D+t wavelet domain video watermarking," *Advances in Multimedia*, vol. 2012, p. 6, 2012.

[14] ——, "Robust watermarking for scalable image coding-based content adaptation," in *Proc. of IET Conference on Image Processing (IPR)*. IET, 2012, pp. 1–6.

[15] ——, "On robustness against JPEG2000: a performance evaluation of wavelet-based watermarking techniques," *Multimedia Systems*, vol. 20, no. 2, pp. 239–252, 2014.

[16] ——, "Quality scalability aware watermarking for visual content," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5158–5172, 2016.

[17] D. Bhowmik, M. Oakes, and C. Abhayaratne, "Visual attention-based image watermarking," *IEEE Access*, vol. 4, pp. 8002–8018, 2016.

[18] M. Oakes, D. Bhowmik, and C. Abhayaratne, "Global motion compensated visual attention-based video watermarking," *Journal of Electronic Imaging*, vol. 25, no. 6, p. 061624, 2016.

[19] H. Mareen, J. De Praeter, G. Van Wallendael, and P. Lambert, "A scalable architecture for uncompressed-domain watermarked videos," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 6, pp. 1432–1444, 2018.

[20] H. Fang, W. Zhang, H. Zhou, H. Cui, and N. Yu, "Screen-shooting resilient watermarking," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 6, pp. 1403–1418, 2019.

[21] D. Bhowmik and C. Abhayaratne, "Embedding distortion analysis in wavelet-domain watermarking," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 15, no. 4, pp. 1–24, 2019.

[22] S. Haddad, G. Coatrieux, A. Moreau-Gaudry, and M. Cozic, "Joint watermarking-encryption-JPEG-LS for medical image reliability control in encrypted and compressed domains," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2556–2569, 2020.

[23] S. Weinstein and P. Ebert, "Data transmission by frequency-division multiplexing using the discrete fourier transform," *IEEE Transactions on Communication Technology*, vol. 19, no. 5, pp. 628–634, 1971.

[24] M. Lang, H. Guo, J. E. Odegard, C. S. Burrus, and R. O. Wells, "Noise reduction using an undecimated discrete wavelet transform," *IEEE Signal Processing Letters*, vol. 3, no. 1, pp. 10–12, 1996.

[25] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Applied and Computational Harmonic Analysis*, vol. 30, no. 2, pp. 129–150, 2011.

[26] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[27] R. Anderson, R. Needham, and A. Shamir, "The steganographic file system," in *Proc. of International Workshop on Information Hiding*, 1998, pp. 73–82.

[28] C.-Y. Lin and S.-F. Chang, "Issues and solutions for authenticating mpeg video," in *Security and Watermarking of Multimedia Contents*, vol. 3657 of Proceedings of SPIE. International Society for Optics and Photonics, 1999, pp. 54–65.

[29] K. Stefan and A. Fabien, "Information hiding techniques for steganography and digital watermarking," *Artech House, London, UK*, 2000.

[30] J. Fridrich, M. Goljan, and A. C. Baldoza, "New fragile authentication watermark for images," in *Proc. of IEEE International Conference on Image Processing*, vol. 1. IEEE, 2000, pp. 446–449.

[31] C.-Y. Lin and S.-F. Chang, "Semifragile watermarking for authenticating jpeg visual content," in *Proc. of Security and Watermarking of Multimedia Contents II*, vol. 3971, 2000, pp. 140–151.

[32] C. Song, S. Sudirman, M. Merabti, and D. Llewellyn-Jones, "Analysis of digital image watermark attacks," in *Proc. of IEEE Consumer Communications and Networking Conference*, 2010, pp. 1–5.

[33] 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, 2011.

[34] F. Mintzer, J. Lotspiech, N. Morimoto, and T. Almaden, "Safeguarding digital library contents and users," *D-Lib Magazine*, vol. 3, no. 7/8, 1997.

[35] C. W. Honsinger, P. W. Jones, M. Rabbani, and J. C. Stoffel, "Lossless recovery of an original image containing embedded data," 2001, US Patent 6,278,791.

[36] B. Macq, "Lossless multiresolution transform for image authenticating watermarking," in *Proc. of the International Conference on European Signal Processing Conference (EUSIPCO),*. IEEE, 2000, pp. 1–4.

[37] C. D. Vleeschouwer, J.-F. Delaigle, and B. Macq, "Circular interpretation of bijective transformations in lossless watermarking for media asset management," *IEEE Transactions on Multimedia*, vol. 5, no. 1, pp. 97–105, 2003.

[38] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proc. of International Conference on Pattern Recognition (ICPR)*, 2010, pp. 2366–2369.

[39] S.-H. Wang and Y.-P. Lin, "Wavelet tree quantization for copyright protection watermarking," *IEEE Transactions on Image Processing*, vol. 13, no. 2, pp. 154–165, 2004.

[40] Y. Liu and J. Zhao, "A new video watermarking algorithm based on 1D DFT and Radon transform," *Signal Processing*, vol. 90, no. 2, pp. 626–639, 2010.

[41] J. Panyavaraporn, "Multiple video watermarking algorithm based on wavelet transform," in *Proc. of 13th International Symposium on Communications and Information Technologies (ISCIT)*. IEEE, 2013, pp. 397–401.

[42] L. Agilandeeswari and K. Ganesan, "A robust color video watermarking scheme based on hybrid embedding techniques," *Multimedia Tools and Applications*, vol. 75, no. 14, pp. 8745–8780, 2016.

[43] P. K. Sharma, "Analysis of image watermarking using least significant bit algorithm," *International Journal of Information Sciences and Techniques (IJIST)*, vol. 2, pp. 666–673, 2012.

[44] G. Kaur and K. Kaur, "Digital watermarking and other data hiding techniques," *International Journal of Innovative Technology and Exploring Engineering*, vol. 2, no. 5, pp. 181–183, 2013.

[45] A. K. Al-Asmari and F. A. Al-Enizi, "A pyramid-based watermarking technique for digital color images copyright protection," in *Proc. of the International Conference on Computing, Engineering and Information*. IEEE, 2009, pp. 44–47.

[46] J. R. Hernandez, M. Amado, and F. Perez-Gonzalez, "DCT-domain watermarking techniques for still images: Detector performance analysis and a new structure," *IEEE Transactions on Image Processing*, vol. 9, no. 1, pp. 55–68, 2000.

[47] E. E. Abdallah, A. B. Hamza, and P. Bhattacharya, "Video watermarking using wavelet transform and tensor algebra," *Signal, Image and Video Processing*, vol. 4, no. 2, pp. 233–245, 2010.

[48] P. Meerwald and A. Uhl, "Survey of wavelet-domain watermarking algorithms," in *Security and Watermarking of Multimedia Contents III*, vol. 4314. International Society for Optics and Photonics, 2001, pp. 505–516.

[49] R. Pickholtz, D. Schilling, and L. Milstein, "Theory of spread-spectrum communications-a tutorial," *IEEE Transactions on Communications*, vol. 30, no. 5, pp. 855–884, 1982.

[50] D. Kirovski and H. S. Malvar, "Spread-spectrum watermarking of audio signals," *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 1020–1033, 2003.

[51] M. Kuribayashi, "Coded spread spectrum watermarking scheme," in *Proc. of International Workshop on Digital Watermarking*. Springer, 2012, pp. 169–183.

[52] J. Fridrich, M. Goljan, and R. Du, "Invertible authentication," in *Security and Watermarking of Multimedia Contents III*, vol. 4314. International Society for Optics and Photonics, 2001, pp. 197–208.

[53] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 13, no. 8, pp. 890–896, 2003.

[54] D. Coltuc and J. Chassery, "Very fast watermarking by Reversible Contrast Mapping," *IEEE Signal Processing Letters*, vol. 14, no. 4, pp. 255–258, 2007.

[55] Y. Hu, H.-K. Lee, K. Chen, and J. Li, "Difference expansion based reversible data hiding using two embedding directions," *IEEE Transactions on Multimedia*, vol. 10, no. 8, pp. 1500–1512, 2008.

[56] 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, 2008.

[57] Z. Ni, Y. Q. Shi, N. Ansari, W. Su, Q. Sun, and X. Lin, "Robust lossless image data hiding designed for semi-fragile image authentication," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 4, pp. 497–509, 2008.

[58] W. Tai, C. Yeh, and 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, 2009.

[59] S. Jung, L. T. Ha, and S. Ko, "A new histogram modification based reversible data hiding algorithm considering the human visual system," *IEEE Signal Processing Letters*, vol. 18, no. 2, pp. 95–98, 2011.

[60] 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, 2012.

[61] 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, 2015.

[62] H. Wu, J. Dugelay, and Y. Shi, "Reversible image data hiding with contrast enhancement," *IEEE Signal Processing Letters*, vol. 22, no. 1, pp. 81–85, 2015.

[63] N. Biggs, E. K. Lloyd, and R. J. Wilson, *Graph Theory, 1736-1936*. Oxford University Press, 1986.

[64] S. Chen, R. Varma, A. Sandryhaila, and J. Kovačević, "Discrete signal processing on graphs: Sampling theory," *IEEE Transactions on Signal Processing*, vol. 63, no. 24, pp. 6510–6523, 2015.

[65] A. Ortega, P. Frossard, J. Kovačević, J. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[66] A. Agaskar and Y. M. Lu, "A spectral graph uncertainty principle," *IEEE Transactions on Information Theory*, vol. 59, no. 7, 2013.

[67] N. Leonardi and D. V. D. Ville, "Tight wavelet frames on multislice graphs," *IEEE Trans. on Signal Processing*, vol. 61, no. 13, pp. 3357–3367, 2013.

[68] A. Sakiyama, Y. Tanaka, T. Tanaka, and A. Ortega, "Efficient sensor position selection using graph signal sampling theory," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 6225–6229.

[69] M. Crovella and E. Kolaczyk, "Graph wavelets for spatial traffic analysis," in *Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies (INFOCOM)*, vol. 3, 2003, pp. 1848–1857.

[70] M. Gavish, B. Nadler, and R. R. Coifman, "Multiscale wavelets on trees, graphs and high dimensional data: Theory and applications to semi supervised learning." in *ICML*, 2010, pp. 367–374.

[71] S. Chen, F. Cerda, P. Rizzo, J. Bielak, J. H. Garrett, and J. Kovačević, "Semi-supervised multiresolution classification using adaptive graph filtering with application to indirect bridge structural health monitoring," *IEEE Transactions on Signal Processing*, vol. 62, no. 11, pp. 2879–2893, 2014.

[72] D. Thanou, D. I. Shuman, and P. Frossard, "Learning parametric dictionaries for signals on graphs," *IEEE Transactions on Signal Processing*, vol. 62, no. 15, pp. 3849–3862, 2014.

[73] F. Zhang and E. R. Hancock, "Graph spectral image smoothing using the heat kernel," *Pattern Recognition*, vol. 41, no. 11, pp. 3328–3342, 2008.

[74] Y. Iizuka and Y. Tanaka, "Depth map denoising using collaborative graph wavelet shrinkage on connected image patches," in *Proc. of IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 828–832.

[75] B. Alwaely and C. Abhayaratne, "Adaptive graph formulation for 3D shape representation," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 1947–1951.

[76] N. Biggs, N. L. Biggs, and B. Norman, *Algebraic graph theory*. Cambridge university press, 1993, vol. 67.

[77] W. Wang and K. Ramchandran, "Random multiresolution representations for arbitrary sensor network graphs," in *Proc. of IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 4, 2006, pp. IV–IV.

[78] S. Zhong, Y. Hu, and J. Lu, "A new geometric-transformation robust and practical embedding scheme for watermarking 2D vector maps in the graph spectral domain," in *Proc. of IEEE International Conference on Communications, Circuits and Systems*, vol. 1, 2006, pp. 24–30.

[79] S. Butler, "Algebraic aspects of the normalized Laplacian," in *Recent Trends in Combinatorics*, 2016, pp. 295–315.

[80] S. K. Narang and A. Ortega, "Perfect reconstruction two-channel wavelet filter banks for graph structured data," *IEEE Transactions on Signal Processing*, vol. 60, no. 6, pp. 2786–2799, 2012.

[81] ——, "Compact support biorthogonal wavelet filterbanks for arbitrary undirected graphs," *IEEE Transactions on Signal Processing*, vol. 61, no. 19, pp. 4673–4685, Oct 2013.

[82] D. I. Shuman, B. Ricaud, and P. Vandergheynst, "Vertex-frequency analysis on graphs," *Applied and Computational Harmonic Analysis*, vol. 40, no. 2, pp. 260–291, 2016.

[83] Y. Tanaka and A. Sakiyama, "M-channel oversampled graph filter banks," *IEEE Transactions on Signal Processing*, vol. 62, no. 14, pp. 3578–3590, 2014.

[84] A. Sakiyama and Y. Tanaka, "Oversampled graph laplacian matrix for graph filter banks," *IEEE Transactions on Signal Processing*, vol. 62, no. 24, pp. 6425–6437, 2014.

[85] D. I. Shuman, C. Wiesmeyr, N. Holighaus, and P. Vandergheynst, "Spectrum-adapted tight graph wavelet and vertex-frequency frames," *IEEE Transactions on Signal Processing*, vol. 63, no. 16, pp. 4223–4235, 2015.

[86] D. B. Tay, Y. Tanaka, and A. Sakiyama, "Near orthogonal oversampled graph filter banks," *IEEE Signal Processing Letters*, vol. 23, no. 2, pp. 277–281, 2016.

[87] A. Sakiyama, K. Watanabe, and Y. Tanaka, "Spectral graph wavelets and filter banks with low approximation error," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 3, pp. 230–245, 2016.

[88] N. Tremblay and P. Borgnat, "Subgraph-based filterbanks for graph signals," *IEEE Transactions on Signal Processing*, vol. 64, no. 15, pp. 3827–3840, 2016.

[89] D. B. Tay and A. Tanaka, Y.and Sakiyama, "Critically sampled graph filter banks with polynomial filters from regular domain filter banks," *Signal Processing*, vol. 131, pp. 66–72, 2017.

[90] K. Watanabe, A. Sakiyama, Y. Tanaka, and A. Ortega, "Critically-sampled graph filter banks with spectral domain sampling," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4054–4058.

[91] Y. Jin and D. I. Shuman, "An M-channel critically sampled filter bank for graph signals," in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 3909–3913.

[92] V. N. Ekambaram, G. C. Fanti, B. Ayazifar, and K. Ramchandran, "Spline-like wavelet filterbanks for multiresolution analysis of graph-structured data," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 1, no. 4, pp. 268–278, 2015.

[93] M. S. Kotzagiannidis and P. L. Dragotti, "Splines and wavelets on circulant graphs," *Applied and Computational Harmonic Analysis*, vol. 47, no. 2, pp. 481–515, 2019.

[94] P. P. Vaidyanathan, "Multirate systems and filter banks," *Prentice Hall Signal Processing Series*, 1993.

[95] A. V. Oppenheim and R. W. Schafer, *Adaptive Discrete-Time Signal Processing*. Prentice Hall, 2009.

[96] M. Vetterli, J. Kovačević, and V. K. Goyal, *Foundations of signal processing*. Cambridge University Press, 2014.

[97] I. Pesenson, "Sampling in paley-wiener spaces on combinatorial graphs," *Transactions of the American Mathematical Society*, vol. 360, no. 10, pp. 5603–5627, 2008.

[98] X. Wang, P. Liu, and Y. Gu, "Local-set-based graph signal reconstruction," *IEEE Transactions on Signal Processing*, vol. 63, no. 9, pp. 2432–2444, 2015.

[99] A. Anis, A. Gadde, and A. Ortega, "Efficient sampling set selection for bandlimited graph signals using graph spectral proxies," *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3775–3789, 2016.

[100] M. Tsitsvero, S. Barbarossa, and P. Di Lorenzo, "Signals on graphs: Uncertainty principle and sampling," *IEEE Transactions on Signal Processing*, vol. 64, no. 18, pp. 4845–4860, 2016.

[101] Y. Tanaka, "Spectral domain sampling of graph signals," *IEEE Transactions on Signal Processing*, vol. 66, no. 14, pp. 3752–3767, 2018.

[102] S. K. Narang and A. Ortega, "Perfect reconstruction two-channel wavelet filter banks for graph structured data," *IEEE Transactions on Signal Processing*, vol. 60, no. 6, pp. 2786–2799, 2012.

[103] ——, "Compact support biorthogonal wavelet filterbanks for arbitrary undirected graphs," *IEEE Transactions on Signal Processing*, vol. 61, no. 19, pp. 4673–4685, 2013.

[104] A. Cohen, I. Daubechies, and J.-C. Feauveau, "Biorthogonal bases of compactly supported wavelets," *Communications on Pure and Applied Mathematics*, vol. 45, no. 5, pp. 485–560, 1992.

[105] S. K. Pal and S. S. Sarma, "Graph coloring approach for hiding of information," *Procedia Technology*, vol. 4, pp. 272–277, 2012.

[106] G. Qu and M. Potkonjak, "Hiding signatures in graph coloring solutions," in *Proc. of International Workshop on Information Hiding*, 1999, pp. 348–367.

[107] A. G. Bors, "Watermarking mesh-based representations of 3-D objects using local moments," *IEEE Transactions on Image processing*, vol. 15, no. 3, pp. 687–701, 2006.

[108] X. Gao, C. Zhang, Y. Huang, and Z. Deng, "A robust high-capacity affine-transformation-invariant scheme for watermarking 3D geometric models," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 8, no. 2S, p. 34, 2012.

[109] J.-W. Cho, R. Prost, and H.-Y. Jung, "An oblivious watermarking for 3-D polygonal meshes using distribution of vertex norms," *IEEE Transactions on Signal Processing*, vol. 55, no. 1, pp. 142–155, 2007.

[110] R. Darazi, R. Hu, and B. Macq, "Applying spread transform dither modulation for 3D-mesh watermarking by using perceptual models," in *Proc. of IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*. IEEE, 2010, pp. 1742–1745.

[111] J.-S. Tsai, J.-T. Hsiao, W.-B. Huang, and Y.-H. Kuo, "Geodesic-based robust blind watermarking method for three-dimensional mesh animation by using mesh segmentation and vertex trajectory," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 1757–1760.

[112] X. Rolland-Neviere and P. Doerr, G.and Alliez, "Security analysis of radial-based 3D watermarking systems," in *Proc. of IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2014, pp. 30–35.

[113] X. Rolland-Neviere, G. Doërr, and P. Alliez, "Anti-cropping blind resynchronization for 3D watermarking," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 1702–1706.

[114] A. G. Bors and M. Luo, "Optimized 3D watermarking for minimal surface distortion," *IEEE Transactions on Image Processing*, vol. 22, no. 5, pp. 1822–1835, 2013.

[115] J.-U. Hou, D.-G. Kim, and H.-K. Lee, "Blind 3D mesh watermarking for 3D printed model by analyzing layering artifact," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 11, pp. 2712–2725, 2017.

[116] S. Borah and B. Borah, "Three-dimensional (3D) polygon mesh authentication using sequential bit substitution strategy," in *Computational Intelligence in Data Mining*, 2020, pp. 617–627.

[117] M. R. Mouhamed, M. M. Soliman, A. Darwish, and A. Hassanien, "A robust and blind 3D mesh watermarking approach based on particle swarm optimization," *International Journal of Sociotechnology and Knowledge Development (IJSKD)*, vol. 12, no. 1, pp. 24–48, 2020.

[118] O. Khalil, A. Elhadad, and A. Ghareeb, "A blind proposed 3D mesh watermarking technique for copyright protection," *The Imaging Science Journal*, pp. 1–10, 2020.

[119] M. Corsini, E. D. Gelasca, T. Ebrahimi, and M. Barni, "Watermarked 3-D mesh quality assessment," *IEEE Transactions on Multimedia*, vol. 9, no. 2, pp. 247–256, 2007.

[120] N. Medimegh, S. Belaid, M. Atri, and N. Werghi, "3D mesh watermarking using salient points," *Multimedia Tools and Applications*, pp. 1–23, 2018.

[121] H. Singh, "Watermarking image encryption using deterministic phase mask and singular value decomposition in fractional mellin transform domain," *IET Image Processing*, vol. 12, no. 11, pp. 1994–2001, 2018.

[122] B. Ahmaderaghi, F. Kurugollu, J. M. D. Rincon, and A. Bouridane, "Blind image watermark detection algorithm based on discrete shearlet transform using statistical decision theory," *IEEE Transactions on Computational Imaging*, vol. 4, no. 1, pp. 46–59, March 2018.

[123] W. Hu, R. Zhou, A. El-Rafei, and S. Jiang, "Quantum image watermarking algorithm based on haar wavelet transform," *IEEE Access*, vol. 7, pp. 121 303–121 320, 2019.

[124] A. Kamble and S. S. Agrawal, "Wavelet based digital image watermarking algorithm using fractal images," in *Proc. of the IEEE International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 2019, pp. 1220–1224.

[125] F. Cayre and B. Macq, "Data hiding on 3-D triangle meshes," *IEEE Trans. on Signal Processing*, vol. 51, no. 4, pp. 939–949, 2003.

[126] T. Xu, Y. Zhang, J. Sun, and Z. Lin, "Watermarking 3D meshes based on fixed spectral basis," in *Proc. of the International Conference on Computational Intelligence and Security Workshops*, Dec 2007, pp. 640–643.

[127] R. Ohbuchi, S. Takahashi, T. Miyazawa, and A. Mukaiyama, "Watermarking 3D polygonal meshes in the mesh spectral domain," in *Graphics Interface*, vol. 2001, 2001, pp. 9–17.

[128] R. Ohbuchi, A. Mukaiyama, and S. Takahashi, "A frequency-domain approach to watermarking 3D shapes," in *Computer Graphics Forum*, vol. 21, no. 3, 2002, pp. 373–382.

[129] E. E. Abdallah, A. B. Hamza, and P. Bhattacharya, "Spectral graph-theoretic approach to 3D mesh watermarking," in *Proceedings of Graphics Interface*, 2007, pp. 327–334.

[130] Y. Liu, B. Prabhakaran, and X. Guo, "A robust spectral approach for blind watermarking of manifold surfaces," in *Proc. of the ACM Workshop on Multimedia and Security*, 2008, pp. 43–52.

[131] K. Wang, M. Luo, A. G. Bors, and F. Denis, "Blind and robust mesh watermarking using manifold harmonics," in *Proc. of IEEE International Conference on Image Processing (ICIP)*, 2009, pp. 3657–3660.

[132] Y. Liu, B. Prabhakaran, and X. Guo, "Spectral watermarking for parameterized surfaces," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 5, pp. 1459–1471, 2012.

[133] X. Feng, W. Zhang, and Y. Liu, "Double watermarks of 3D mesh model based on feature segmentation and redundancy information," *Multimedia Tools and Applications*, vol. 68, no. 3, pp. 497–515, 2014.

[134] X. Qi, S. Shi, and X. Yang, "3D point cloud model watermarking algorithm based on feature points extraction," *Journal of Computer Applications*, vol. 5, p. 023, 2014.

[135] F. Xiaoqing, "A watermarking for 3D point cloud model using distance normalization modulation," in *Proc. of IEEE International Conference on Computer Science and Network Technology (ICCSNT)*, vol. 1. IEEE, 2015, pp. 1449–1452.

[136] J. Shang, L. Sun, W. Wang, Y. Qin, and Z. Zhou, "Holographic digital blind watermark algorithm for 3D point cloud model based on discrete cosine transform," *Packaging Engineering*, vol. 13, p. 026, 2015.

[137] J. Liu, Y. Yang, D. Ma, Y. Wang, and Z. Pan, "A watermarking algorithm for 3D point cloud models using ring distribution," in *Transactions on Edutainment XIV*. Springer, 2018, pp. 56–68.

[138] S. Kanai, T. Date, H.and Kishinami *et al.*, "Digital watermarking for 3D polygons using multiresolution wavelet decomposition," in *Proc. Sixth IFIP WG*, vol. 5, 1998, pp. 296–307.

[139] F. Uccheddu, M. Corsini, and M. Barni, "Wavelet-based blind watermarking of 3D models," in *Proceedings of the Workshop on Multimedia and Security*, 2004, pp. 143–154.

[140] S. Valette and P. Prost, "Wavelet-based multiresolution analysis of irregular surface meshes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 10, no. 2, pp. 113–122, 2004.

[141] M.-S. Kim, S. Valette, H.-Y. Jung, and R. Prost, "Watermarking of 3D irregular meshes based on wavelet multiresolution analysis," in *Proc. of International Workshop on Digital Watermarking*, 2005, pp. 313–324.

[142] K. Wang, G. Lavoué, F. Denis, and A. Baskurt, "Hierarchical blind watermarking of 3D triangular meshes," in *Proc. of IEEE International Conference on Multimedia and Expo*, 2007, pp. 1235–1238.

[143] ——, "Hierarchical watermarking of semiregular meshes based on wavelet transform," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 4, pp. 620–634, 2008.

[144] J. Y. Kim, D.-H. Im, H.-Y. Lee, and H.-K. Lee, "Watermarking curves using 2D mesh spectral transform," in *Proc. of IEEE International Symposium on Circuits and Systems*, May 2008, pp. 2969–2972.

[145] M. Hachani *et al.*, "Wavelet based watermarking on 3D irregular meshes," in *Proc. of IEEE International Conference on European Signal Processing Conference (EUSIPCO)*, 2012, pp. 1742–1746.

[146] M. Hamidi, M. E. Haziti, H. Cherifi, and D. Aboutajdine, "A robust blind 3-D mesh watermarking based on wavelet transform for copyright protection," in *Proc. of IEEE International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, May 2017, pp. 1–6.

[147] M. Hamidi, A. Chetouani, M. El Haziti, M. El Hassouni, and H. Cherifi, "Blind robust 3D mesh watermarking based on mesh saliency and wavelet transform for copyright protection," *Information*, vol. 10, no. 2, p. 67, 2019.

[148] S. Yi and Y. Zhou, "Adaptive code embedding for reversible data hiding in encrypted images," in *Proc. of IEEE International Conference on Image Processing (ICIP)*, Sep. 2017, pp. 4322–4326.

[149] I. C. Dragoi and D. Coltuc, "Reversible data hiding in encrypted images based on reserving room after encryption and multiple predictors," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2102–2105.

[150] 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.

[151] I. C. Dragoi and D. Coltuc, "Reversible data hiding in encrypted color images based on vacating room after encryption and pixel prediction," in *Proc. of IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 1673–1677.

[152] Y. Wang, Z. Cai, and W. He, "A new high capacity separable reversible data hiding in encrypted images based on block selection and block-level encryption," *IEEE Access*, vol. 7, pp. 175 671–175 680, 2019.

[153] Y. Qiu, Q. Ying, X. Lin, Y. Zhang, and Z. Qian, "Reversible data hiding in encrypted images with dual data embedding," *IEEE Access*, vol. 8, pp. 23 209–23 220, 2020.

[154] H. Wu, X. Li, Y. Zhao, and R. Ni, "Improved PPVO-based high-fidelity reversible data hiding," *Signal Processing*, vol. 167, p. 107264, 2020.

[155] N. Li and F. Huang, "Reversible data hiding for JPEG images based on pairwise nonzero AC coefficient expansion," *Signal Processing*, p. 107476, 2020.

[156] F. Peng, Y. Zhao, X. Zhang, M. Long, and W.-q. Pan, "Reversible data hiding based on RSBEMD coding and adaptive multi-segment left and right histogram shifting," *Signal Processing: Image Communication*, vol. 81, p. 115715, 2020.

[157] T. Zhang, X. Li, W. Qi, and Z. Guo, "Prediction-error value ordering for high-fidelity reversible data hiding," in *Proc. of the International Conference on Multimedia Modeling*, 2020, pp. 317–328.

[158] A. Malik, H.-X. Wang, Y. Chen, and A. N. Khan, "A reversible data hiding in encrypted image based on prediction-error estimation and location map," *Multimedia Tools and Applications*, pp. 1–24, 2020.

[159] Y. Ke, M.-Q. Zhang, J. Liu, T.-T. Su, and X.-Y. Yang, "Fully homomorphic encryption encapsulated difference expansion for reversible data hiding in encrypted domain," *IEEE Transactions on Circuits and Systems for Video Technology*, 2020.

[160] G. Kaur, S. Singh, and R. Rani, "A high capacity reversible data hiding technique based on pixel value ordering using interlock partitioning," in *Proc. of the IEEE International Conference on Signal Processing and Integrated Networks (SPIN)*. IEEE, 2020, pp. 727–732.

[161] X. Yin, W. Lu, J. Zhang, and W. Liu, "Reversible data hiding in halftone images based on minimizing the visual distortion of pixels flipping," *Signal Processing*, p. 107605, 2020.

[162] C. Zhang, B. Ou, and D. Tang, "An improved VLC mapping method with parameter optimization for reversible data hiding in JPEG bitstream," *Multimedia Tools and Applications*, pp. 1–18, 2020.

[163] X. Gao, Z. Pan, E. Gao, and G. Fan, "Reversible data hiding for high dynamic range images using two-dimensional prediction-error histogram of the second time prediction," *Signal Processing*, p. 107579, 2020.

[164] 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.

[165] P. Wang and C. Wang, "Reversible data hiding for point-sampled geometry." *Journal of Information Science & Engineering*, vol. 23, no. 6, 2007.

[166] Z.-M. Lu and Z. Li, "High capacity reversible data hiding for 3D meshes in the PVQ domain," in *Proc. of the International Workshop on Digital Watermarking*, 2007, pp. 233–243.

[167] W. Hong, T.-S. Chen, and J. Chen, "Reversible data hiding using delaunay triangulation and selective embedment," *Information Sciences*, vol. 308, pp. 140–154, 2015.

[168] L. Li, S. Wang, S. Zhang, T. Luo, and C.-C. Chang, "Homomorphic encryption-based robust reversible watermarking for 3D model," *Symmetry*, vol. 12, no. 3, p. 347, 2020.

[169] H. Lee, Ç. Dikici, G. Lavoué, and F. Dupont, "Joint reversible watermarking and progressive compression of 3D meshes," *The Visual Computer*, vol. 27, no. 6-8, pp. 781–792, 2011.

[170] R. Jiang, W. Zhang, D. Hou, H. Wang, and N. Yu, "Reversible data hiding for 3D mesh models with three-dimensional prediction-error histogram modification," *Multimedia Tools and Applications*, pp. 1–18, 2017.

[171] W. Shah, M.and Zhang, H. Hu, H. Zhou, and T. Mahmood, "Homomorphic encryption-based reversible data hiding for 3D mesh models," *Arabian Journal for Science and Engineering*, vol. 43, no. 12, pp. 8145–8157, 2018.

[172] H. Luo, Z.-M. Lu, and J.-S. Pan, "A reversible data hiding scheme for 3D point cloud model," in *Proc. of the IEEE International Symposium on Signal Processing and Information Technology*. IEEE, 2006, pp. 863–867.

[173] H.-t. Wu and J.-L. Dugelay, "Reversible watermarking of 3D mesh models by prediction-error expansion," in *Proc. of the Workshop on Multimedia Signal Processing*. IEEE, 2008, pp. 797–802.

[174] Q. Zhang, X. Song, T. Wen, and C. Fu, "Reversibility improved data hiding in 3D mesh models using prediction-error expansion and sorting," *Measurement*, vol. 135, pp. 738–746, 2019.

[175] S. Borah and B. Borah, "Prediction error expansion (PEE) based reversible polygon mesh watermarking scheme for regional tamper localization," *Multimedia Tools and Applications*, pp. 1–22, 2020.

[176] D. Bhowmik, M. Oakes, and C. Abhayaratne, "Visual attention-based image watermarking," *IEEE Access*, vol. 4, pp. 8002–8018, 2016.

[177] 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, 1997.

[178] B. Kumar, H. V. Singh, S. P. Singh, A. Mohan *et al.*, "Secure spread-spectrum watermarking for telemedicine applications," *Journal of Information Security*, vol. 2, no. 02, p. 91, 2011.

[179] S. Liu, B. M. Hennelly, C. Guo, and J. T. Sheridan, "Robustness of double random phase encoding spread-space spread-spectrum watermarking technique," *Signal Processing*, vol. 109, pp. 345–361, 2015.

[180] S. Borah and B. Borah, "Quantization index modulation (QIM) based watermarking techniques for 3D meshes," in *Proc. of IEEE International Conference on Image Information Processing (ICIIP)*. IEEE, 2017, pp. 1–6.

[181] H. Al-khafaji and C. Abhayaratne, "Graph watermarking dataset," https://figshare.shef.ac.uk/s/3d92615f2a87f0ea719e, 2019, accessed: 18-02-2019.

[182] N. Perraudin, J. Paratte, D. Shuman, L. Martin, V. Kalofolias, P. Vandergheynst, and D. K. Hammond, "GSPBOX: A toolbox for signal processing on graphs," *ArXiv e-prints*, 2014.

[183] X. Chen, X. Li, B. Yang, and Y. Tang, "Reversible image watermarking based on a generalized integer transform," in *Proc. of IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, 2010, pp. 2382–2385.

[184] J. Zhou and O. C. Au, "On the determination of capacity parameters in pee-based reversible image watermarking," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 1809–1812.

[185] X. Gui, X. Li, and B. Yang, "Efficient reversible data hiding based on two-dimensional pixel-intensity-histogram modification," in *Proc of IEEE International Conference onAcoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 7420–7424.

[186] I. Caciula and D. Coltuc, "Improved control for low bit-rate reversible watermarking," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 7425–7429.

[187] Z. Yin, A. Abel, X. Zhang, and B. Luo, "Reversible data hiding in encrypted image based on block histogram shifting," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 2129–2133.

[188] I. C. Dragoi and D. Coltuc, "Reversible data hiding in encrypted images based on reserving room after encryption and multiple predictors," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2102–2105.

[189] N. Wang and X. Zhao, "2D vector map reversible data hiding with topological relation preservation," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2097–2101.

[190] D. Chou, C.-Y. Jhou, S.-C. Chu *et al.*, "Reversible watermark for 3D vertices based on data hiding in mesh formation," *International Journal of Innovative Computing, Information and Control*, vol. 5, no. 7, 2009.

[191] H. Wu and Y. Cheung, "A reversible data hiding approach to mesh authentication," in *Proc. of IEEE/WIC/ACM International Conference on Web Intelligence*, 2005, pp. 774–777.

[192] C.-Y. Jhou, J.-S. Pan, and D. Chou, "Reversible data hiding base on histogram shift for 3D vertex," in *Proc. of IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, vol. 1, 2007, pp. 365–370.

[193] Z. Sun, Z. Lu, and Z. Li, "Reversible data hiding for 3D meshes in the PVQ-compressed domain," in *Proc. of IEEE International Conf. on Intelligent Information Hiding and Multimedia Signal Processing*, 2006, pp. 593–596.

[194] G. Xuan, Q. Yao, C. Yang, J. Gao, P. Chai, Y. Q. Shi, and Z. Ni, "Lossless data hiding using histogram shifting method based on integer wavelets," in *Proc. of International Workshop on Digital Watermarking: (IWDW)*, vol. 4283, 2006, p. 323.

[195] R. Jiang, H. Zhou, W. Zhang, and N. Yu, "Reversible data hiding in encrypted three-dimensional mesh models," *IEEE Transactions on Multimedia*, vol. 20, no. 1, pp. 55–67, 2018.