

Reversible Data Hiding in Grayscale Images Using Prediction Error Expansion Based Techniques

A thesis submitted to University of Hyderabad in partial fulfillment

for the degree of

Doctor of Philosophy

by

Ravi Uyyala

Reg. No. 14MCPC15



SCHOOL OF COMPUTER AND INFORMATION SCIENCES

UNIVERSITY OF HYDERABAD

HYDERABAD -500046

Telangana

India

March, 2020



CERTIFICATE

This is to certify that the thesis entitled “**Reversible Data Hiding in Grayscale Images Using Prediction Error Expansion Based Techniques**” submitted by **Ravi Uyyala** bearing **Reg. No. 14MCPC15** in partial fulfillment of the requirements for the award of **Doctor of Philosophy in Computer Science** is a bonafide work carried out by him under my supervision at the **Institute for Development and Research in Banking Technology, Hyderabad**. This thesis is free from plagiarism and has not been submitted previously in part or in full to this or any other University or Institution for award of any degree or diploma. Parts of this thesis have been published online in the following publications:

- 1 IET Image Processing, October-2019 (Chapter 3)
- 2 Proceedings of CVIP-2016, IIT Roorkee (Chapter 5)
- 3 Proceedings of SPIN-2016, Noida (Chapter5)
- 4 Proceedings of IEEE TenSymp-2018, Sydney, Australia (Chapter 4)
- 5 Proceedings of SPIN-2020, Noida (Chapter 5)

Further, the student has passed the following courses towards fulfilment of course-work requirement for Ph.D:

	Course Code	Name	Credits	Pass/Fail
1	BT701	Data structure and algorithms	4	Pass
2	BT702	Operating system and programming	4	Pass
3	BT704	Digital image processing and computer vision	4	Pass
4	BT705	Pattern recognition	4	Pass

Supervisor	Director	Dean
Dr. Rajarshi Pal	Dr. A.S. Ramasastry	Prof. Chakravarthy Bhagvati
IDRBT	IDRBT	SCIS, UoH
Hyderabad-500 057, India	Hyderabad-500 057, India	Hyderabad-500 046, India

DECLARATION

I, **Ravi Uyyala**, hereby declare that this thesis entitled “**Reversible Data Hiding in Grayscale Images Using Prediction Error Expansion Based Techniques**” submitted by me under the supervision of **Dr. Rajarshi Pal**, is a bonafide research work and is free from plagiarism. I also declare that it has not been submitted previously in part or in full to this University or any other University or Institution for the award of any degree or diploma. I hereby agree that my thesis can be deposited in Shodganga/INFLIBNET.

A report on plagiarism statistics from the University Librarian is enclosed.

Date:

Signature of the Student

(Ravi Uyyala)

Reg. No.: 14MCPC15

//Countersigned//

Signature of the Supervisor

(Dr. Rajarshi Pal)

Dedicated To My Family & Teachers

Acknowledgements

I am deeply indebted to **Dr. Rajarshi Pal**, my thesis supervisor for his guidance, wisdom and support during the course of my Ph. D at IDRBT. His consistent encouragement and positive reinforcement have made it a gratifying experience. I have learned from him the virtues of rigor and professionalism. I am grateful to him for keeping me focused at the times when it mattered a lot. I express deep respect and gratitude to my supervisor for astute guidance and constant inspiration, without which this work would have never been possible.

I thank my doctoral review committee members, Prof. B. M. Mehtre and Dr. P. Syam Kumar for their encouragement, insightful comments and hard questions which strengthened my knowledge.

It is my privilege to thank **Dr. A. S. Ramasastry**, Director, IDRBT for his considerate support and encouragement throughout the tenure of my research work and for extending the facilities to pursue my research. I thank **Prof. Chakravarthy Bhagvati**, Dean, School of Computer and Information Sciences (SCIS), University of Hyderabad, Hyderabad, for his academic support during my research work.

I express my sincere gratitude to every member of IDRBT family for providing a smooth and enriching environment so that I enjoyed the tenure at the institute. I thank every member at my lab for providing a lovely experience to me.

I, specially, thank my wife, daughter and other family members for their support and inspiration during the Ph.D work. At the end, I am grateful to **Institute for Development and Research in Banking Technology** (IDRBT) and **University of Hyderabad** (UoH), for making it a memorable experience.

Abstract

Reversible data hiding is a special kind of data hiding technique, where original cover media can be restored along with extraction of hidden data. In this thesis, reversible data hiding is discussed in the context of grayscale image as cover media. Several types of reversible data hiding techniques exist in literature. Prediction error expansion based reversible data hiding techniques exhibit superiority in performance over other types of reversible data hiding techniques. In a prediction error expansion based technique, a pixel value is predicted using a pixel prediction strategy. Then, data bit is added in the expanded prediction error of a pixel. A good pixel prediction strategy is key to this technique. A small prediction error leads to less embedding distortion.

In this thesis, several novel reversible data hiding techniques are proposed by exploiting several strategies for pixel value prediction. In the first of the proposed techniques, B-tree triangular decomposition technique is used to obtain a set of reference pixels. Non-reference pixel values are interpolated (predicted) using these reference pixel values. In the second of the proposed techniques, reference pixels are randomly distributed throughout an image. Non-reference pixel values are predicted using weighted median of the values at the nearby reference pixels. In the third of the proposed technique, a pixel value is predicted as an average of few linearly predicted values in the selected directional contexts. A few directions are selected by analyzing the pixel values in an 8-neighborhood of the pixel. Similarly, in the fourth of the proposed techniques, gradient estimations at several directions are used to select directional contexts. Then, a weighted average of

two linearly predicted values in the selected directions provides the final predicted value of the pixel. Finally, the performances of several neighborhood-based and gradient-based predictors are compared to highlight the need of a multi-predictor strategy. A novel multi-predictor strategy is proposed to combine the outcomes of multiple predictors by considering the median of these values. Adaptive embedding strategies are used in each of the proposed reversible data hiding techniques. Either one bit or two bits of data are embedded in the prediction error of a pixel depending on local complexity of the pixel.

Finally, the proposed reversible data hiding techniques are applied to ensure integrity of bank cheque image in an image-based cheque clearing environment. Performances of the proposed reversible data hiding techniques are observed in this context.

Contents

Acknowledgments	v
Abstract	vi
List of Figures	xii
List of Tables	xvi
1 Introduction	1
1.1 Reversible Data Hiding	3
1.2 Properties of Reversible Data Hiding	3
1.2.1 Imperceptibility of Change	4
1.2.2 Similarity Between Cover Image and Marked Image	4
1.2.3 Reversibility	4
1.2.4 Fragility or Robustness	5
1.3 Applications of Reversible Data Hiding	5
1.3.1 Ensuring Integrity	6
1.3.2 Reversible Steganography	7
1.3.3 Reducing Storage and Bandwidth Utilization of Stereo Images	7
1.3.4 Intra-Frame Error Concealment for Improved Video Trans- mission Quality	8
1.3.5 Reversible Adversarial Example	8
1.3.6 Reversible Image Editing	8
1.4 Research Objective	9
1.5 Contribution of the Thesis	10
1.6 Test Images	12

1.7	Organization of the Thesis	13
2	Literature Survey on Reversible Data Hiding	15
2.1	Difference Expansion Based Reversible Data Hiding Techniques .	15
2.2	Histogram Bin Shifting Based Reversible Data Hiding Techniques	19
2.3	Prediction Error Expansion Based Reversible Data Hiding Tech- niques	22
2.3.1	Prediction of a Pixel Value	23
2.3.1.1	Edge and Gradient Based Predictors	23
2.3.1.2	Prediction Using Simple Computation with Neighborhood	25
2.3.1.3	Pixel Value Ordering Based Prediction	29
2.3.1.4	Reference Pixel Based Prediction	30
2.3.1.5	Other Prediction Strategies	31
2.3.2	Embedding Strategies	32
2.4	Other Reversible Data Hiding Techniques	36
2.5	Summary	37
3	Reversible Data Hiding Using B-tree Triangular Decomposition Based Prediction	39
3.1	Prediction Based on B-Tree Triangular Decomposition	40
3.1.1	B-Tree Triangular Decomposition	41
3.1.2	Prediction Scheme	43
3.2	Embedding Using Adaptive Prediction Error Expansion	44
3.2.1	Estimation of Local Complexity	44
3.2.2	Adaptive Embedding Based on Prediction Error and Local Complexity	46
3.2.2.1	Embedding Two Bits in a Pixel	46
3.2.2.2	Embedding Single Bit in a Pixel	47
3.2.3	Order of Embedding	47
3.2.4	Auxiliary Information	48
3.3	Extraction	49
3.3.1	Extraction of Two Bits	50

3.3.2	Extraction of Single Bit	51
3.4	Experimental Results and Discussion	51
3.5	Summary	59
4	Reversible Data Hiding Based on Random Distribution of Reference Pixels	64
4.1	Prediction Based on Random Distribution of Reference Pixels . .	65
4.1.1	Random Distribution of Reference Pixels	66
4.1.2	Prediction Based on Nearby Reference Pixels	68
4.2	Adaptive Prediction Error Expansion	69
4.2.1	Estimation of Local Complexity	71
4.2.2	Order of Embedding	71
4.2.3	Adaptive Embedding Based on Prediction Error and Local Complexity	72
4.3	Extraction	72
4.4	Experimental Results	73
4.5	Summary	82
5	Reversible Data Hiding Using Selected Directional Context Based Prediction	86
5.1	Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood	87
5.1.1	Proposed Selected Context Based Prediction Using Eight Neighborhood	88
5.1.2	Embedding Using Adaptive Histogram Bin Shifting	91
5.1.2.1	Estimation of Local Complexity	93
5.1.2.2	Adaptive Embedding	93
5.1.2.3	Auxiliary Information	94
5.1.3	Extraction	96
5.1.3.1	Extraction of Two Bits	97
5.1.3.2	Extraction of a Single Bit	98
5.1.4	Experimental Results	98
5.2	Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation	102

5.2.1	Extended Gradient Based Selective Weighting (EGBSW) Predictor	105
5.2.2	Proposed Gradient Based Predictor Using 5×5 Neighborhood	107
5.2.3	Embedding and Extraction	110
5.2.4	Experimental Results	110
5.3	Summary	114
6	Reversible Data Hiding Using Multiple Predictors	119
6.1	Performance Comparison of Pixel Predictors	120
6.2	Strategies for Combining Multiple Predicted Values	124
6.3	Embedding and Extraction	127
6.4	Experimental Results	127
6.5	Summary	133
7	Application: Prediction Error Expansion Based Reversible Data Hiding for Detecting Tampered Cheque Image	138
7.1	Brief Introduction of Cheque Truncation System	139
7.2	Prediction Error Expansion Based Reversible Data Hiding Tech- niques for Detecting Cheque Image Tampering	141
7.2.1	Template Based Identification of Important Portions . . .	142
7.2.2	Embedding	143
7.2.3	Extraction	146
7.3	Experimental Results	146
7.4	Summary	148
8	Conclusion and Future Work	152
8.1	Summary of Contributions	152
8.2	Future Research Directions	157
	References	161

List of Figures

1.1	Embedding process of conventional data hiding	2
1.2	Extraction process of conventional data hiding	2
1.3	Conventional data hiding versus reversible data hiding	3
1.4	Standard test images. Top row (left to right): Lena, Lake, Boat, and Mandrill. Bottom row (left to right): Elaine, Airplane, Peppers, and Tiffany	13
2.1	Types of reversible data hiding techniques	16
2.2	Embedding of data bit in difference expansion technique	17
2.3	Extraction of data bit and recovery of original pixel values in difference expansion technique	18
2.4	Context of MED predictor	24
2.5	Contexts of gradient based predictors	26
2.6	Checkerboard pattern of pixel traversal	27
2.7	Inner region and outer region in a sample prediction error histogram	34
3.1	Recursive decomposition of an image into right-angled triangles .	41
3.2	B-tree triangular decomposition of Tiffany image for various values of T_d as (a) 7, (b) 10, (c) 13, and (d) 15, respectively	42
3.3	(a) Original Tiffany image and (b) the predicted Tiffany image using threshold $T_d = 7$	44
3.4	Prediction error histograms of Tiffany image for the proposed prediction scheme with various decomposition thresholds (a) $T_d = 7$, (b) $T_d = 10$, (c) $T_d = 13$, and (d) $T_d = 15$	45

3.5	Prediction error histograms of various predictors. (Row-wise left to right) Top row: Proposed B-tree triangular decomposition based predictor and Delaunay-triangulation based predictor; Middle row: multi-predictor (GMA) and Rhombus-average; Bottom row: SGAP and MED	55
3.6	Embedding capacity versus PSNR plots using various values of local complexity threshold T_{lc} for various images.	61
3.7	Results of the proposed B-tree triangular decomposition based reversible data hiding scheme for various test images (row-wise). Columns from left to right: Original image, marked images with 0.1 bpp, 0.2 bpp, 0.3 bpp, and 0.4 bpp.	62
3.8	Performance comparison among several reversible data hiding techniques for various images	63
4.1	An example of predicting a pixel value by weighted median of reference pixel values in the neighbourhood ($d = 2$)	70
4.2	Original image (Peppers) and its predicted image (number of reference pixels = 25% of total number of pixels in the image, block size= 4×4 , and maximum chess board distance of a pixel in the considered neighbourhood = 3)	70
4.3	Payload size versus PSNR plots for various parameter combinations (percentage of reference pixels and block size) with local complexity threshold $T_{lc} = 0$	76
4.4	Normalized prediction error histograms of various predictors for Peppers image: (a) Proposed predictor using random distribution of reference pixels having 25% of pixels as reference pixels and 4×4 block size, (b) GMA , and (c) B-tree triangular decomposition based prediction with decomposition threshold T_d as 8	77
4.5	Payload size versus PSNR plots for various local complexity threshold values with pre-selected values of percentage of reference pixel and block size	80
4.6	Payload size versus PSNR plots for several reversible data hiding techniques	81

LIST OF FIGURES

4.7	Results of the random distribution of reference pixels based reversible data hiding scheme for various test images (row-wise). Columns from left to right: Original image, marked images with 0.1 bpp, 0.2 bpp, 0.3 bpp, and 0.4 bpp.	85
5.1	8-neighborhood of a pixel at coordinate (m,n)	88
5.2	Context pixel values of 8-neighbors of pixel (m,n) during prediction	91
5.3	Payload size versus PSNR plots using various values of local complexity threshold T_{lc} for various images.	101
5.4	Comparison of PSNR values between the cover images and the marked images with various payload sizes among several reversible data hiding techniques	103
5.5	Results of the selected directional context using 8-neighborhood based reversible data hiding technique for various test images (row-wise). Columns from left to right: Original image, marked images with 5000 bits, 10000 bits, and 15000 bits.	104
5.6	4×4 neighborhood of pixel (m,n) in EGBSW	105
5.7	5×5 neighborhood of pixel (m,n) as being considered in the proposed predictor	107
5.8	Payload size versus PSNR plots using various values of local complexity threshold T_{lc} for various images.	113
5.9	Comparison of PSNR values between the cover images and the marked images with various payload sizes among several reversible data hiding techniques	115
5.10	Performance comparison between the proposed reversible data hiding techniques in Section 5.1 and Section 5.2	116
5.11	Results of the Improved Gradient based reversible data hiding scheme for various test images (row-wise). Columns from left to right: Original image, marked images with 5000 bits, 10000 bits, 15000 bits and 20000 bits	117
6.1	Performances of various combination strategies for the multi-predictor scheme	129

LIST OF FIGURES

6.2	Embedding capacity versus PSNR plots using various values of local complexity threshold T_{lc} for various images.	132
6.3	Results of the proposed multi-predictor scheme based reversible data hiding scheme for various test images (row-wise). Columns from left to right: Original image, marked images with 5000 bits, 10000 bits, 15000 bits and 20000 bits	134
6.4	Comparison of PSNR values between the cover images and the marked images with various payload sizes among several reversible data hiding techniques	135
6.5	Performance comparison of the proposed multi-predictor based reversible data hiding technique with the techniques using selected directional context with 3×3 neighborhood (Section 5.1) and improved gradient based prediction (Section 5.2)	136
7.1	An original cheque image	143
7.2	Template for the cheque image in figure 7.1	144
7.3	An original cheque image	144
7.4	Template for the cheque image in figure 7.3	145
7.5	Original bank cheque images (Cheques numbered 1 to 10 in row-major order)	150
7.6	Marked images for cheque image in figure 7.1 using various reversible data hiding techniques: random distribution of reference pixel based (top-left), eight neighborhood based (top-right), improved gradient based (bottom-left), and multi-predictor based (bottom-right)	151
7.7	Tampered images from the marked images in figure 7.6	151

List of Tables

3.1	PSNR between the cover and the marked images for all test images while hiding 10000 bits with various decomposition threshold (T_d)	53
3.2	Number of reference pixels, non-reference pixels and triangles for the selected values of decomposition threshold T_d	54
3.3	Achievable embedding capacity (in bpp – rounded upto two digits after decimal point) with single embedding ($T_{lc} = 0$)	56
3.4	Achievable embedding capacity (in bpp – rounded upto two digits after decimal point) with adaptive embedding ($T_{lc} = 16$)	57
4.1	Various parameter combinations for reported experiments	74
4.2	Achievable embedding capacity (in bpp – rounded upto two digits after decimal point) with single embedding ($T_{lc} = 0$)	77
4.3	Achievable embedding capacity (in bpp – rounded upto two digits after decimal point) with adaptive embedding ($T_{lc} = 16$)	79
4.4	Minimum, average and maximum of standard deviations of PSNR values over ten executions for each combination of parameter values for various payload sizes	82
5.1	Achievable embedding capacities with histogram bin pairs (-1, 0) and (0, 1) using single bit embedding ($T_{lc} = 0$) for various test images.	99
5.2	Achievable embedding capacities with histogram bin pairs (-1, 0) and (0, 1) using adaptive embedding ($T_{lc} = 16$) for various test images.	99
5.3	Achievable embedding capacities with histogram bin pairs (-1, 0) and (0, 1) using single bit embedding ($T_{lc} = 0$) for various test images.	111

LIST OF TABLES

5.4	Achievable embedding capacities with histogram bin pairs (-1, 0) and (0, 1) using adaptive embedding ($T_{lc} = 16$) for various test images.	112
6.1	Percentage of pixels for which each predictor produces the lowest absolute value of the prediction error	123
6.2	Percentage of pixels for which each predictor produces the lowest absolute value of the prediction error (continuation of table 6.1)	124
6.3	Achievable embedding capacities and corresponding PSNR values with histogram bin pairs (-1, 0) and (0, 1) using single bit embedding ($T_{lc} = 0$) for various test images.	130
6.4	Achievable embedding capacities and corresponding PSNR values with histogram bin pairs (-1, 0) and (0, 1) using adaptive embedding ($T_{lc} = 16$) for various test images.	130
7.1	PSNR values between a cover image and corresponding marked image at maximum achievable embedding capacity for the comparing techniques	148
7.2	Length of secret data string (achievable embedding capacity) for various test images using each comparing techniques	149

Chapter 1

Introduction

Advances in communication, storage and processing capabilities have enabled an increased use of digital media (image, video, audio and text) for personal consumption and business requirements. Some of these use cases require secret communication and storage of data using digital media and/or ensuring integrity of digital media. To ensure the above, data are imperceptibly inserted into the digital media. In general, it can be referred as data hiding [1]. Data hiding is defined as secret embedding of data into a digital media such that the inserted data can be recovered later on a need-to-know basis. Insertion of the data into the cover media is known as embedding process. This process may consider a secret key (optional) to decide the sequence of locations in which the data is embedded. The modified media (due to embedding of data) is termed as marked media. This media is also referred as stego media depending on the purpose of data hiding. During the extraction process, the secret data is extracted from the marked media with the help of the secret key. Use of the secret key during embedding and extraction processes is optional. The embedding and the extraction processes of a data hiding technique are schematically shown in figure 1.1 and 1.2.

There can be two broad categories of usage of data hiding [1, 2]: watermarking and steganography. Watermarking is a process of inserting a watermark (i.e., data) into a cover media [3]. The watermark can be an image, a text, or a binary data. Major applications of watermarking include ownership identification, copyright protection, and ensuring integrity of a media. Embedding of the watermark in a cover media generates a watermarked media. The extraction process

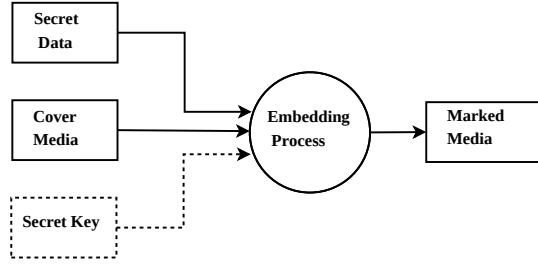


Figure 1.1: Embedding process of conventional data hiding

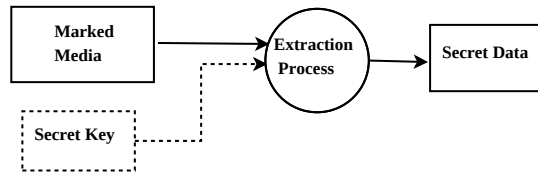


Figure 1.2: Extraction process of conventional data hiding

retrieves the watermark from the watermarked media. A watermarking scheme can be robust against any type of modification of the watermarked media. Hence, in this type of watermarking scheme, the watermark can be extracted in spite of deliberate or accidental modifications in the watermarked media. This is useful for ownership identification and copyright protection. On the contrary, a watermarking scheme can be fragile too. In this type of watermarking scheme, a slight modification of the watermarked media makes the retrieval of watermark impossible. This type of scheme is useful to ensure integrity of the digital media.

Steganography is a process of hiding a secret information into a cover media for the purpose of covert communication or storage [4]. The hidden data can not be detected by unwanted persons. In the parlance of steganography, embedding of the secret data into a cover media generates a stego media. Fundamental requirement of steganography is perceptual indistinguishability between the stego media and the cover media up to an extent that it does not raise suspicion to a mere onlooker.

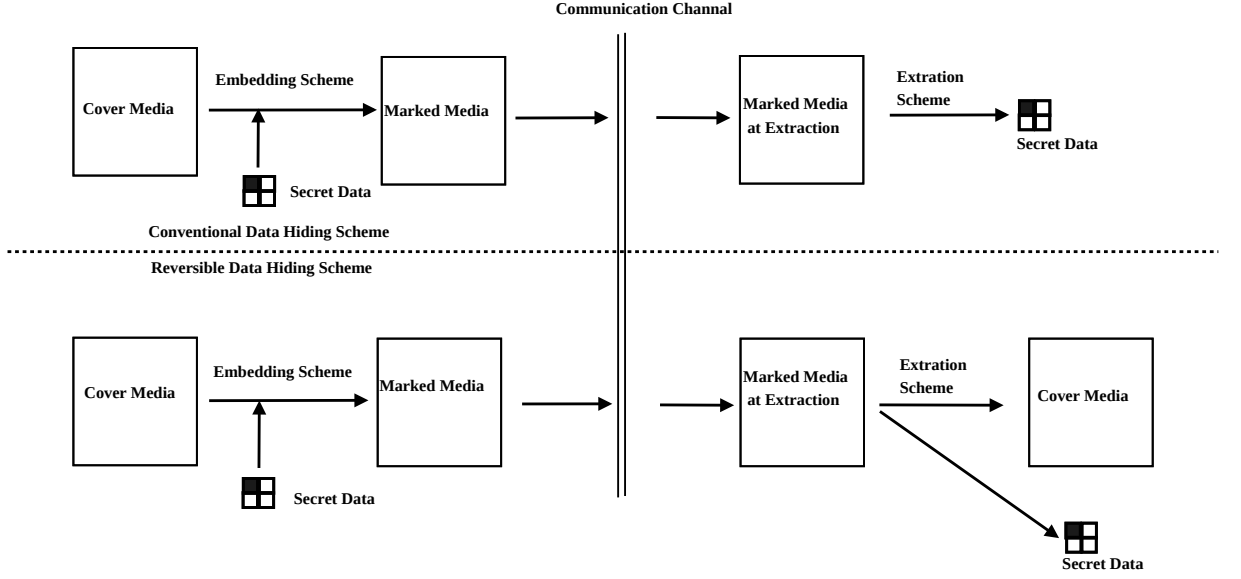


Figure 1.3: Conventional data hiding versus reversible data hiding

1.1 Reversible Data Hiding

The conventional data hiding is known as lossy data hiding because it cannot recover an original cover media from a marked media. On the contrary, in reversible data hiding (RDH) [2], both the inserted data and the cover media can be restored from the marked media. Hence, reversible data hiding is also known as lossless data hiding due to its capability of restoring the cover media from the marked media. This basic difference between conventional data hiding and reversible data hiding is shown in figure 1.3. The works being reported in this thesis consider image as the medium. Hence, subsequent discussions in this thesis deal with reversible data hiding in images.

1.2 Properties of Reversible Data Hiding

Reversible data hiding is a special kind of data hiding. Several properties of this reversible data hiding are stated in this section.

1.2.1 Imperceptibility of Change

Change in an image due to embedding is proportional to the amount of data (number of bits) being embedded in the image, i.e., payload size. Higher payload size introduces larger change in the image. But this change in the image due to embedding should remain imperceptible to our normal eyes. This is a general property of any invisible data hiding technique, like steganography and invisible watermarking.

1.2.2 Similarity Between Cover Image and Marked Image

Change in an image due to embedding of secret data should be as minimum as possible. Similarity between a cover image and a marked image can be estimated using peak signal-to-noise ration (PSNR) between these two images. PSNR between a cover image and a marked image is calculated as:

$$PSNR = 10 \log_{10} \frac{MAX^2}{MSE}$$
$$where \quad MSE = \frac{\sum_{i=1}^M \sum_{j=1}^N (x(i,j) - x'(i,j))^2}{M \times N} \quad (1.1)$$

where $MAX = 255$; $MSE = \text{Mean Square Error}$;
and $M \times N = \text{Size of the cover image}$.

Here, x is the original cover image. x' is the marked image after embedding of secret message. Similarity between these two images is inversely proportional to payload size. A good reversible data hiding technique results in high similarity between these two images. This is also a general property of any invisible data hiding technique like steganography and invisible watermarking.

1.2.3 Reversibility

Main distinguishing characteristic of reversible data hiding is reversibility of a marked image to get back the cover image. Data is inserted into the cover image through a reversible transformation of the cover image pixel values. The reversible transformation ensures restoration of the original cover image from the marked image along with extraction of the hidden data. Then, each pixel value in the

1.3 Applications of Reversible Data Hiding

restored image is same as the value of the corresponding pixel in the cover image. Because of the reversibility, this technique has become useful in several sensitive applications, where recovery of the cover image is also important.

1.2.4 Fragility or Robustness

Due to involvement of reversible integer transforms (as pixel values are integers), several reversible data hiding schemes are fragile in nature. Extraction of hidden data is not possible in case of any modification in the pixel values of a marked image by an adversary. Hence, a fragile reversible data hiding scheme is mainly used for ensuring integrity of an image. If the marked image is not tampered, then hidden data is extracted correctly. Bit error rate, which is a ratio between number of erroneous bits and total number of extracted bits, is zero in this case. Otherwise, if the marked image is tampered, then bit error rate is positive. Bit error rate is mathematically defined as follows:

$$\text{Bit Error Rate} = \frac{\text{Number of Erroneous Bits}}{\text{Total Number of Extracted Bits}} \quad (1.2)$$

In the case of a robust reversible data hiding, the hidden data withstands modifications (or adversarial attacks) on the marked image. Extraction of hidden data is possible in a such a robust scheme in spite of changes in the pixel values in the marked image.

1.3 Applications of Reversible Data Hiding

Traditionally, reversible data hiding is used for ensuring integrity of an image. In this section, it is explained why reversible data hiding is adopted as the primary technique of ensuring integrity of certain types of images. Moreover, several other applications of reversible data hiding have emerged in recent times. Those applications have also been discussed in this section. A good discussion on emerging applications of reversible data hiding can be found in [5].

1.3.1 Ensuring Integrity

A fragile reversible data hiding scheme makes it suitable for ensuring integrity of an image. A secret data is embedded in an image. Extraction of the same data from the marked image (i.e., bit error rate as zero) ensures integrity of the marked image. Moreover, several applications demand restoration of the original image from the marked image. These applications cannot afford even minimum distortion (though imperceptible in our eyes) in the image. Even a small change in the marked image (with respect to the original cover image) may lead to wrong analysis (often automated) in the context of these sensitive applications. Application areas of this type include the following:

1. Medical images carry very sensitive contents about a patient's health. Hospitals and diagnostic centers need to ensure integrity of these images. Moreover, original cover image is needed for automated diagnosis of the health condition. Hence, in contrast to traditional watermarking, reversible data hiding is a suitable choice for ensuring integrity of medical images [6, 7, 8, 9].
2. Remote sensing images also contain important contents for military and other civilian applications. Automated analysis of these images is important for target location and tracking, environment analysis and other geographical studies. Reversible data hiding ensures integrity of remote sensing images while cover images are also restored for automated analysis [10].
3. Image authentication is an important requirement for a court of law to establish the authenticity of an image of a crime scene. Watermarking (one kind of data hiding) of the image helps in achieving this. In contrast to traditional watermarking, reversible data hiding restores the original image (without any distortion), which can be accepted by a court of law. Hence, reversible data hiding can be an useful method for authenticating images for law enforcement [2].
4. Image of a bank cheque is transmitted from one bank to another bank for clearing process. An adversary can easily manipulate the contents of the cheque image to defraud the payer. Reversible data hiding can be used to

1.3 Applications of Reversible Data Hiding

ensure integrity of the cheque image [11, 12] in such a remote cheque clearing environment. Reversible data hiding also restores the original cheque image, which can be automatically analyzed for presence of certain security features in the cheque (such as micro-lettering line, pantograph etc).

5. Geographical information system uses two-dimensional vector data to indicate point coordinates. Reversible data hiding is used to ensure integrity of this two-dimensional vector data [13]. Reversibility property ensures restoration of the original vector data.
6. Two-dimensional vector data is also useful in computer-aided design (CAD) engineering graphics. Hence, reversible data hiding is applied to ensure integrity of this data for CAD engineering graphics [14].

1.3.2 Reversible Steganography

Steganography is a technique for covert communication and storage. Undetectability of the secret data is key contribution of a steganography technique. But a cover image is changed irreversibly in a traditional steganography scheme. Hence, reversible steganography [15, 16] provides a mechanism for getting back the cover image, while maintaining the undetectability of the secret data. Mainly, it is useful for covert storage, where the cover image can be obtained back after erasing the storage data.

1.3.3 Reducing Storage and Bandwidth Utilization of Stereo Images

In [17], a method of efficient storage and transmission of a pair of stereo images is proposed using reversible data hiding. At first, the information to generate the right image from the left image is computed. Then, this information is reversibly embedded in the left image. This reduces storage and bandwidth requirements as only the marked left image is stored and/or transmitted instead of the pair of images. During extraction process, the embedded information can be extracted along with the exact restoration of the left image. Subsequently, the right image can be derived from the left image using this extracted information.

1.3.4 Intra-Frame Error Concealment for Improved Video Transmission Quality

Video transmission is error prone as quality of service is not guaranteed. An intra-frame error concealment algorithm can recover corrupted macro-blocks. Thus, this algorithm helps in improving the quality of video transmission. In [18], a reversible data hiding based approach has been proposed for intra-frame error concealment.

1.3.5 Reversible Adversarial Example

Storing of personal photographs in social media is a common phenomenon now-a-days. Various deep learning models are developed to automatically analyze these images in order to derive useful business information. An adversarial example (i.e., image) can be generated as privacy preserving technique against these deep learning based analysis. Normally, such an adversarial example is generated by adding certain perturbation to the original image. But addition of perturbation may irreversibly change the original image. Hence, a concept of reversible adversarial example has been introduced recently [19]. It hides the perturbation in the generated adversarial image using reversible data hiding. Hence, it is possible to extract the perturbation information and to restore the adversarial image. Subsequently, the original image can be generated by subtracting the perturbation from the adversarial image.

1.3.6 Reversible Image Editing

A traditional image editing method changes an image irreversibly. It is not possible to get back the original image from the edited image, unless a copy of the original image is separately stored. In [20], a reversible image editing method is proposed using reversible data hiding. Similarly, a reversible contrast enhancement method is used in [21] to enhance the contrast of region-of-interests in medical images.

1.4 Research Objective

Since the initial work in reversible data hiding [22] in (2001), this field has been enriched by several important research contributions. Over the years, researchers have attempted (i) to improve embedding capacity of reversible data hiding schemes and simultaneously, (ii) to reduce embedding distortion. A specific genre of these techniques, namely prediction error expansion (PEE) based reversible data hiding technique [23], has been found as better than other genres of techniques. In a prediction error expansion based technique, a pixel value is predicted using the values of a set of context pixels. Data bits are hidden in the expanded prediction error of the concerned pixel. The superiority of PEE-based reversible data hiding techniques has attracted more attention of researchers. Hence, scope of the research work, as reported in this thesis, is also centered around PEE-based reversible data hiding. It is to be noted that the work in this thesis, like several other techniques in the literature, considers 8-bit gray-scale images only.

A good predictor causes less prediction error for a pixel value. As data bits are embedded in the expanded prediction error, a good pixel predictor causes less embedding distortion in the pixel. Hence, objective of the research, as reported in this thesis, is to come up with novel reversible data hiding techniques by exploring suitable pixel prediction schemes. Novel pixel prediction strategies have been proposed in the context of reversible data hiding. Performances of several state-of-the art pixel prediction strategies have also been investigated. The need to combine multiple predictors is also examined in this context. Usage of appropriate pixel prediction strategies has led to a better trade-off between embedding capacity and embedding distortion.

In the context of pixel prediction strategy, the research explores the suitability of the following pixel prediction schemes for reversible data hiding:

- pixel value prediction using a set of reference pixels in the vicinity of the concerned pixel
- prediction of a pixel value by using 8-neighborhood of the pixel

- gradient (as estimated using the pixels in the local neighborhood) based prediction of a pixel value
- multi-predictor scheme to combine the outcomes of multiple predictors

It is to be noted here that the research, as reported in this thesis, focuses only on fragile reversible data hiding techniques.

Finally, proposed PEE-based reversible data hiding techniques have been applied to ensure integrity of a bank cheque image.

1.5 Contribution of the Thesis

Major contributions of the thesis are highlighted in this section. Contributions of the thesis are stated as proposing novel prediction error expansion based reversible data hiding schemes by exploiting novel pixel prediction strategies as following:

1. Prediction of pixel values in an image using a B-tree triangular decomposition based prediction strategy is proposed. The original image is decomposed into a set of right angled triangles. The right angled triangles are generated from the original image by following a B-tree structure. Initially, the root of the tree refers to the original image. The image is divided, at first, into two right-angled triangles with respect to the main diagonal of the image. Later, each right angled triangle is further decomposed recursively into smaller triangles. The triangle division is stored in a B-tree structure. Finally, the pixels, which fall inside or on the three sides of a triangle, are predicted using the pixel values at the vertices of the triangle as the reference pixels using an interpolation technique. A B-tree triangular decomposition based prediction technique was originally proposed for image compression in [24]. A similar strategy with necessary adaptation to the context of reversible data hiding is adopted in this thesis. Finally, an adaptive embedding strategy is used for embedding data into the expanded prediction error.

2. Next, it is further investigated whether a structured method (like the above mentioned B-tree triangular decomposition) can be avoided to decide a set of reference pixels. In this context, a novel mechanism for random distribution of reference pixels is proposed. A set of reference pixels is randomly distributed depending on a local complexity estimation of the non-overlapping blocks of an image. Variance of the pixel values in a block is used to estimate the local complexity of the block. Then, the values of the reference pixels within a certain distance from a non-reference pixel are used for predicting the non-reference pixel value using a weighted median based approach. Finally, an adaptive embedding strategy is used for reversibly hiding the data into an expanded prediction error of a non-reference pixel.

3. A novel prediction scheme based on 8-neighborhood of a pixel is also proposed in this thesis. At first, the differences between pixel values of a pair of neighboring pixels in horizontal, vertical, diagonal and anti-diagonal directions are computed. Similarly, the averages of two neighboring pixel values in each of above four directions are estimated. The computed differences and the computed averages are used to select a set of directions. Then, a novel pixel prediction strategy is proposed by combining linear predictors in the selected directions. Finally, a novel prediction error expansion based reversible data hiding technique is proposed using this selected directional context based prediction. An adaptive prediction error histogram bin shifting strategy is used here to adaptively embed either one bit or two bits of data based on local complexity of the pixel.

4. Next, a gradient based pixel prediction strategy is proposed. This work is inspired by a gradient based reversible data hiding in [25]. But contrary to [25], this work considers a 5×5 neighborhood to estimate gradient. On this front, the prediction is an extension of the gradient based prediction using 4×4 neighborhood in [25] to 5×5 neighborhood. In the proposed work, gradients in horizontal, vertical, diagonal and anti-diagonal directions are computed using a 5×5 neighborhood. Moreover, linear predictors considering a set of neighborhood pixels are considered for each of the above four directions. Then, finally, a weighted average of two predicted values in the

direction of the least two gradients estimates the final predicted value. Finally, a novel prediction error histogram bin shifting based reversible data hiding technique is proposed using this gradient based prediction. The prediction error histogram bin shifting based strategy is extended to adaptively embed either one bit or two bits of data based on local complexity of the pixel.

5. Apart from the proposed predictors as above, there already exist plethora of predictors in the literature. An experimental study is conducted to assess the performance of several of these local context based and gradient based predictors. The experiment reveals that none of the predictors are perfect in predicting the pixel values. But some of these predictors perform better than rest of the others. Then, a multi-predictor based strategy is proposed by considering few of the good predictors. Several schemes for combining predicted values are experimentally evaluated to find out the best way of combining multiple predicted values. Finally, similar to previous two approaches, an adaptive prediction error histogram bin shifting based strategy is adopted to consider variable number of bits for embedding in a pixel.
6. Finally, the proposed reversible data hiding techniques are applied on a set of bank cheque images in order to ensure integrity of the cheque images. Performances of several of the above proposed reversible data hiding techniques are evaluated in the context of bank cheque images.

1.6 Test Images

A set of standard test images (Lena, Lake, Boat, Mandrill, Elaine, Airplane, Peppers, and Tiffany) of size 512×512 is used to evaluate the performances of the proposed reversible data hiding techniques against several existing techniques in the literature. 8-bit gray scale versions of these images are considered in this thesis for reporting the experimental results. These images are presented in figure 1.4.



Figure 1.4: Standard test images. Top row (left to right): Lena, Lake, Boat, and Mandrill. Bottom row (left to right): Elaine, Airplane, Peppers, and Tiffany

1.7 Organization of the Thesis

Rest of the thesis is organized as following:

- A literature survey of existing reversible data hiding techniques is presented in **Chapter 2** .
- In **Chapter 3**, a novel reversible data hiding technique using B-tree triangular decomposition based prediction is proposed.
- In **Chapter 4**, a novel random pixel distribution based prediction and subsequent reversible data hiding technique are proposed.
- In **Chapter 5**, a selected local context based prediction and a gradient based prediction are independently used to develop novel reversible data hiding techniques.
- The need for multi-predictor based techniques is analyzed in **Chapter 6**. Subsequently, several strategies for multi-predictor schemes are experimentally evaluated in **Chapter 6**.

- Application of prediction error expansion based reversible data hiding on bank cheque images is reported in **Chapter 7**.
- Conclusive remarks of this thesis are drawn in **Chapter 8**. Additionally, future research directions are highlighted in this chapter.

Chapter 2

Literature Survey on Reversible Data Hiding

A survey of existing reversible data hiding (RDH) techniques is presented in this chapter. These techniques can be divided mainly into three major categories: (1) difference expansion (DE), (2) histogram bin shifting (HBS) and (3) prediction error expansion (PEE). A pictorial representation of this categorization is given in figure 2.1. Subsequent sections discuss existing techniques in each of these categories. Survey on reversible data hiding techniques can also be found in [2, 26].

2.1 Difference Expansion Based Reversible Data Hiding Techniques

Difference expansion (DE) method for reversible data hiding was first introduced in [27]. Subsequently, a detailed description of this method is published in [28]. In this technique, one bit data is embedded into a pair of neighboring pixels in the cover image by expanding the difference between the neighboring pixel values. The key idea of this technique is demonstrated in figures 2.2 and 2.3 through an example of embedding and extraction phases, respectively. Let a pair of adjacent neighboring pixel values be $(x, y) = (206, 201)$. Difference between this pair of pixel values is computed as $d = x - y = 206 - 201 = 5$. Additionally, an integer

2.1 Difference Expansion Based Reversible Data Hiding Techniques

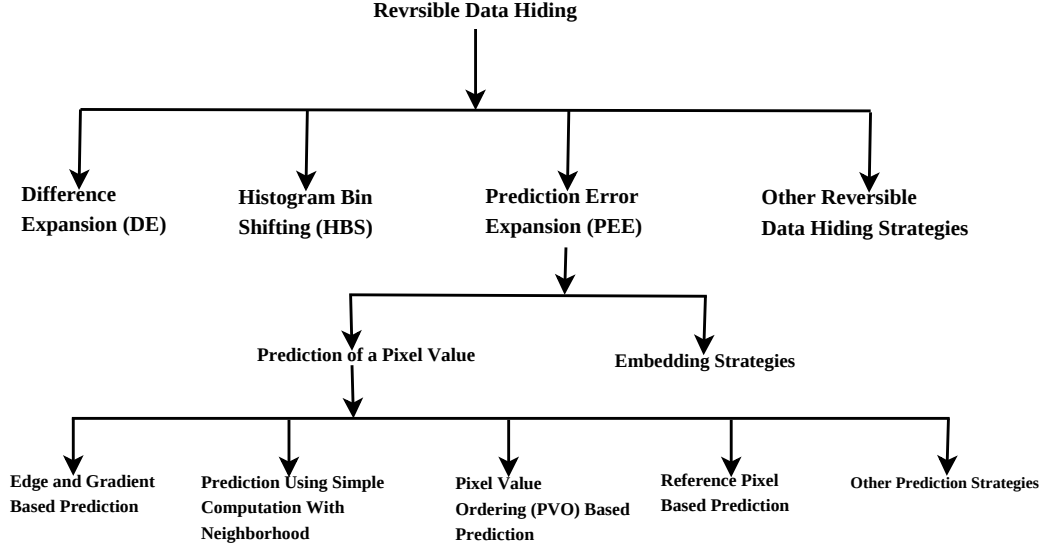


Figure 2.1: Types of reversible data hiding techniques

average a of the pair of pixel values is computed as $\lfloor \frac{x+y}{2} \rfloor = \lfloor \frac{206+201}{2} \rfloor = 203$. $\lfloor \cdot \rfloor$ indicates the largest integer which is smaller than its argument. The binary form of the difference d is 101. Let the data bit be 1. The data bit is appended at the least significant bit (LSB) of the expanded difference (i.e., left-shifted difference bit string) as $d'=1011$. Then, marked pixel values are obtained as $x' = a + \lfloor \frac{d'+1}{2} \rfloor$ and $y' = a - \lfloor \frac{d'}{2} \rfloor$. Hence, the pair of marked pixel values are $(x', y') = (209, 198)$. From the marked pixel values, the embedded data bit is extracted and original pixel values are computed as following: The difference between the pair of marked pixel values is computed as $(d' = 209 - 198 = 11)$. The LSB of this difference value (in binary representation) is 1. Therefore, the embedded data bit is correctly extracted. The original difference between the cover image pixel values is obtained as $d = \lfloor \frac{d'}{2} \rfloor = 5$. The integer average of the marked pixel values a is computed as $a = \lfloor \frac{x'+y'}{2} \rfloor = \lfloor \frac{209+198}{2} \rfloor = 203$. It is to be noted that the integer averages of the original pixel values and the marked pixel values are same. From the difference $(d = 5)$ and the integer average of original pixel values $(a = 203)$, the original cover image pixel values are recovered as $(x, y) = (a + \lfloor \frac{d+1}{2} \rfloor, a - \lfloor \frac{d}{2} \rfloor) = (206, 201)$. It is to be noted that marked pixel values (x', y') may not be within the range $[0, 255]$ due to above difference expansion technique. If the marked pixel value

2.1 Difference Expansion Based Reversible Data Hiding Techniques

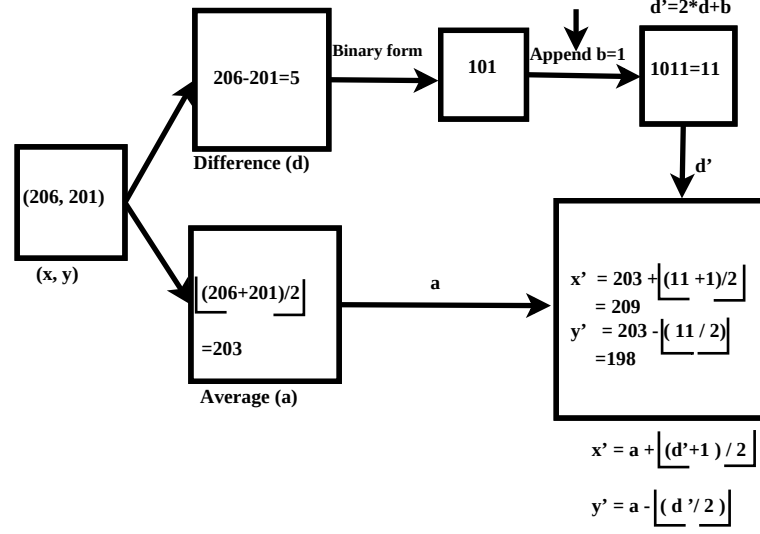


Figure 2.2: Embedding of data bit in difference expansion technique

exceeds 255, the condition is known as overflow. On the other hand, if the marked pixel value is less than 0, the condition is termed as underflow. Both of these conditions are prohibitive, as a pixel value must be in the range $[0, 255]$. The technique in [28] identifies a set of difference values, where overflow/underflow does not arise. Data bits are embedded only in those difference values to avoid overflow/underflow.

Different approaches have been evolved from the basic DE based technique. An improved version of the DE based technique is presented in [29], where few more difference values are also considered for embedding. This is achieved by shifting these difference values such that overflow/underflow will not arise in the shifted difference values. Then, embedding of data bits is also carried out in the shifted difference values. Thus, this technique in [29] achieves higher embedding capacity than the technique in [28]. Difference expansion based reversible data hiding using triplet of values has been explored in [30]. Similar integer transforms like those in [28] are used in this work with necessary amendments due to the usage of a triplet of values. Differences of two of these three values from the third value are expanded in this technique. Differences within a triplet of values are expanded for data embedding in [31] too. The technique in [32] generalizes these transformations for a vector of any length as compared to those for triplet

2.1 Difference Expansion Based Reversible Data Hiding Techniques

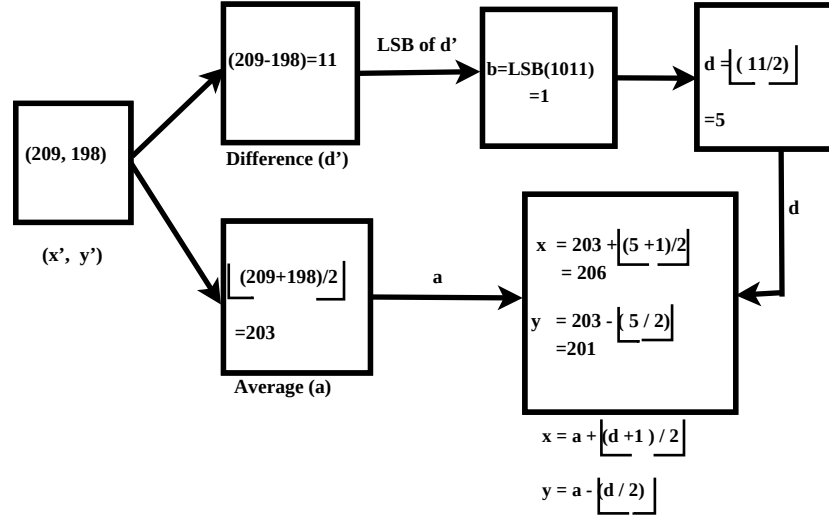


Figure 2.3: Extraction of data bit and recovery of original pixel values in difference expansion technique

of values in [30]. Further generalization of these transformations can be found in [33]. Similarly, in [34, 35], data bits are embedded by expanding the differences in pixel values in a quad of adjacent pixels. In [36], reversible data hiding is applied in medical images, where the image is divided into smooth and non-smooth regions. In this technique, either the original DE-based approach using pair of pixels [28] or the approach using quad of pixels [34] is used to embed data in non-smooth regions. Similarly, another adaptive embedding is proposed in [37], where amount of embedding in a block is decided using the smoothness of pixel values in the block.

A common problem with these DE-based techniques is the occurrence of overflow/underflow. Overflow/underflow condition arises if the changed pixel value after embedding exceeds the range $[0, 255]$. A location map is used to specify these locations where embedding should not be carried out to avoid the above condition. This location map needs to be embedded in the image. The method in [38] suggests a technique to reduce the overflow/underflow location map. Moreover, an intelligent pairing of pixels in the DE-based approach completely eliminates the need for an overflow/underflow location map [12]. In this technique, pixel pairs are chosen in such a way that overflow/underflow condition does not arise.

2.2 Histogram Bin Shifting Based Reversible Data Hiding Techniques

In another enhancement of DE-based techniques, a two-dimensional difference expansion technique is proposed for embedding data bits in [39]. The possibility of utilizing image characteristics such as standard deviation, smoothness, and uniformity as a guide to choose a threshold with a block-based 2-dimensional difference expansion scheme is proposed in [40].

2.2 Histogram Bin Shifting Based Reversible Data Hiding Techniques

Reversible data hiding based on histogram bin shifting method was first introduced in [41]. A histogram of the grayscale values of the original cover image is generated. A pair of peak and zero points in the histogram are identified. A peak point indicates a histogram bin whose height is the highest among all bins in the histogram. A zero point indicates a histogram bin whose height is zero (i.e., any pixel does not have the concerned gray value). Let the peak point and zero point in the histogram are denoted by the bins having gray values h_p and h_z , respectively. Here, two cases may arise. (1) If the zero point is at the right side of the peak point (i.e., $h_p < h_z$), then the histogram bins for $h_p + 1$ to h_z are shifted at their right side by one position. This means that the gray values in the range $[h_p + 1, h_z]$ shift to the range $[h_p + 2, h_z + 1]$. Each gray value in this range is increased by 1. It creates an empty bin for the gray value $h_p + 1$. Then, one bit data is embedded into a pixel which is associated with peak point, i.e., histogram bin for gray value h_p . If the data bit is 0, the pixel value is not changed. If the data bit is 1, the pixel value is increased by 1. It moves the pixel from the bin having gray value h_p to the empty bin having gray value $h_p + 1$. (2) If the zero point is at the left side of the peak point (i.e., $h_z < h_p$), then the histogram bins for $h_p - 1$ to h_z are shifted at their left side by one position. This means that the gray values in the range $[h_z, h_p - 1]$ shift to the range $[h_z - 1, h_p - 2]$. Each gray value in this range is decreased by 1. It creates an empty bin for the gray value $h_p - 1$. Then, one bit data is embedded into a pixel which is associated with peak point, i.e., histogram bin for gray value h_p . If the data bit is 0, the pixel value is not changed. If the data bit is 1, the pixel value is decreased by 1. It moves

2.2 Histogram Bin Shifting Based Reversible Data Hiding Techniques

the pixel from the bin having gray value h_p to the empty bin having gray value $h_p - 1$. Basically, histogram bin shifting method creates an empty bin adjacent to the peak point by shifting all other bins till a zero point is encountered. Then, one bit data is embedded into the pixels corresponding to the peak bin. If the data bit is 0, a pixel remains in the same bin. If the data bit is 1, a pixel moves to the adjacent empty bin.

In [41], a data bit as 1 causes modification in the pixel. On the contrary, a data bit as 0 does not modify the pixel value. Hence, more number of 1 in the data bit string as compared to number of 0 causes more distortion than a case where number of 0 is more than number of 1 in the bit string. The histogram bin shifting based reversible data hiding technique in [41] is enhanced in the work in [42] by complementing the data bit string if number of 1 is more than number of 0 in the bit string. Thus, the technique in [42] reduces the embedding distortion as compared to the basic histogram bin shifting technique in [41]. Moreover, a block-based complement strategy is introduced to reduce the distortion further. For each block in the data bit string, count of 1 and 0 bits decide whether complement of the block is to be considered [42].

One major artifact of traditional histogram bin shifting based reversible data hiding technique is that the peak bin in the histogram is completely split into two bins due to embedding of data bits into the pixels in the peak bin. Hence, identity of the original peak bin is lost in the histogram of the gray values for the marked image. Then, the information about the original peak bin is separately passed to the extraction process. Therefore, data bits are not embedded in the pixels corresponding to the peak bin in the histogram bin shifting based technique in [43]. Rather, two neighboring bins of the peak bin are used to embed the data bits. The peak bin is treated as the reference point for data embedding. Therefore, identification of the same peak bin is possible from the marked image during extraction process. It guides to the neighboring bins where embedding has taken place. Additionally, this technique in [43] applies the above mentioned histogram bin shifting based embedding for each non-overlapping square block in the image separately. Moreover, a multi-layer embedding is suggested to embed more data. According to this multi-layer strategy, the marked image is again considered as the cover image for next layer of embedding in order to embed more data bits.

2.2 Histogram Bin Shifting Based Reversible Data Hiding Techniques

Another version of histogram bin shifting based reversible data hiding technique is proposed in [44]. Here, a histogram bin is selected for embedding data bits such that the number of pixels being shifted (i.e., number of pixels in the bins in between the selected bin and a zero bin) are minimum.

In the above histogram bin shifting based techniques, the histogram of pixel values in the image/block is constructed. In a completely different approach of histogram bin shifting [45, 46], the histogram is constructed for the differences in neighboring pixel values. In [46], two differences are computed by considering how different a pixel value is from its top and left neighboring pixels. Then, the peak bin of the histogram is used to insert the secret data and all other bins are shifted until a zero bin is encountered. Similarly, in [47], a histogram of differences in pixel values in a block is constructed. The histogram bin shifting method uses this difference histogram of a block for embedding data bits. This technique considers one pixel in the block as a reference pixel. The differences in pixel values are computed with respect to the value of the reference pixel. In another histogram bin shifting based reversible data hiding technique [48], modification to the histogram of absolute differences between neighboring pixel values is proposed. Moreover, the peak bin is not used for embedding. Like the technique in [43], adjacent bins of the peak bin are used for embedding data bits. Multiple peak bins can be used for embedding more data. Binary tree structure is used for properly communicating the peak bins to the extraction process.

In another histogram bin shifting based reversible data hiding technique [49], few higher significant bit (HSB) planes are considered instead of the complete pixel value. The binary representation of the pixel value in an original cover image is divided into two parts based on bit significance: the HSB and the least significant bit (LSB) parts. The higher significant bit difference is calculated from the HSB values of adjacent pixels in the cover image. Data is embedded in the pixels causing the peak points in the histogram of higher significant bit differences. Another histogram bin shifting based reversible data hiding technique using a bit-plane slicing method is proposed in [50].

A reversible data hiding technique based on a block-based histogram bin shifting is proposed in [51]. An image is partitioned into non-overlapping square blocks. An average of pair-wise differences of adjacent pixel values is estimated

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

for each block. This estimated value controls the amount of shifting for the histogram bins of the block. Another block based histogram bin shifting method is proposed in [52], where the histogram bin corresponding to a either minimum or maximum pixel value of a block is considered as a reference bin. It is a two-pass embedding approach. The minimum and the maximum pixel values in a block are used as reference in the first and the second passes, respectively.

A generalized framework for histogram shifting based reversible data hiding is proposed in [53].

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

In difference expansion based techniques (Section 2.1), one data bit is embedded in the expanded difference of two pixel values. These pixels may be neighbors of each other or they may be part of same square block. The idea of selecting two such pixels is that neighboring pixel values (or the values of pixels in a close vicinity of each other in a block) may have high correlation. As a result, the difference between these pixel values may be small. Obtaining a small difference value is good for these reversible data hiding techniques, as the difference is expanded due to embedding. A small difference value causes less embedding distortion in the marked image. One natural extension of this difference expansion based technique is prediction error expansion (PEE) based technique. Prediction error expansion based technique was first introduced in [54]. Later, a detailed presentation of this technique is found in [23]. In a PEE-based technique, a pixel value is predicted by exploiting its correlation with neighboring pixel values. The prediction error is estimated by subtracting the predicted value from the original pixel value. Then, one data bit is embedded in expanded prediction error of the pixel. Thus, prediction error plays a significant role to reduce the distortion of the marked image. Therefore, a PEE-based technique requires a good predictor which can exploit the correlation among a set of neighboring pixels.

Let an original cover image pixel be x . Let the predicted value of the pixel be x' . Then, the prediction error (PE) is the difference between original and

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

predicted values, i.e., $PE = x - x'$. Then, one bit data w is embedded into the expanded prediction error to generate a marked pixel value x^w using equation 2.1.

$$\begin{aligned} x^w &= x' + 2 \times (x - x') + w \\ &= x' + 2 \times x - 2 \times x' + w \\ &= 2 \times x - x' + w \\ &= x + x - x' + w \\ &= x + PE + w \end{aligned} \tag{2.1}$$

The PEE-based reversible data hiding techniques, in general, exhibit better performance as compared to traditional difference expansion based techniques. Since the introduction of the first technique [54] in this category, a lot of researches are being carried out in this direction. As the focus of this thesis is on PEE-based reversible data hiding, more emphasis is given in the following subsections on presenting the literature of this genre of techniques. It can be seen from the initial discussion on this technique that two major steps of this technique are: (1) prediction of pixel value and (2) embedding of data bits in the prediction error.

2.3.1 Prediction of a Pixel Value

Key to success of a prediction error expansion based technique is usage of a good predictor to predict a pixel value. Researchers have come up with various pixel prediction schemes. A compilation of these pixel prediction schemes is presented in this subsection.

2.3.1.1 Edge and Gradient Based Predictors

Initially, prediction strategies are heavily influenced by the researches in image coding. For example, a Median Edge Detector (MED) [55], as is adopted by the PEE-based reversible data hiding in [23, 54, 56], was initially proposed for a lossless image coding algorithm. According to MED predictor, a pixel value is predicted using the top, the left, and the top-left neighboring pixel values depending on a presence of edge at immediate left or at immediate top of it.

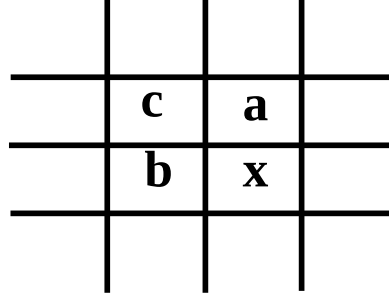


Figure 2.4: Context of MED predictor

Mathematically, MED predictor is defined based on the presented neighborhood in figure 2.4 using equation 2.2.

$$x' = \begin{cases} \max(a, b) & \text{if } c \leq \min(a, b) \\ \min(a, b) & \text{if } c \geq \min(a, b) \\ a + b - c & \text{Otherwise} \end{cases} \quad (2.2)$$

On the contrary, the right, the bottom and the bottom-right neighbors of the current pixel are involved in Median Edge Detector (MED) predictor in [57, 58, 59]. In both variations of MED predictor, three neighboring pixels are explored for predicting the pixel value.

Gradient Adjusted Predictor (GAP) uses vertical and horizontal gradients for predicting the current pixel [58, 60]. Initially, GAP was also proposed for lossless image coding in [61]. The predicted value of a pixel depends on the difference between strengths of the vertical and the horizontal gradients. Moreover, the computation of gradient is simplified in a Simplified Gradient Adjusted Predictor (SGAP) [58].

Unlike the gradient estimations in two directions, several predictors estimate gradients in four directions. For example, in Gradient Based Selective Weighting (GBSW) predictor [62], gradients are computed in horizontal, vertical, diagonal, and anti-diagonal directions. Two directions having the least two gradient magnitudes are considered for predicting a pixel value. The predicted pixel value is estimated as a weighted average of the causal pixel values in the selected directions. In Extended Gradient Based Selective Weighting (EGBSW) [25] predictor, the above concept of gradient estimations are extended using a larger set of neighborhood pixels than the neighborhood pixels in GBSW predictor. Another key

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

difference with GBSW predictor is that EGBSW predictor uses linear predictor in each direction instead of just a causal pixel value. Linear predictor in each direction considers the average of two neighboring pixel values in the same direction. The final predicted pixel value is obtained as a weighted average of two linearly predicted values in the directions having the least two gradient magnitudes.

Threshold-controlled Gradient Adaptive Planar Prediction (TGAPP) [63] combines the simple gradient estimation (in horizontal and vertical direction) and MED predictor. Unlike the use of an ordering of neighboring pixel values in original MED predictor [55], TGAPP uses the difference between two gradient magnitudes to decide the final predicted value.

Another approach of gradient based prediction of a pixel value can be found in [64]. At first, values of 4-diagonal neighboring pixels are predicted using a simple average of the values of 4-neighbors of each of the diagonal neighboring pixel. Then, these predicted values are used to estimate gradients in horizontal and vertical directions. Use of predicted values of neighboring pixels to estimate the gradient differentiates this scheme from previous other schemes of gradient estimation. Finally, a comparison between the gradient magnitudes in horizontal and vertical directions decides the final predicted value as the average of values of either two vertical neighbors, two horizontal neighbors, or 4-neighbors of the pixel.

Several varied approaches for gradient estimations can be found in above gradient based predictors. The variation in considered neighborhood pixels for gradient estimation is presented in figure 2.5. Here, x indicates the pixel whose value is predicted.

2.3.1.2 Prediction Using Simple Computation with Neighborhood

Several other predictors rely on a simple computation using neighboring pixel values. For example, a pixel value is predicted as an average of the values of its left and right neighbors [65]. Average of the values at the top and the left neighbors of a pixel is computed to predict a pixel value in [66, 67]. Similarly, a pixel value is predicted using an average of top, left and top-right neighboring pixel values in [68]. An average of four neighboring pixel values (two horizontal

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

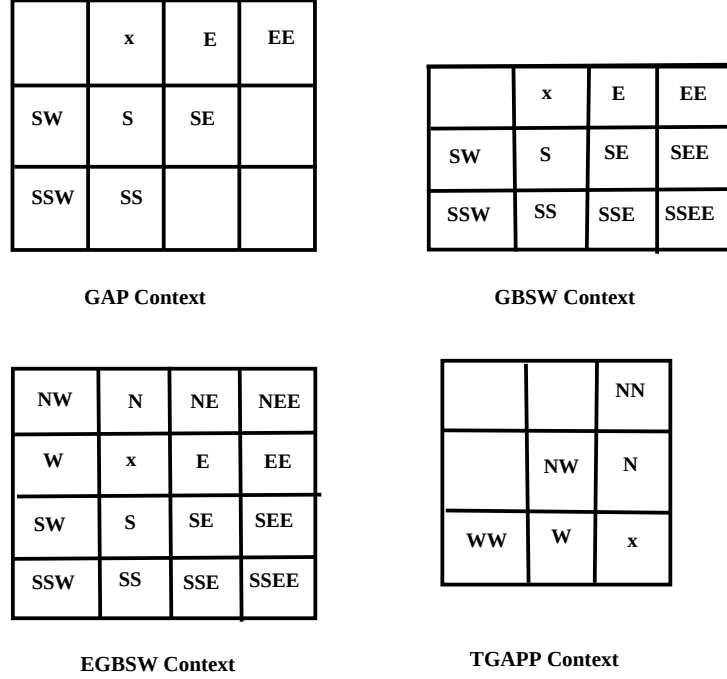


Figure 2.5: Contexts of gradient based predictors

and two vertical neighbors) can also be used to predict the center pixel value in the rhombus predictor [69, 70, 71, 72, 73]. In multi-layer scheme [71], the rhombus predictor is applied in every alternate pixel in the first round and in remaining pixels in the second round. An improved rhombus predictor in [74] uses an average of pixel values in either two horizontal neighbors, two vertical neighbors, or 4-neighbors depending on the homogeneity of neighboring pixel values. In [75, 76], an average of pixel values in either two horizontal neighbors or two vertical neighbors is considered as the predicted value. In [77], a pixel value is predicted by considering the averages of pair of neighboring pixel values in a few selected directions. In [78], a pair of predicted values are used for the reversible data hiding scheme. Pixel values at 4-neighbors of the current pixel are sorted in ascending order. Then, three different predictors are proposed in [78]. In the first predictor, the first and the last pixel values in the sorted order are used as two predicted values. In the second predictor, average of the first two and average of the last two pixel values are used as two predicted values. Finally, in the third predictor, average of the first three and the last three pixel values are

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

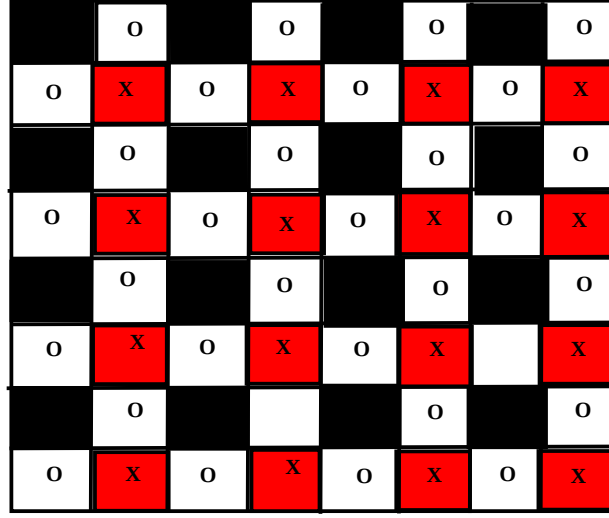


Figure 2.6: Checkerboard pattern of pixel traversal

considered as two predicted values.

A weighted average of four neighboring pixel values predict the center pixel value in [79]. The weight associated with a neighboring pixel is inversely proportional to the absolute difference between the simple average and the neighboring pixel value. In [80], at first, average of two horizontal neighbors and average of two vertical neighbors are computed. Then, a weighted average of above two averages predicts the center pixel value.

In [81], the pixels are traversed in checkerboard pattern, which is given in figure 2.6. The prediction of pixel values are carried out in two passes. In the first pass, the pixels as denoted by 'x' are predicted using average of pixel values in either two diagonal neighbors, two anti-diagonal neighbors, or four diagonal neighbors. Similarly, in the second pass, the pixels as denoted by 'o' are predicted using average of pixel values in either two horizontal neighbors, two vertical neighbors, or 4-neighbors.

In [82], a weighted average of values in six neighboring pixels (top-left, top, top-right, bottom-left, bottom, and bottom-right neighbors) predicts the center pixel value. More weights are assigned to the top and the bottom neighbors. In this scheme, the pixels at the same row as the current pixel (i.e., left and right neighbors) are not considered for predicting the current pixel. Another weighted

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

average scheme using similar context of neighborhood pixels is also considered in [83].

In a completely different approach in [84], a pixel value is, at first, predicted using an average of the values in its 4-neighbors. Subsequently, the predicted value is iteratively modified using differences of the pixel values from neighboring pixel values.

In all above prediction schemes, the prediction context is taken from a fixed block size surrounding every pixel in the image. Interestingly, the prediction scheme in [85] proposes to select the block size for each pixel based on its context characteristics.

A reversible data hiding technique based on a least square (LS) predictor is proposed in [86]. It determines the weights for the neighboring pixels such that it minimizes the sum of squares of prediction errors. According to the technique in [86], a distinct LS predictor is obtained for each pixel. Alternatively, a distinct LS predictor is obtained for each block in [87]. These least square predictors are evaluated with several block sizes to reduce the prediction error. Similarly, LS based predictors are also used in [88, 89]. In another interesting LS based approach [90], enhanced pixel values in the neighborhood are considered for predicting a pixel value.

Instead of the average based computation of the prediction value, a weighted median of a set of pixel values is considered as the predicted value in [91]. Weight being associated to a pixel value is inversely proportional to the distance of the pixel from the current pixel.

Moreover, bit-wise logic operation (using AND and OR logic) involving binary representations of neighboring pixel values is proposed for predicting a pixel value in [92].

Few other approaches consider various interpolation techniques for predicting a pixel value using a set of neighboring pixels. For example, neighbor mean interpolation (NMI) [93], interpolation by neighboring pixels (INP) [94] and several other interpolation schemes [95, 96, 97, 98, 99] are used for pixel value prediction in the context of reversible data hiding. Interestingly, genetic algorithm is used in [100] to find out optimum parameters for interpolation. Moreover, performance of particle swarm optimization is also tested in this context [100].

2.3.1.3 Pixel Value Ordering Based Prediction

Pixel Value Ordering (PVO) based technique [101] adds another direction to the research in reversible data hiding techniques. The image is divided into non-overlapping blocks of similar size. The pixel values in a block are arranged in ascending order. If the values are same for two pixels, then their relative locations decide the ordering between these pixels. This ordering of values plays a vital role in the PVO based reversible data hiding technique. The second-left and the second-right pixel values in the ordered sequence are used for predicting the pixel values for the left-most and the right-most pixels, respectively, in a block. This PVO-based technique in [101] cannot handle the case when the largest value (or the smallest value) appears in more than one pixel.

Hence, a generalization of the above scheme is suggested in [102]. This technique can operate even if multiple pixels contain the largest (or the smallest) pixel value in the block. Number of pixels having the largest (or the smallest) value in the block varies from one block to another. In this technique, the second largest (or the second smallest) pixel value is used to predict the values of the pixels at even multiple locations, where the largest (or the smallest) value appears.

As an extension of the original PVO-based technique [101], a multi-pass PVO based approach is proposed in [103]. Here, the k -largest and the k -smallest pixels are considered for k -pass embedding.

Unlike the fixed-sized blocks in the previous techniques, a dynamic block partitioning is proposed in [104]. Here, sizes of each block are not same. Moreover, a quad tree decomposition is adopted in [105] for dynamic block partitioning. An image is recursively divided into several blocks of dissimilar sizes based on redundancy of pixel values in a block.

In another variation of the above techniques, the constraint of fixed block partitions of PVO-based approaches is removed in Pixel-based PVO (PPVO) prediction [106]. In PPVO, a context surrounding each pixel is considered as the block. The prediction uses a sorted ordering of these context pixels. In [107], the context pixels are selected in such a way that the two boundary lines of the current pixel and context pixels form an obtuse angle. PPVO-based prediction is

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

applied using such a context. Variations of the PPVO scheme are also found in [108, 109].

In another extension of PVO-based techniques, namely Pixel Value Grouping (PVG) [110], the sorted pixel values are categorized into two groups based on similarity of their values with the largest and the smallest values in a block. The predictions are carried out within these groups.

In another variation of PVO-based techniques, a pairwise pixel value ordering technique is proposed in [111].

Recently, a pixel-based pixel value grouping method in [112] exploits the advantages of both pixel-based approach of PPVO technique and Pixel Value Grouping (PVG) technique.

Several other variations of PVO-based techniques can be found in [113, 114, 115, 116, 117].

2.3.1.4 Reference Pixel Based Prediction

This section discusses existing reference pixels based prediction strategies for reversible data hiding. Initially, a set of reference pixels is distributed throughout an image. Then, the values of these reference pixels are used for predicting the values of non-reference pixels.

In the reversible data hiding technique in [118], the reference pixels are distributed based on a local complexity estimation of the pixels in the image. Difference between the maximum value and the minimum value in the neighborhood of a pixel is considered as local complexity estimation. A pixel is either considered as a reference pixel or a non-reference pixel depending on its local complexity. Then, interpolation using the values of reference pixels is used for predicting the non-reference pixel values.

In Delaunay triangulation based reversible data hiding technique [119], a concept of Delaunay triangulation is used to obtain a set of reference pixels. In a Delaunay triangulation mesh, the minimum angle of each triangle in the mesh is maximized. Basically, the triangles are formed in such a way that the circumcircle of any triangle in the mesh does not include any other triangle vertex. The vertices of the triangles in such a Delaunay mesh are considered as the reference

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

pixels. The non-reference pixel values are predicted using an interpolation of the reference pixel values.

In [120], a low resolution image of size $\frac{M}{2} \times \frac{N}{2}$ is obtained by direct down-sampling of an image of size $M \times N$. Then, pixels in this low resolution image (or rather, their corresponding pixels in the input image) are used as reference pixels to predict other pixels. The predictions of non-reference pixel values are carried out at two stages. At first stage, prediction of some of the non-reference pixel values is carried out using values of four surrounding reference pixels. These four reference pixels are put into two groups, where the groups of pixels form orthogonal directions with respect to each other. The lines joining these pairs of diagonal pixels are orthogonal to each other. Initially, the estimated value in each direction is computed by taking the mean of the pair of pixel values. A weighted sum of these two estimated values is the final predicted value of the non-reference pixel. At the second stage, values of rest of the non-reference pixels are predicted using four pixel values using a similar technique as above. But, in the second stage, the four surrounding pixels also include some of the already predicted pixels of the first stage along with the reference pixels.

Similarly, a reversible data hiding technique in [121], a set of reference pixels is considered as pixels at every alternate rows and columns. In a two-stage procedure as similar to [120], values of few of the non-reference pixels are interpolated, at first, using four reference pixel values. Then, values of the remaining non-reference pixels are interpolated using two already interpolated values and two reference pixel values.

2.3.1.5 Other Prediction Strategies

Few other prediction strategies can also be found in the context of prediction error expansion based reversible data hiding.

Several neural network based pixel value predictors can be found in the literature of prediction error expansion based reversible data hiding [122, 123]. In [122], an input image is down-sampled. The pixel values at the down-sampled image (or rather, their corresponding pixel values at the input image) are used to predict other pixel values. An extreme learning machine is utilized in [122]

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

for predicting the pixel values using a regression framework. Similarly, extreme learning machine based pixel value prediction is also found in [124]. Moreover, a convolutional neural network is used to predict pixel values in a pair of stereo images [123].

A different pixel prediction mechanism based on graph signal processing (GSP) is recently proposed in [125]. The prediction of a pixel is designed as a graph signal restoration problem. A minimum rate prediction strategy can be found in [126].

Moreover, several reversible data hiding techniques combine multiple predictors to obtain a single predicted value [127, 128]. These techniques vary in how multiple predictors are combined. In [127], multiple predicted values are sorted. If all predicted values are same, then this value is considered as the final predicted value. If a pixel value is less than or equal to the smallest of these predicted values, then the final predicted value for the pixel is the smallest of those multiple predicted values. On the other hand, if a pixel value is greater than or equal to the highest of those predicted values, then the final predicted value for the pixel is the highest of those predicted values. Final predicted value is not decided if a pixel value lies in between the smallest and the highest of the predicted values. Embedding of data bit is not carried out in such a pixel. Similarly, multiple predictor based strategy in [128] suggests a two-phase embedding. In the first phase, the maximum of multiple predicted values is selected as the predicted value. In the second phase of embedding, the minimum of all predicted values is considered as the predicted value.

In another interesting work, unlike the prediction of a pixel value in all above techniques, prediction error is predicted in [129] using a set of prediction errors in neighboring pixels.

2.3.2 Embedding Strategies

Several embedding strategies are used for producing a marked image in prediction error expansion (PEE) based reversible data hiding techniques. This section discusses these embedding strategies. Taking the cue from difference expansion (DE) based reversible data hiding technique [28], a PEE based technique expands the prediction error (i.e., difference between original pixel value and its predicted

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

value) of a pixel by multiplying it with 2. Then, one bit of data is embedded in the expanded prediction error [23] (equation 2.1). A small prediction error leads to low embedding distortion. Hence, a threshold on the prediction error is used to select the pixels, where embedding of data bit may be carried out. Expansion of the prediction error due to embedding modifies the histogram of the prediction error. The above mentioned threshold partitions the prediction error histogram into two regions, namely inner region and outer region. These two regions in a prediction error histogram are depicted in figure 2.7. The prediction errors in the inner region are used for embedding data bits. As a result, inner region is expanded. Then, the bins in the outer region of the prediction error histogram are shifted away from the zero-th bin. Shifting of these bins is carried out to avoid overlapping between histogram bins of expanded inner region and outer region. Histogram bins of inner region and outer region have to be maintained separately for perfect recovery of the hidden data bits and for restoration of original pixel values. The threshold value is dynamically selected to ensure that entire bit string is embedded. This simple PEE-based embedding strategy is also used in several reversible data hiding techniques in [25, 57, 59, 67, 68, 70, 74, 84, 86, 91]. Additionally, in reversible data hiding technique in [69], pixels are traversed for embedding by following an ascending order of a local complexity values of the pixels. Local complexity of a pixel captures difference among the values of 4-neighboring pixels. Hence, embedding is carried out in the pixels in a smooth region of an image. The simple prediction error histogram modification technique as above is used for this technique too.

In the above PEE based embedding strategies, the amount of modification in a pixel value due to embedding is $PE + w$ (equation 2.1). Here, PE is the prediction error in a pixel and w is one bit data. Alternatively, a fractional part of the amount of modification (i.e., $PE + w$) is added in a pixel where embedding is carried out [58, 130, 131]. Remaining part of the modification is added in its neighboring pixel values. As a result, the embedding distortion in a single pixel is shared with its neighboring pixels.

Instead of single bit embedding in a pixel, an adaptive embedding strategy is proposed in [132] to increase the embedding capacity of a reversible data hiding technique. Either one bit or two bits of data are embedded in a pixel based on

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

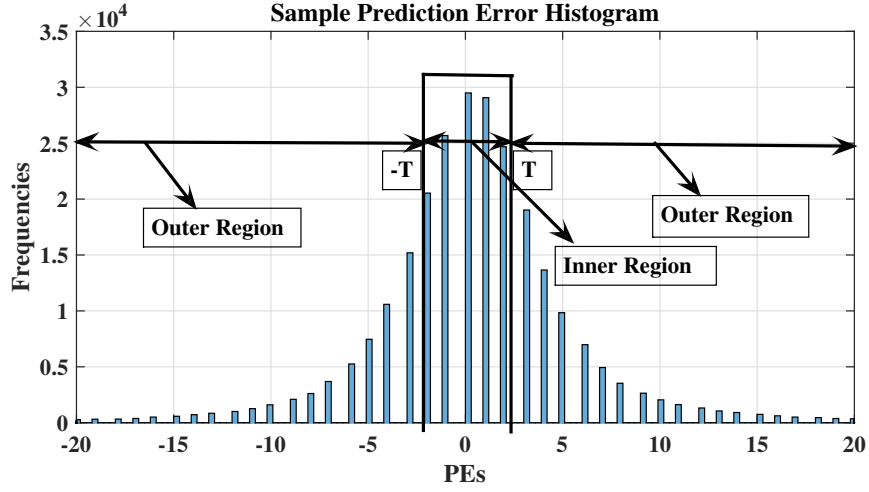


Figure 2.7: Inner region and outer region in a sample prediction error histogram

an estimated local complexity value of the pixel. Idea, here, is to embed more data bits in a pixel in a smooth region of an image as compared to a pixel in a relatively rough region of the image. Embedding of a single bit data is carried out by expanding the prediction error by multiplication of 2. Embedding of two bits is carried out by expanding the prediction error by multiplication of 4. In [132], local complexity of a pixel is estimated as standard deviation of pixel values at right, lower, lower-left and lower-right diagonal neighbors. The above adaptive embedding strategy is also adopted in [80]. But, the local complexity of a pixel is computed using the standard deviation of pixel values in the 8-neighborhood of the pixel in [80]. Another adaptive embedding strategy in [133] classifies each non-overlapping block of pixel either as a smooth or a rough block. Like the previous adaptive embedding strategies, more data bits are embedded in smooth blocks.

Instead of a multiplicative prediction error expansion in the above techniques, an additive prediction error expansion is introduced in [134]. Here, the prediction error is expanded by adding (or subtracting) either 0 or 1 (depending on the data bit) with the prediction error. Addition of the binary bit is carried out if the prediction error is either zero or positive. Subtraction of the binary bit is carried out if the prediction error is negative. Alternatively, in [135], a pair of histogram bins (left bin and right bin) are selected for embedding. In this technique, addition

2.3 Prediction Error Expansion Based Reversible Data Hiding Techniques

of a binary data bit is carried out if the prediction error falls in the right bin. Subtraction of the binary bit is carried out if the prediction error falls in the left bin.

The above conventional PEE based techniques mainly focused on correlations among the image pixels. In the conventional PEE based techniques, 1-D prediction error expansion is used to embed the data into the prediction error of a pixel. But correlation among the adjacent prediction errors is considered in 2-D prediction error expansion in [136]. Consecutive prediction errors are paired to construct a 2-D prediction error histogram. Expansion and shifting of these histogram bins are used to effectively embed data bits in the original pixels. An adaptive pairing of pixels for pairwise embedding is proposed in [73]. Embedding is carried out in the pixels in specific histogram bins. These pixels are sorted in ascending order of their local complexity values. Prediction error of a pixel is paired with the prediction error of another pixel in the neighborhood based on the local complexity of the pixel pairs. The pairing of prediction errors to form a 2-D prediction error histogram is carried out such that the local complexity values of these two pixels are very close to each other. Several other extensions of pairwise embedding strategy also exist, such as adaptive pairing of pixels using geodesic path to form 2-D prediction error histogram [137], dynamic programming based selection of suitable histogram bins for embedding [138], etc. Another 2-D prediction error histogram is constructed in [139] by considering two prediction errors of a single pixel from a pair of prediction strategies.

In another variation of prediction error histogram modification based embedding for reversible data hiding [140], multiple prediction error histograms are generated. Pixels having the same local complexity value form a group to generate a single prediction error histogram. Thus, multiple histograms are formed based on the distinct local complexity values. Then, a pair of histogram bins are selected for embedding in each histogram using an optimum histogram bin selection strategy. Similar strategy of forming multiple prediction error histograms using local complexity based grouping can also be found in [64]. As an extension of the above works, a greedy search algorithm is proposed in [141] for selecting the optimal pair of histogram bins in each histogram. Instead of local complexity

2.4 Other Reversible Data Hiding Techniques

based formation of multiple histograms, fuzzy c-means (FCM) clustering technique is used for grouping similar prediction errors into a single histogram in [142]. Thus, multiple histograms are formed for different groups of prediction errors.

Taking inspiration from the histogram bin shifting based reversible data hiding technique [41], histogram bin shifting of prediction error histograms is developed in [143]. This method finds pairs of peak and zero points in the prediction error histogram. Data bits are embedded in the pixels corresponding to the peak bin in the prediction error histogram. Histogram bins in between the peak and the zero points are shifted. An improved version of the technique in [143] is presented in [144], where embedding is carried out only in the smooth regions in the image. Otherwise, the embedding procedure in [144] remains the same as [143]. An adaptive prediction error histogram bin shifting based embedding is proposed in [145]. An improved version of communicating peak and zero points through a binary tree structure is proposed in [145]. An error energy of the prediction error is computed. Based on this prediction error energy, few of the prediction errors are chosen for embedding the data bits.

In another dimension in prediction error histogram bin shifting methods for embedding data bits, namely asymmetric histogram shifting [146], more than one prediction errors are derived for a single pixel using multiple predictors. Then, an asymmetric selection function selects a suitable prediction error to form an asymmetric histogram of prediction error. Embedding is carried out using this asymmetric histogram of prediction error. Similarly, asymmetric histogram shifting based embedding using multiple prediction errors can also be found in [78, 128, 147, 148].

Few other prediction error histogram modification based reversible data hiding techniques can be found in [149, 150, 151].

2.4 Other Reversible Data Hiding Techniques

Apart from above categories of reversible data hiding techniques, few other interesting techniques can also be found in literature. For example, a Sudoku based technique is proposed in [152]. The technique uses Sudoku properties for pairing

of pixels. The secret data is represented using 9-base number system. Then, the pixel pairs are modified to insert the converted secret data.

A reversible data hiding technique using code division multiplexing is proposed in [153]. The secret data is represented using several orthogonal spreading sequences. Walsh Hadamard matrix is used to generate these orthogonal spreading sequences. These sequences are then embedded in the cover image.

An odd-even property based reversible data hiding technique is proposed in [154]. The image is divided into several blocks. Pixel values in each block are converted to either odd value or even value by computing a certain optimal value of the block. Then, an odd-even property is used to embed the data bits into these blocks.

Reversible data hiding techniques using dual images are also proposed in [155, 156, 157, 158].

Unlike all above techniques, reversible data hiding in frequency domain is proposed in [159]. In this technique, a two-level data hiding is suggested. At first level, data is hidden using histogram modification in spatial domain. At the second level, data is also embedded in frequency domain representation of the image.

Few other reversible data hiding techniques include a multi-sub-blocking based technique [160] and a Slantlet transformation based technique [161, 162].

2.5 Summary

A survey of existing reversible data hiding techniques in a grayscale image is presented in this chapter. Earlier techniques of reversible data hiding expanded the difference between pair of pixel values to embed data bits. Prediction error expansion based techniques were introduced subsequently as natural extension of difference expansion based techniques. Prediction error is measured as the difference between a pixel value and its predicted value. Several pixel value prediction techniques have also been discussed in this context. In prediction error expansion based techniques, a prediction error is expanded to embed the data bits. Histogram bin shifting methods also hold an important place in the related literature. In this chapter, majority of the important contributions in the field

of reversible data hiding are highlighted to provide an overall perspective of the developments in this field.

Chapter 3

Reversible Data Hiding Using B-tree Triangular Decomposition Based Prediction

There exists a category of prediction error expansion based reversible data hiding schemes, where a set of reference pixels are fixed. Then, values of the remaining (non-reference) pixels are predicted using values of these reference pixels. In [118], the set of reference pixels is obtained using a reference pixel distribution mechanism. Based on the gray values of this initial set of reference pixels, the number of reference pixels is locally adjusted. Interpolation using these reference pixels predicts the non-reference pixel values. Delaunay triangulation (DT) based scheme [119] generates a 3-D Delaunay mesh considering the coordinates and the intensity values of the image pixels. The vertices of these triangles act as reference pixels. Then, the DT based method [119] uses a linear interpolation of the set of reference pixels for predicting the non-reference pixels.

Inspired by this category of works (and more specifically, due to the Delaunay triangulation (DT) based scheme [119]) of using a set of reference pixels for predicting other pixels, a novel reversible data hiding technique based on a B-tree triangular decomposition based method is proposed in this chapter. The proposed work also draws inspiration from the fact that an image coding technique can be used to predict pixel values for PEE-based reversible data hiding (like [23]). B-tree triangular coding (BTTC) has been proposed for image coding in

3.1 Prediction Based on B-Tree Triangular Decomposition

[24]. The image is recursively decomposed into right-angled triangles. But unlike [24], where decomposition into two triangles is carried out based on the prediction errors, the proposed method considers homogeneity of cover image pixel values for decomposition. Vertices of these triangles serve as reference pixels. Values of the non-reference pixels in a triangle are predicted using a planar linear interpolation of the pixel values at the vertices of the triangle. Payload bits are embedded in the expanded prediction errors of the pixels. Moreover, an adaptive embedding strategy (like [132]) is adopted for embedding either one bit or two bits of data in a pixel based on the estimated local complexity of a triangle. The novelty of the proposed reversible data hiding scheme is due to: (i) usage of B-tree triangular decomposition for determining the reference pixels and subsequent prediction, (ii) use of cover image pixel values for B-tree triangular decomposition, and (iii) an estimate of local complexity of a triangle based on the pixel values at the vertices of the triangle. Moreover, triangles are sorted in ascending order of their local complexities to embed the payload in corresponding non-reference pixels. Experimental results show the superiority of (i) the proposed B-tree triangular decomposition based prediction with respect to several other predictors, and (ii) the proposed reversible data hiding technique in comparison with few other existing reversible data hiding techniques.

The remainder of this chapter is organized as following: B-tree triangular decomposition and subsequent prediction strategy are explained in Section 3.1. The adaptive embedding scheme is explained in Section 3.2. Extraction of the data and recovery of the original pixel values are stated in Section 3.3. Experimental results are discussed in Section 3.4. Finally, summary of the contributions in this chapter is stated in Section 3.5.

3.1 Prediction Based on B-Tree Triangular Decomposition

In this chapter, a B-tree triangular decomposition based method is suggested for pixel value prediction. This section presents the concept of B-tree triangular decomposition and subsequent prediction scheme.

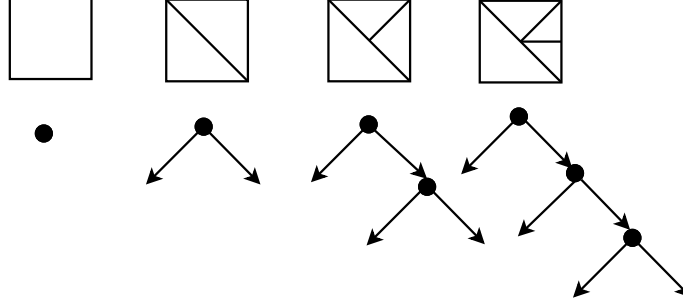


Figure 3.1: Recursive decomposition of an image into right-angled triangles

3.1.1 B-Tree Triangular Decomposition

According to B-tree triangular decomposition method, an image is recursively decomposed into a set of right-angled triangles until all pixels in the newly formed triangles satisfy a certain condition. A representational diagram of the concept is provided in figure 3.1. At first, an image is divided into two right-angled triangles through its main diagonal as shown in figure 3.1. Then, B-tree triangular decomposition suggests a recursive decomposition of each right-angled triangle into two right-angled triangles. Further decomposition of a triangle Δ is stopped if (i) the difference between the maximum and the minimum pixel values in a triangle (also considering the sides of the triangle) is less than or equal to a threshold T_d (equation 3.1) or (ii) the triangle is very small in size (i.e., the number of pixels in a triangle including the pixels on the sides of the triangle is less than or equal to 5).

$$|max(\Delta) - min(\Delta)| \leq T_d \quad (3.1)$$

$max(\Delta)$ and $min(\Delta)$ denote the maximum and the minimum pixel values in a triangle Δ (including the sides of the triangle). This ensures that the image plane is decomposed into relatively smooth smaller regions. Hence, the differences of pixel values between the vertices of a triangle and other pixels in the triangle are small (as being controlled by the threshold T_d). As these vertex pixel values are used to predict all other pixel values in the triangle, the said small differences lead to better prediction. On the contrary, the Delaunay triangulation (DT) based scheme [119] considers a set of randomly distributed pixels to form the

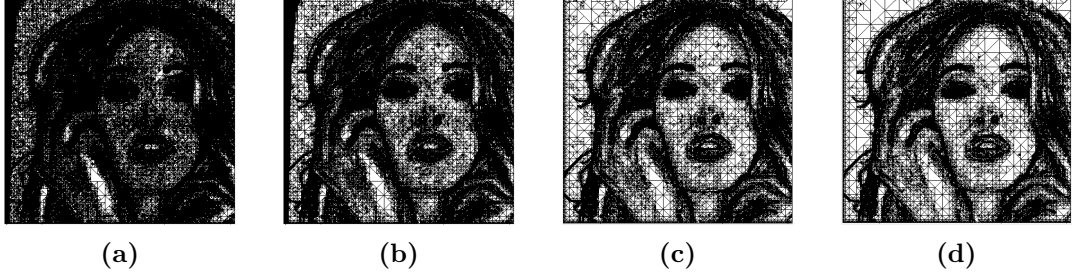


Figure 3.2: B-tree triangular decomposition of Tiffany image for various values of T_d as (a) 7, (b) 10, (c) 13, and (d) 15, respectively

triangles as per Delaunay triangulation property. This property ensures that no other vertex is positioned within the circumcircle of a Delaunay triangle. It does not consider the closeness of values among the vertex pixels and non-vertex pixels. Therefore, DT based decomposition and subsequent prediction of the non-vertex pixels using the Delaunay vertices cannot guarantee a good prediction as compared to the proposed B-tree triangular decomposition based predictor. This is evident in the experimental results (as will be presented later).

In the proposed method, the decomposition depends on the threshold T_d . Figure 3.2 shows the results of the proposed decomposition for Tiffany image for various values of T_d . Bigger triangles can be observed in the smoother regions in the image, whereas the rough regions of the image are decomposed into tiny triangles. Moreover, higher values of T_d also generate bigger triangles on an average. For example, the image is decomposed into very tiny triangles for threshold T_d as 7.

Originally, the concept of B-tree triangular decomposition for image coding was proposed in [24]. It considered the prediction errors within a triangle to decide the need for further decomposition. On the contrary, the proposed method, in this chapter, examines the range of pixel values in a triangle for this purpose. So, the proposed method does not require a set of predicted values before decomposition. Rather, the decomposition is carried out to obtain the predicted values.

3.1.2 Prediction Scheme

Vertices of the triangles (obtained as above) act as reference pixels for the proposed prediction scheme. All other pixels (non-reference) within a triangle (including the pixels which are on the three sides of the triangle) are predicted using the vertices of the triangle. Let the three vertices of a right-angled triangle be $V_1 = (m_1, n_1)$, $V_2 = (m_2, n_2)$, and $V_3 = (m_3, n_3)$. Here, V_1 is assumed to be the vertex associated with the right angle. Moreover, let x_1 , x_2 , and x_3 be the intensity values at those vertex pixels V_1 , V_2 , and V_3 , respectively. Then, any pixel (m, n) within the triangle or on the sides of the triangle can be predicted as the planar linear interpolation of the values at the vertex pixels as following [24]:

$$x' = x_1 + \alpha \times (x_2 - x_1) + \beta \times (x_3 - x_1); \quad (3.2)$$

$$\alpha = \frac{(m - m_1) \times (n_3 - n_1) - (n - n_1) \times (m_3 - m_1)}{(m_2 - m_1) \times (n_3 - n_1) - (n_2 - n_1) \times (m_3 - m_1)} \quad (3.3)$$

$$\beta = \frac{(m_2 - m_1) \times (n - n_1) - (n_2 - n_1) \times (m - m_1)}{(m_2 - m_1) \times (n_3 - n_1) - (n_2 - n_1) \times (m_3 - m_1)} \quad (3.4)$$

It is to be noted that pixels residing on the sides of a triangle are shared by two triangles. Then, two predicted values are obtained for such pixels. To tackle this problem, triangles are always visited in a particular order. The prediction value as obtained using a triangle, which is visited later, overwrites the previous predicted value. The order of visiting triangles is maintained to generate the same predicted values for these pixels during extraction too. Figure 3.3 shows the original and the predicted Tiffany images as obtained by the proposed strategy using the threshold $T_d=7$.

Prediction error is computed by taking the difference between the original pixel value x and the predicted value x' as following:

$$PE = x - x' \quad (3.5)$$

Prediction error histogram of Tiffany image for four different threshold values are presented in figure 3.4. Height of the zero bin is more for threshold $T_d=7$ as compared to other three cases. It reflects that smaller value of the threshold reduces the prediction error. This is because smaller threshold produces smoother triangles.

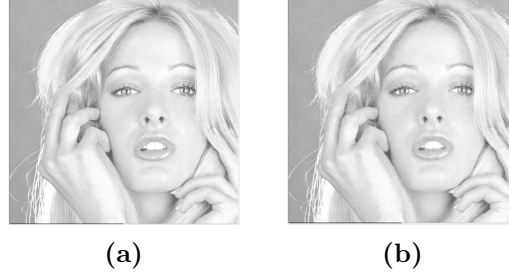


Figure 3.3: (a) Original Tiffany image and (b) the predicted Tiffany image using threshold $T_d = 7$

3.2 Embedding Using Adaptive Prediction Error Expansion

The adopted embedding strategy has the following key characteristics: (i) As per prediction error expansion strategy, the data bits are embedded in the expanded prediction error [23]. Hence, in order to reduce the embedding distortion, data bits are only expanded where prediction error is less than a threshold (T_{PE}). (ii) Moreover, similar to [132], this chapter adopts the adaptive embedding strategy where either 1 bit or 2 bits of data are being embedded depending on an estimated local complexity of the triangle in which a pixel belongs to. (iii) Triangles are visited in a sorted (ascending) order of their local complexities and embedding is carried out accordingly.

Following subsections discuss the above points of the adaptive prediction error expansion strategy.

3.2.1 Estimation of Local Complexity

According to adaptive embedding strategy [132], in general, 1 bit data is embedded in a pixel whose local complexity is high. Otherwise, 2 bits of data are embedded in a pixel having less local complexity. Embedding 2 bits in a pixel significantly increases the embedding capacity of the method, though the local complexity based decision controls the embedding distortion. Local complexity is measured either by summation of absolute differences among the context pixels

3.2 Embedding Using Adaptive Prediction Error Expansion

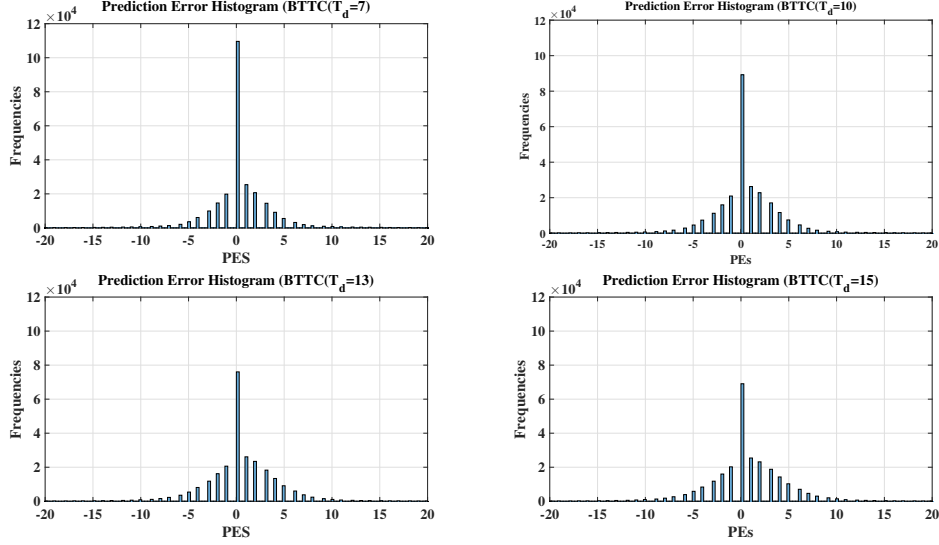


Figure 3.4: Prediction error histograms of Tiffany image for the proposed prediction scheme with various decomposition thresholds (a) $T_d = 7$, (b) $T_d = 10$, (c) $T_d = 13$, and (d) $T_d = 15$

[78], summation of squared differences among the context pixels [82], or standard deviation of the context pixels [132]. Basically, local complexity captures how the context pixel values are varying. The definition of context pixels varies across these methods. But the context pixels are chosen in such a way that the same values are available for each of these pixels during embedding and extraction to obtain the same local complexity estimation.

A simplified version of the local complexity (lc) estimation is proposed in this section using the vertex (reference) pixel values of a triangle as in equation 3.6. It reflects how different the intensity of a right-angled vertex (x_1) is from the intensities of other two vertices. The same local complexity values need to be generated for both embedding and extraction processes. Hence, only the vertex (reference) pixel values of a triangle can be used to estimate the local complexity. As all other non-reference (non-vertex) pixels may be modified during the embedding process, those non-vertex pixels in a triangle cannot be used to compute the local complexity. Therefore, a simplified estimation of local complexity as in equation 3.6 is proposed in this work.

$$lc = |x_2 - x_1| + |x_3 - x_1| \quad (3.6)$$

where $|\cdot|$ indicates the absolute value of its argument.

3.2.2 Adaptive Embedding Based on Prediction Error and Local Complexity

The adopted adaptive embedding strategy is based on the work in [132].

3.2.2.1 Embedding Two Bits in a Pixel

If the estimated local complexity lc is less than a threshold T_{lc} , two bits of data w_1w_2 are embedded as following:

$$\begin{aligned}
 x^w &= x' + 2 \times (2 \times (PE) + w_1) + w_2 \\
 &= x' + 4 \times PE + 2 \times w_1 + w_2 \\
 &= x' + 4 \times (x - x') + w \text{ as } w = 2 \times w_1 + w_2 \\
 &= 4 \times x - 3 \times x' + w = x + 3 \times PE + w
 \end{aligned} \tag{3.7}$$

where $w \in \{0, 1, 2, 3\}$ indicates two bits of data. x and x^w are the pixel values before and after embedding the data, respectively.

It can be observed from the above equation that expanding the prediction error to embed two bits of data causes significant change in the pixel value ($3 \times PE + w$). It is more if the prediction error PE is high. Hence, like the method in [132], an additional condition is added in terms of the prediction error. If the absolute value of the prediction error is less than or equal to a threshold T_{PE} , then the embedding is carried out as mentioned in above equation. Otherwise, shifting of the prediction error (and hence, shifting of the pixel value) is carried out as following:

$$\begin{aligned}
 x^w &= x - 3 \times T_{PE} \quad \text{if } PE < -T_{PE} \\
 x^w &= x + 3 \times T_{PE} + 3 \quad \text{if } PE > T_{PE}
 \end{aligned} \tag{3.8}$$

This shifting is required to identify during extraction whether the embedding has been carried out or not. During extraction, this can be understood by observing the prediction error PE^w at the marked image. Because, embedding of two bits shifts the prediction error from $[-T_{PE}, T_{PE}]$ to $[-4T_{PE}, 4T_{PE} + 3]$. Then, the prediction errors at the remaining pixels need to be shifted.

3.2.2.2 Embedding Single Bit in a Pixel

If the estimated local complexity lc is greater than or equal to the threshold T_{lc} and the absolute value of the prediction error is less than or equal to T_{PE} , one bit data w is embedded as following:

$$\begin{aligned} x^w &= x' + 2 \times PE + w \\ &= x' + 2 \times (x - x') + w \\ &= 2 \times x - x' + w = x + PE + w \end{aligned} \tag{3.9}$$

x and x^w are the pixel values before and after embedding the data, respectively. Similar to the case of 2-bit embedding, the shifting of the prediction error (and hence, shifting of the pixel value) is carried out as following:

$$\begin{aligned} x^w &= x - T_{PE} \quad \text{if } PE < -T_{PE} \\ x^w &= x + T_{PE} + 1 \quad \text{if } PE > T_{PE} \end{aligned} \tag{3.10}$$

This shifting is required to identify during extraction whether the embedding has been carried out or not. During extraction, this can be understood by observing the prediction error PE^w at the marked image. Because, embedding of one bit shifts the prediction error from $[-T_{PE}, T_{PE}]$ to $[-2T_{PE}, 2T_{PE} + 1]$. Then, the prediction errors at the remaining pixels need to be shifted.

3.2.3 Order of Embedding

It has been previously reported in [69] that embedding in the pixels in the order of their local variance enabled more embedding of bits with less degradation. Hence, the proposed work too visits the triangles (local regions) in the increasing order of their local complexities (as estimated using equation 3.6). If multiple triangles have same local complexity value (equation 3.6), then such triangles are sorted using the coordinates of their centroids. Upon visiting a triangle, embedding is carried out in the pixels within the triangle or on the sides of the triangle. Pixels within a triangle are visited in a raster scan order for embedding. It is to be noted that pixels on the sides of a triangle are part of two different triangles. To avoid embedding in such a pixel for the second time, a flag is used to mark the pixels where embedding is already carried out.

3.2 Embedding Using Adaptive Prediction Error Expansion

Experiments are also performed without visiting the triangles in such an ascending order of their local complexities as above. Rather, the embedding is carried out in the raster scan order of the pixels in the image. It is experimentally observed that the embedding based on the sorted order of triangles performs better than the simple raster scan order of embedding.

3.2.4 Auxiliary Information

Certain information about the embedding process is crucial to accurately extract the embedded data as well as to restore the cover image. These include: (a) position of the last pixel where embedding has been carried out, (b) the value of the threshold T_{lc} , (c) the value of the threshold T_{PE} , (d) whether overflow/underflow has occurred during embedding and shifting, (e) length of the compressed overflow/underflow location map, and (f) the compressed overflow/underflow location map. The items (e) and (f) are only required at the occurrence of overflow or underflow.

An overflow occurs when a pixel value exceeds 255 due to either embedding or shifting. Similarly, an underflow occurs when a pixel value becomes less than 0 due to either embedding or shifting. Hence, every non-reference (i.e., non-vertex) pixel in the image is inspected for the occurrence of the overflow/underflow condition before the actual embedding and shifting are carried out. As discussed in the previous section, the thresholds T_{lc} and T_{PE} are used for this inspection. Because the occurrence of overflow/underflow depends on whether (i) 1 bit or 2 bits of data are being embedded, and (ii) whether the pixel is being used for embedding or shifting. A binary map indicating the location of the pixels having overflow/underflow condition is termed as overflow/underflow location map. This location map is losslessly compressed using the arithmetic coding method.

At first, the auxiliary information is embedded into the first few non-reference pixels (in raster scan sequence) of the cover image using the least significant bit (LSB) replacement technique. Then, the LSBs of those non-reference pixel values are appended at the end of data stream to be embedded. But as this step is carried out before the actual embedding of payload, the position of the last pixel (part (a) of auxiliary information), where embedding has been carried out, is

not known apriori. For an image of size 512×512 , 18 bits (i.e., $2 \times \log_2(512)$) are required to specify the position of any pixel. Hence, the LSBs for first 18 non-reference pixels values (in raster scan sequence) are reserved for storing the position of the last embedded pixel. The position of the last embedded pixel will be stored at these reserved bits after the embedding of the payload. But the LSBs of these 18 pixels values are extracted and added to the payload. Next, the payload (data and LSBs) are embedded into the non-reference pixels of the cover image using the proposed adaptive prediction error expansion technique. But neither embedding nor shifting has taken place in the pixels as indicated by overflow/underflow location map. It is to be noted that the auxiliary information may occupy the first few rows in the image. Hence, those few rows are not used to embed the payload.

Once the embedding of the entire payload is completed, the position of the last embedded pixel is stored as part (a) of auxiliary information at the bits which were reserved to store this information (i.e., LSBs at the first 18 non-reference pixel values in the raster scan sequence of the image pixels as indicated above).

3.3 Extraction

The extraction process is possible with the help of auxiliary information. Initially, the auxiliary information is retrieved from the LSBs of the non-reference pixel values of the first few rows. The compressed overflow/underflow location map is obtained from it. Then, this location map is decompressed to obtain the original overflow/underflow location map. It helps to identify the pixels where change (due to embedding or shifting) has not been taken place. The position of the last embedded pixel is also obtained to recognize the completion of the extraction process. Because to extract the payload, pixels are visited in the same sequence as they were visited during embedding. Hence, knowledge about the last embedded pixel is useful to stop the extraction process. The two threshold values T_{lc} and T_{PE} are also obtained from the auxiliary information. Moreover, the set of reference (i.e., vertex) pixels (which are not modified during the embedding stage) is obtained through secret channel information. All the non-reference pixel values are predicted using these reference pixel values by following the strategy

in Section 3.1.2. The reference pixels are not changed during embedding. Therefore, same predicted values are obtained for the non-reference pixels. Similarly, the local complexities of triangles also remain the same as with embedding. The prediction error is obtained as:

$$PE^w = x^w - x' \quad (3.11)$$

The payload (data and LSBs) is extracted from the non-reference pixels which have not been specified in the overflow/underflow location map. These pixels (excluding vertex pixels and overflow/underflow locations) are traversed in the same order as mentioned in Section 3.2.3 based on the local complexities and the coordinates of the centroids of the triangles. Based on the prediction error PE^w at the marked image for such a pixel, it can be found out whether embedding or shifting has been carried out in it. Similarly, using the local complexity of the triangle, number of embedded bits (either 1 or 2) can be estimated. Simultaneously, the cover image pixel value is also restored. This is repeated until the entire payload is extracted. The knowledge about the position of the last embedded pixel (as obtained from auxiliary information) helps to mark the complete extraction of the payload. The payload comprises of data bits and LSBs of non-reference pixel values in first few rows (where auxiliary information was stored). These LSBs are used to restore the pixel values (using LSB replacement technique) of those non-reference pixels in first few rows.

3.3.1 Extraction of Two Bits

If the local complexity of a triangle is less than the threshold T_{lc} and the prediction error (PE^w) of a pixel within or on the sides of a triangle lies in the range $[-4T_{PE}, 4T_{PE} + 3]$, then the payload bits are extracted and the original pixel value is restored as following:

$$\begin{aligned} w &= \lfloor PE^w \rfloor - 4 \times \lfloor \frac{PE^w}{4} \rfloor \\ x &= x^w - 3 \times \lfloor \frac{PE^w}{4} \rfloor - w \\ &\text{where } w \in \{0, 1, 2, 3\} \end{aligned} \quad (3.12)$$

3.4 Experimental Results and Discussion

where $\lfloor \cdot \rfloor$ refers to the largest integer which is smaller than its argument.

If the local complexity of a triangle is less than T_{lc} and the prediction error is not within the range $[-4T_{PE}, 4T_{PE} + 3]$, then the pixel values are shifted back as following:

$$\begin{aligned} x &= x^w + 3 \times T_{PE} \quad \text{if } PE^w < -4T_{PE} \\ x &= x^w - 3 \times T_{PE} - 3 \quad \text{if } PE^w > 4T_{PE} + 3 \end{aligned} \quad (3.13)$$

3.3.2 Extraction of Single Bit

If the local complexity of a triangle is greater than or equal to T_{lc} and the prediction error of a pixel in the triangle lies within the range $[-2T_{PE}, 2T_{PE} + 1]$, the extraction of one bit data and the restoration of original pixel value are as following:

$$\begin{aligned} w &= \lfloor PE^w \rfloor - 2 \times \lfloor \frac{PE^w}{2} \rfloor \\ x &= x^w - \lfloor \frac{PE^w}{2} \rfloor - w \\ \text{where } w &\in \{0, 1\} \end{aligned} \quad (3.14)$$

If the local complexity of a triangle is greater than or equal to T_{lc} and the prediction error of a pixel is not within the range $[-2T_{PE}, 2T_{PE} + 1]$, then the pixel value is shifted back as following:

$$\begin{aligned} x &= x^w + T_{PE} \quad \text{if } PE^w < -2T_{PE} \\ x &= x^w - T_{PE} - 1 \quad \text{if } PE^w > 2T_{PE} + 1 \end{aligned} \quad (3.15)$$

3.4 Experimental Results and Discussion

Eight standard test images (figure 1.4) of size 512×512 are considered for testing the performance of the proposed reversible data hiding technique. The proposed method uses B-tree triangular decomposition based prediction. The image is recursively decomposed into right-angled triangles based on a suitable decomposition threshold (equation 3.1). The performance of the pixel prediction scheme (and hence, the proposed reversible data hiding scheme) depends on this decomposition threshold T_d . A good prediction scheme ensures lesser distortion between

3.4 Experimental Results and Discussion

the cover and the marked images. Therefore, a suitable threshold value is selected for each of the test images by observing the peak signal-to-noise ratio (PSNR) between the cover and the marked images for various threshold values in the range of 1 to 15. A payload of size 10000 bits were initially used to observe the PSNR values as listed in table 3.1. For this initial experiment, the value of T_{lc} was set to 0. It corresponded to single bit embedding. Moreover, the value of T_{PE} was chosen for each image to be the minimum value which is sufficient to embed 10000 bits. For each of the test images, a decomposition threshold T_d has been selected which maximizes the PSNR. As an example, this decomposition thresholds which generate the highest PSNR for Lena and Tiffany images are 4 and 7, respectively. Hence, a single decomposition threshold may not be used for all images under experiment. The test images are finally decomposed using the selected thresholds in table 3.1. It can be observed for Lake image in table 3.1 that two threshold values (7 and 8) result in same PSNR value, which is the highest among the PSNR values for this image. Finally, the decomposition threshold T_d is selected as 8 for Lake image, as a higher threshold value may generate larger triangles. Numbers of reference pixels, non-reference pixels, and triangles for these test images using the selected decomposition threshold T_d are given in table 3.2.

3.4 Experimental Results and Discussion

Table 3.1: PSNR between the cover and the marked images for all test images while hiding 10000 bits with various decomposition threshold (T_d)

Image/ T_d	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Selected T_d
Lena	56.94	56.98	57.04	57.05	56.94	56.80	56.61	56.45	56.25	56.11	55.93	55.84	55.73	55.62	55.50	4
Lake	54.10	54.22	54.03	54.44	54.64	54.85	54.91	54.91	54.85	54.76	54.65	54.55	54.47	54.43	54.30	8
Boat	53.64	53.66	54.09	54.30	54.51	54.66	54.81	54.91	54.97	55.03	54.96	54.91	54.80	54.76	54.68	10
Mandrill	51.12	51.12	51.12	51.15	51.18	51.22	51.26	51.29	51.38	51.41	51.40	51.38	51.33	51.26	51.18	10
Elaine	53.49	53.42	53.50	53.51	53.42	53.33	53.15	53.04	52.81	53.53	53.57	53.58	53.57	53.54	53.48	8
Airplane	59.12	59.03	58.72	58.46	57.98	57.59	57.21	57.03	56.91	56.77	56.58	56.38	56.24	56.17	56.20	1
Peppers	54.45	54.46	54.49	54.59	54.75	54.97	55.06	55.11	55.10	55.04	54.97	54.90	54.85	54.77	54.74	8
Tiffany	52.85	52.73	52.59	52.45	52.22	52.89	52.97	52.83	52.71	52.50	52.35	52.17	52.03	51.82	51.66	7

3.4 Experimental Results and Discussion

Table 3.2: Number of reference pixels, non-reference pixels and triangles for the selected values of decomposition threshold T_d

Image	T_d	Ref. Pixels	Non-Ref. Pixels	Triangles
Lena	4	124943	137201	238861
Lake	8	114442	147702	216914
Boat	10	110246	151898	203812
Mandrill	10	127344	134800	247410
Elaine	8	122095	140049	233800
Airplane	1	128562	133582	252533
Peppers	8	109888	152256	199864
Tiffany	7	93727	168417	169585

Subsequently, the proposed B-tree triangular decomposition based prediction scheme is compared with several other existing prediction schemes such as (i) Median Edge Detector (MED) [58], (ii) Simplified Gradient Adjusted Predictor (SGAP) [58], (iii) a multi-predictor scheme combining MED and GAP ([128]), (iv) rhombus average predictor [74], and (v) Delaunay triangulation (DT) based prediction [119]. Prediction error histogram (PEH) plots of all these comparing methods for the Tiffany image are shown in figure 3.5. As the decomposition threshold T_d has been selected to be 7 for Tiffany image (according to table 3.1), the PEH plot for this image is reported for this specific value for T_d . Hence, the PEH plot for the proposed B-tree triangular decomposition (BTTC) (top-left plot of figure 3.5) is same as the PEH plot at the top-left of figure 3.4. These plots in figure 3.5 show that the prediction error histogram of the proposed predictor (with $T_d = 7$) is more concentrated around the zero-bin. For example, the height of the zero-bin (corresponding to the pixels where prediction error is zero) for the proposed B-tree triangular decomposition based prediction (top-row left plot in figure 3.5) is much higher than that of the Delaunay triangulation based prediction (top-row right plot in figure 3.5). It implies that the proposed B-tree triangular decomposition based prediction causes lesser prediction error as compared to all other predictors as above. It establishes the superiority of the proposed B-tree triangular decomposition based prediction over these comparing methods.

Apart from the decomposition threshold T_d and the prediction scheme, two

3.4 Experimental Results and Discussion

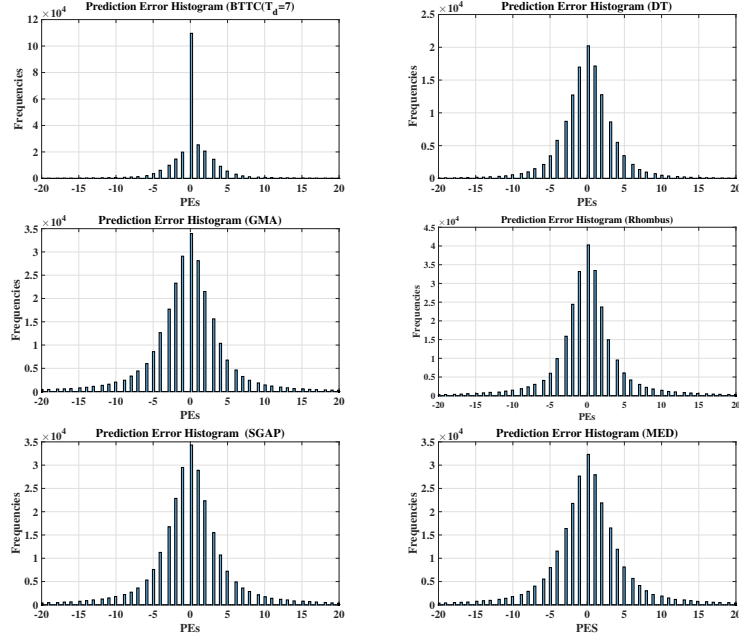


Figure 3.5: Prediction error histograms of various predictors. (Row-wise left to right) Top row: Proposed B-tree triangular decomposition based predictor and Delaunay-triangulation based predictor; Middle row: multi-predictor (GMA) and Rhombus-average; Bottom row: SGAP and MED

more thresholds - the threshold on prediction error T_{PE} and the local complexity threshold T_{lc} - influence the outcome of the proposed reversible data hiding technique. The threshold T_{PE} also decides the pixels where data bits will be embedded (equations 3.7, 3.8, 3.9 and 3.10). Hence, an increase in the value of T_{PE} also increases the embedding capacity. But to reduce the embedding distortion, the value of T_{PE} is minimally chosen which suffices to embed the payload of a certain length. Table 3.3 reports the achievable embedding capacity of the proposed scheme when only single embedding is used to reduce the distortion. This reported achievable embedding capacity corresponds to a maximum T_{PE} value of 20. Embedding capacity is measured as the ratio of the number of bits which can be embedded in an image and the number of pixels in the image. Hence, it is represented as bits per pixel (bpp). For example, a maximum of 0.52 bits per pixel can be embedded in Lena image for single embedding with the given parameter setting. Experiments were not carried out with even higher values of

3.4 Experimental Results and Discussion

Table 3.3: Achievable embedding capacity (in bpp – rounded upto two digits after decimal point) with single embedding ($T_{lc} = 0$)

Image	T_d	T_{PE}	T_{lc}	achievable bpp
Lena	4	20	0	0.52
Lake	8	20	0	0.54
Boat	10	20	0	0.55
Mandrill	10	20	0	0.41
Elaine	8	20	0	0.52
Airplane	1	20	0	0.50
Peppers	8	20	0	0.55
Tiffany	7	20	0	0.43

T_{PE} as it would introduce more embedding distortion.

But to achieve a higher embedding capacity, an adaptive embedding strategy can be adopted. Either 1 bit or 2 bits are embedded in a pixel based on a pre-decided local complexity threshold T_{lc} . The local complexity threshold T_{lc} controls the embedding capacity of the proposed scheme by controlling the number of pixels with 1-bit and 2-bit embedding. A value of 0 for T_{lc} implies that 2 bits will never be embedded in any pixel and thus, refers to single embedding. An increase in the value of T_{lc} implies that 2 bits will be embedded in more number of pixels. Hence, it increases the embedding capacity. But on the contrary, it will increase the distortion. Based on the need to embed a larger payload, a higher value of T_{lc} can be used. The achievable embedding capacity for adaptive embedding with $T_{lc} = 16$ is reported in table 3.4 for all the test images. It increases the embedding capacity for all the images (except Tiffany image) in comparison to single embedding. For example, now a maximum of 0.81 bits per pixel can be embedded in Lena image with reported parameter setting as compared to 0.52 bits per pixel for single embedding. Plots of PSNR values between the cover and the marked images against various payload sizes (bpp) have been presented in figure 3.6 for various values of T_{lc} (0, 2, 4, 8, and 16). A common observation across all test images is that the single embedding ($T_{lc} = 0$) achieves the highest PSNR. But it is limited in embedding capacity. By increasing the value of T_{lc} , higher

3.4 Experimental Results and Discussion

Table 3.4: Achievable embedding capacity (in bpp – rounded upto two digits after decimal point) with adaptive embedding ($T_{lc} = 16$)

Image	T_d	T_{PE}	T_{lc}	achievable bpp
Lena	4	20	16	0.81
Lake	8	20	16	0.76
Boat	10	20	16	0.76
Mandrill	10	20	16	0.49
Elaine	8	20	16	0.73
Airplane	1	20	16	0.82
Peppers	8	20	16	0.84
Tiffany	7	20	16	0.27

embedding capacity is achieved by compromising the PSNR. The exception has been observed for the Tiffany image. Here, the increase in the local complexity threshold T_{lc} reduces the embedding capacity (from 0.43 for single embedding with $T_{lc} = 0$ to 0.27 for $T_{lc} = 16$). This exception has been attributed to increase in size of compressed overflow/underflow location map, which is also being stored in the image as auxiliary information. More lighter pixel in the Tiffany image causes the increase in the size of overflow/underflow location map, as more overflow occurs in this case. More presence of lighter pixels in the Tiffany image as compared to other test images are visibly evident in figure 1.4.

The proposed embedding technique uses a sorted order of triangles (Section 3.2.3). As indicated in Section 3.2.3, the experiments are also carried out without considering this sorted order triangles for embedding. Plots in figure 3.6 report the performances of both kinds of embedding. These plots use 'BTTCs' as legend to indicate the embedding based on the sorted order of triangles, whereas the legend 'BTTC' denotes the embedding without the sorted order. It can be observed across all test images that the embedding using the sorted order of triangles produces marginally better PSNR values in comparison to embedding without sorting for each value of T_{lc} .

The marked images as per the proposed reversible data hiding technique for various payload size are presented in figure 3.7. Finally, the performance of the

3.4 Experimental Results and Discussion

proposed reversible data hiding using B-tree triangular based decomposition is compared with several existing reversible data hiding techniques like (i) an improved Median Edge Detector (MED) [58], (ii) an improved Simplified Gradient Adjusted Predictor (SGAP) [58], (iii) a multi-predictor scheme combining GAP, MED and Asymmetric histogram shifting (GMA) ([128]), (iv) context embedding with rhombus average predictor [74], (v) Delaunay triangulation (DT) scheme [119], (vi) Improved pairwise embedding [82] and (vii) the significant bit expansion [49]. Peak signal-to-noise ratio (PSNR) between the original image and the marked image is estimated for various payload sizes as given in figure 3.8. The proposed technique with single embedding ($T_{lc}=0$) has outperformed most of the other comparing techniques in majority of the images. Only the improved pairwise embedding technique [82] achieves better PSNR than the proposed technique across all images. But the embedding capacity of this improved pairwise technique is less. For example, a payload having 78642 bits (i.e., $\frac{78642}{512 \times 512} = 0.3$ bpp) cannot be embedded in Lena image using the improved pairwise embedding technique [82]. Similarly, Mandrill image can not accommodate even a payload of 26214 bits ($\frac{26214}{512 \times 512} = 0.1$ bpp). The improved pairwise technique is the best among the comparing methods, but it is only effective for small payload. Hence, it can be concluded that the proposed B-tree triangular decomposition based technique performs reasonably well among the comparing techniques. It is to be noted for Tiffany image that the GMA technique [128] outperforms the proposed technique for several payload sizes. Again, it can be attributed to the larger overflow location map in the proposed technique due to the lighter pixel values in Tiffany image. On the contrary, the GMA technique does not use the overflow/underflow location map.

Additionally, superiority of the proposed B-tree triangular decomposition over another existing triangular decomposition (Delaunay triangulation) is experimentally established as following: The performance of the proposed reversible data hiding technique using B-tree triangular decomposition is compared against the combination of Delaunay triangulation based decomposition/prediction and the same adaptive embedding strategy (DelaunayTri-AE as per the legend in the plots in figure 3.8) as with the proposed technique. It can be seen in figure 3.8 that the Delaunay triangulation based decomposition and prediction does not perform as

good as the proposed B-tree triangular decomposition based reversible data hiding technique, even after using the same embedding strategy. This justifies the use of the proposed B-tree triangular decomposition over the Delaunay triangulation based decomposition.

3.5 Summary

A novel reversible data hiding technique is proposed in this chapter based on B-tree triangular decomposition based prediction and an adaptive embedding strategy. B-tree triangular decomposition recursively partitions the image into right-angled triangles. The recursive decomposition continues until either the pixels in a triangle become homogeneous or a triangle becomes too small to decompose. Homogeneity of the pixels in a triangle is decided by comparing the range of the pixel values in the triangle with a decomposition threshold T_d . As it can be seen from table 3.1 that there cannot be a common threshold for all images. Hence, the threshold for each image is experimentally selected.

The vertices of the triangles in the decomposed image act as reference pixels for the prediction scheme. Other non-reference pixel values (in a triangle) are predicted using the values at the vertices of the triangle using a planar linear interpolation. The proposed prediction strategy using this B-tree triangular decomposition and planar linear interpolation outperforms several other existing predictors. It is evident from the concentration of the prediction errors near the zero-th bin of the prediction error histogram. The performance of any prediction error expansion based reversible data hiding technique depends on the performance of the predictor. Hence, numerous articles in the literature concentrate on proposing a good prediction scheme. Contribution of the proposed technique mainly lies in the stated superiority of its predictor.

Subsequently, an adaptive embedding strategy, as inspired by [132], is adopted for the proposed reversible data hiding technique. Either 1 bit or 2 bits of the payload are embedded in a pixel based on the local complexity of the local region (triangle). Unlike all other previous methods, a simplified estimate of the local complexity of the triangle is carried out using the pixel values at the vertices of the triangle. Single bit embedding in the pixels reduces the embedding distortion.

It is suggested if the payload size is relatively less. But the embedding capacity can be increased using the adaptive embedding with a compromise in the quality of the marked image. Embedding is carried out in the non-reference pixels in the ascending order of the local complexities of these triangles. Performance improvement due to this is evident from the experimental results.

With all the above components of the proposed work, the proposed B-tree triangular decomposition based reversible data hiding technique outperforms most of the existing reversible data hiding techniques. At the end, an extensive set of experimental results has been reported by varying various parameters/components of the proposed scheme.

Finally, it is to be noted that the proposed technique as well as all other existing reversible data hiding techniques in literature are fragile in nature (not robust against tampering/attack). This is due to the nature of the adopted embedding strategy using integer transforms which leads to restoration of original cover image pixels back. Hence, these reversible data hiding techniques are applicable to ensure integrity of the image, as required by several applications in medical and satellite image analysis. In these techniques, unlike robust watermarking techniques, a minor change in the marked pixel value destroys the hidden data and hence, enables identification of tampering (or rather ensures integrity of the image).

3.5 Summary

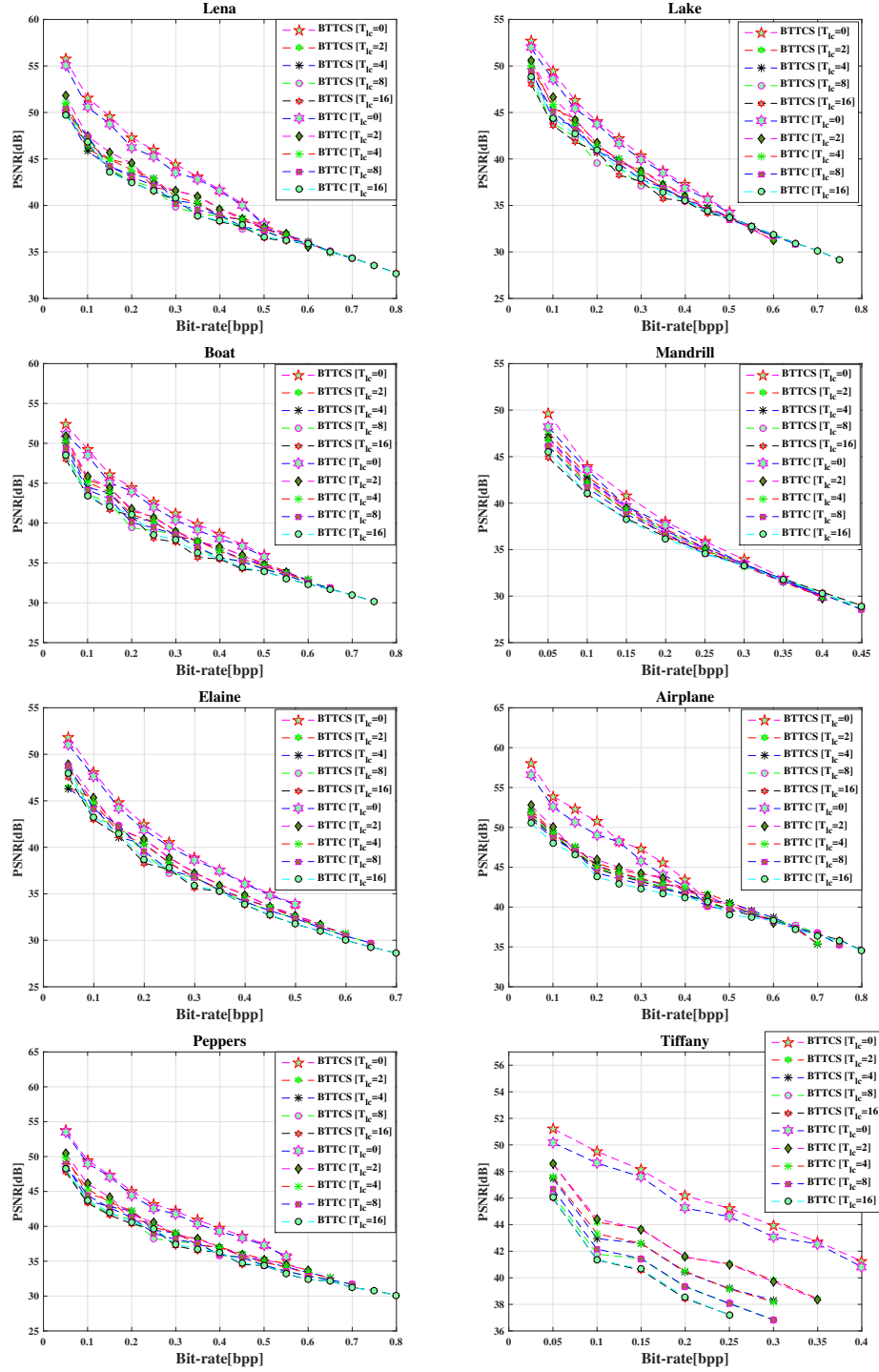


Figure 3.6: Embedding capacity versus PSNR plots using various values of local complexity threshold T_{lc} for various images.



Figure 3.7: Results of the proposed B-tree triangular decomposition based reversible data hiding scheme for various test images (row-wise). Columns from left to right: Original image, marked images with 0.1 bpp, 0.2 bpp, 0.3 bpp, and 0.4 bpp.

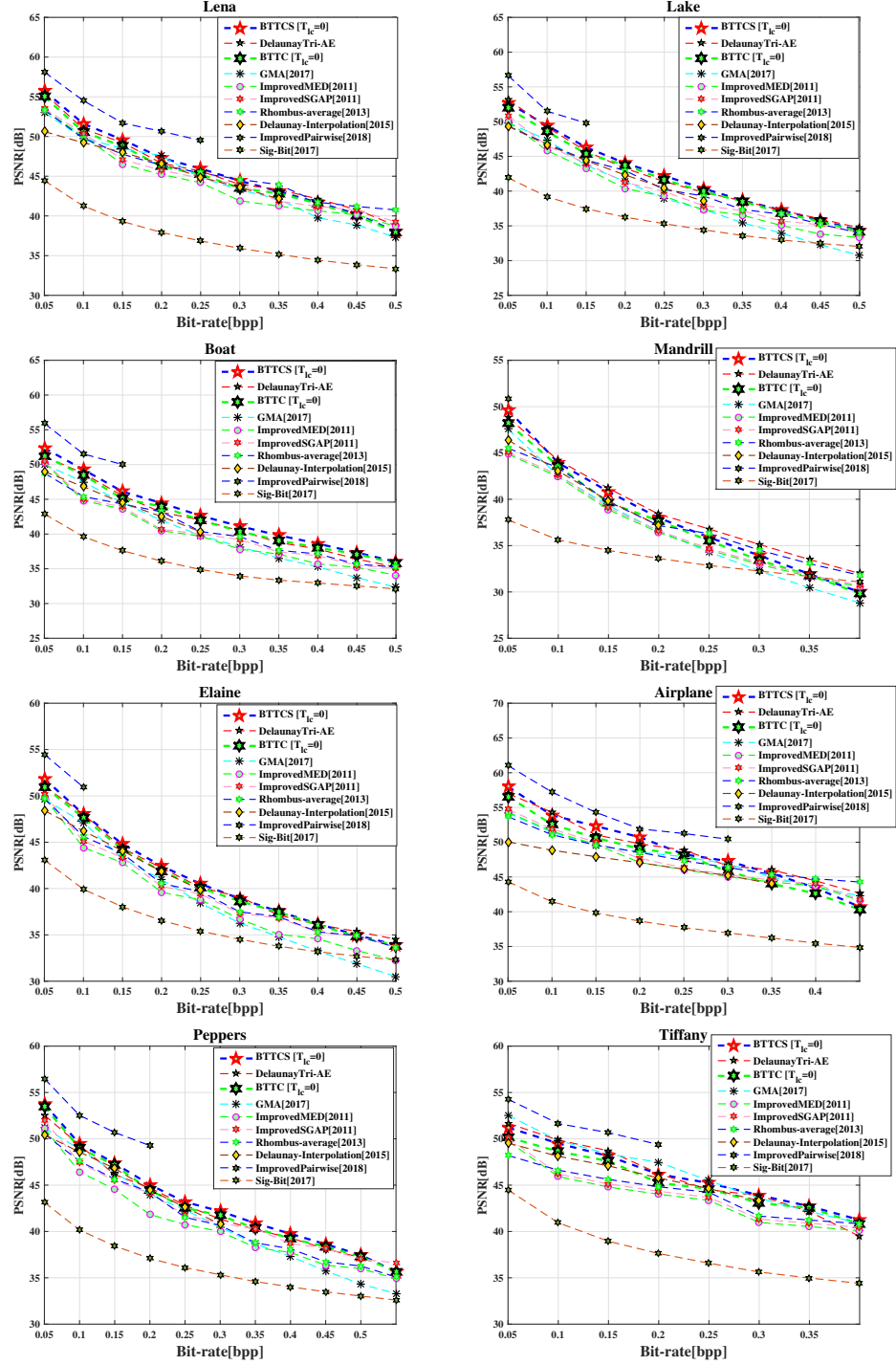


Figure 3.8: Performance comparison among several reversible data hiding techniques for various images

Chapter 4

Reversible Data Hiding Based on Random Distribution of Reference Pixels

In the reversible data hiding technique based on B-tree triangular decomposition based prediction (in Chapter 3), an image is partitioned into homogeneous regions having the shape of right-angled triangles using B-tree triangular decomposition method. Vertices of these triangles are used as reference pixels to predict other non-reference pixel values using a planar linear interpolation. On the contrary, an idea of randomly distributing the reference pixels throughout the image is proposed in this chapter. The image is divided into non-overlapping blocks of uniform sizes. Number of reference pixels in a block is proportional to the roughness of the non-overlapping blocks of the image. The value of a non-reference pixel is predicted as a weighted median of values at the reference pixels, which are located within a certain distance from the non-reference pixel. Similar to B-tree triangular decomposition based reversible data hiding technique, the adaptive embedding strategy is used in the proposed random distribution of reference pixel based reversible data hiding technique to embed either one bit or two bits of data based on the local complexity. Detailed experiments are carried out by varying the block size and the percentage of reference pixels among all pixels in the image.

The remainder of this chapter is organized as following: A prediction mechanism based on random distribution of reference pixels is proposed in Section

4.1. Embedding and extraction strategies of the proposed technique are similar to those strategies in the B-Tree Triangular decomposition based reversible data hiding technique (Chapter 3). Hence, only the differences with respect to the previous technique are highlighted in Section 4.2 and 4.3. Experimental results are discussed in Section 4.4. Finally, a summary of the contributions in this chapter is stated in Section 4.5.

4.1 Prediction Based on Random Distribution of Reference Pixels

Prediction of a pixel value is a key issue in a prediction error expansion based reversible data hiding technique. A good predictor significantly improves the performance of a reversible data hiding technique. It can be seen from the literature that several predictors have been proposed to predict a pixel value in the context of reversible data hiding. A few of those approaches consider a set of pixels in the image as reference pixels ([118] and [119]). Similarly, the B-tree triangular decomposition based prediction in Chapter 3 also establishes a set of reference pixels. Other pixel values (non-reference) are predicted using the values of these reference pixels. But these approaches vary in how (i) the set of reference pixels is decided and (ii) how a non-reference pixel value is predicted using these reference pixel values. Contrary to previous approaches, the proposed work provides a guidance to distribute the reference pixels throughout the image based on the variances of the pixel values in the non-overlapping square blocks. This strategy of random distribution of reference pixels is proposed in Section 4.1.1. Then, the value at a non-reference pixel is predicted using a set of reference pixels which are located within a certain distance from the non-reference pixel. A weighted median based predictor is considered here. The weight to a reference pixel is inversely proportional to the distance of the reference pixel from the non-reference pixel. The proposed prediction strategy is explained in Section 4.1.2.

4.1.1 Random Distribution of Reference Pixels

The proposed strategy of randomly distributing a fixed number of reference pixels throughout the image is presented in this subsection as following:

- The original cover image is divided into non-overlapping square blocks of equal size.
- Variance of pixel values in a block decides the number of reference pixels in the block. Hence, the variance of pixel values in a b^{th} block is computed as:

$$v_b = \frac{1}{p-1} \sum_{i=1}^p (x_{i,b} - \bar{x}_b)^2 \quad (4.1)$$

$$\text{where } \bar{x}_b = \frac{1}{p} \sum_{i=1}^p x_{i,b}$$

where $x_{i,b}$ refers to the value of the i^{th} pixel in the b^{th} block. $\bar{x}_b$ is the mean pixel value of the b^{th} block. p is the number of pixels in the block. (As each block is of same size, the suffix b is not used for the number of pixels p).

- Reference pixels are distributed among these equally sized blocks such that the number of reference pixels in a block is loosely proportional to the variance of pixel values in the block. The variance v_b of pixel values in a block b is used as the measure of roughness of the block. Hence, a rough block will have more reference pixels as compared to a relatively smooth block. Let N be the total number of reference pixels. These reference pixels are distributed in each block by following a procedure as mentioned here. At first, a real number r_b is obtained as following:

$$r_b = N \times \frac{v_b}{\sum_{j=1}^B v_j} = i_b + f_b \quad (4.2)$$

where B is the total number of blocks in the image. i_b and f_b are the integer and the fractional parts of the real number r_b which is obtained from equation 4.2. Initially, the number of allocated reference pixels (n_b) for the b^{th} block is computed as the median of 1, i_b , and $(p-1)$:

$$n_b = \text{median}(1, i_b, p-1) \quad (4.3)$$

4.1 Prediction Based on Random Distribution of Reference Pixels

where i_b is the integer part as in equation 4.2. Equation 4.3 ensures that at least one reference pixel is present in each block. It also ensures that a maximum of $(p - 1)$ number of reference pixels are there in a block having p pixels. Therefore, in a block, at least one non-reference pixel is present whose value will be predicted. The integer part of r_b is considered in equation 4.3 because the number of reference pixels in a block is always an integer.

- But as a result of equations 4.2 and 4.3, the summation of number of reference pixels in each block may not match with the total number of reference pixels N .

$$\sum_{b=1}^B n_b \neq N \quad (4.4)$$

The above inequality is converted into equality so that N number of reference pixels can be perfectly distributed among B number of blocks. The difference between the right and the left hand sides of the inequality in equation 4.4 ($N - \sum_{b=1}^B n_b$) is adjusted by adopting the following strategy:

- A positive difference value indicates that more reference pixels are needed in some of the blocks to bring down the difference to zero. The blocks are analyzed in decreasing order of the fractional part f_b in equation 4.2. If the number of reference pixels n_b in a block b is less than $(p - 1)$, then n_b is increased by 1, i.e., an additional reference pixel is allotted to this block. This maintains the constraint that there is at least one non-reference pixel in a block. This process is repeated for all the blocks in decreasing order of fractional part f_b until the difference becomes zero.
- A negative difference value indicates that more than the desired number of pixels have been allotted as reference pixels. Hence, the blocks are analyzed in increasing order of the fractional part f_b in equation 4.2. If the number of reference pixels n_b in a block b is greater than 1, then n_b is decreased by 1, i.e., one reference pixel is withdrawn for this block. This maintains the constraint that there is at least one

4.1 Prediction Based on Random Distribution of Reference Pixels

reference pixel in a block. This process is repeated for all the blocks in increasing order of f_b until the difference becomes zero.

- Finally, n_b number of pixels are randomly selected as reference pixels among the p number of pixels in each block b .

4.1.2 Prediction Based on Nearby Reference Pixels

Following strategy is adopted for predicting a non-reference pixel value from the surrounding reference pixel values:

- Every reference pixel within a distance d from the non-reference pixel is considered for predicting the pixel value. A chess board distance measure is used for this purpose. The chess board distance between two pixels (m_1, n_1) and (m_2, n_2) is computed as following:

$$d_{12} = \max\{|m_1 - m_2|, |n_1 - n_2|\} \quad (4.5)$$

where $|\cdot|$ indicates absolute value of its argument.

- A weighted median of the reference pixel values within the chess board distance d is considered as the predicted value for the non-reference pixel. To compute the weighted median of k numbers, each number x_i is repeated w_i times. Here, w_i is the weight associated with the number x_i . The sequence of numbers (with repetition) is sorted. Finally, the median is computed as the middle value of the sorted sequence as following:

$$\begin{aligned} WM(x_1, x_2, \dots, x_k) &= \\ median(\{x_1, x_2, \dots, x_k\} \diamond \{w_1, w_2, \dots, w_k\}) & \end{aligned} \quad (4.6)$$

$$\begin{aligned} &= median(x_1 \diamond w_1, x_2 \diamond w_2, \dots, x_k \diamond w_k) \\ &\quad median(y_1, y_2, \dots, y_K) \\ &= \begin{cases} y_{(K+1)/2} & \text{if } K \text{ is odd} \\ \frac{y_{K/2} + y_{K/2+1}}{2} & \text{if } K \text{ is even} \end{cases} \end{aligned} \quad (4.7)$$

where w_i is a positive integer weight. $x_i \diamond w_i$ implies w_i number of repetitions of reference pixel value x_i . The weight w_i has been assigned as

$$w_i = d + 1 - d_i \quad (4.8)$$

4.2 Adaptive Prediction Error Expansion

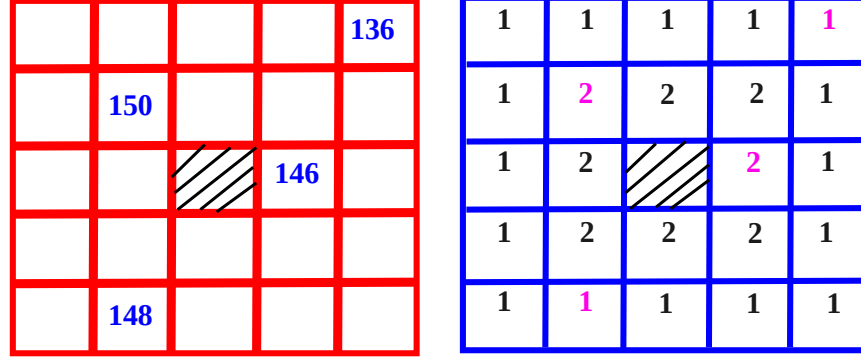
where d_i is the chess board distance between the i^{th} reference pixel and the current non-reference pixel. The weights are inversely proportional to the distance of the reference pixel from the non-reference pixel d_i . d is the maximum chess board distance being considered to find a reference pixel in the neighbourhood. Normally, the value of d is used as one less than the length at one side of the block. For example, the value of d is considered as 3 for a 4×4 block. Following justification can be given for such a value of d . A reference pixel is considered for prediction of a pixel even if the reference pixel and the pixel being predicted are at diagonally opposite corners of a block. This strategy will incorporate the reference pixels from the other blocks too. But inclusion of more reference pixels may provide better prediction.

The example in figure 4.1 explains how the weighted median based predictor works. Let the center pixel value in the 5×5 window is predicted using four reference pixel values in the neighbourhood within chess board distance 2, as it is shown in the left side of figure 4.1. The weight matrix can be derived using equation 4.8 as in the right side of figure 4.1. The weights being associated with the reference pixel values are highlighted in red. The weights being associated with the reference pixel values 136 and 148 are 1. Similarly, the weights being associated with the reference pixel values 146 and 150 are 2. The weighted median is obtained as 147, as it is shown in figure 4.1.

Based on the proposed prediction strategy, the original Peppers image and its predicted image are shown in figure 4.2. Number of reference pixels N , the block size and the chess board distance are assumed to be 25% of total number of pixels in the image, 4×4 and 3, respectively.

4.2 Adaptive Prediction Error Expansion

In a prediction error expansion (PEE) based reversible data hiding technique, the prediction error (PE) is expanded to embed the data into a cover image pixel. Prediction error (PE) is obtained by estimating the difference between the original



Calculation of Weighted Median= Median (136,146,146,148,150,150)
 $= (148+146)/2=147$

Figure 4.1: An example of predicting a pixel value by weighted median of reference pixel values in the neighbourhood ($d = 2$)



Figure 4.2: Original image (Peppers) and its predicted image (number of reference pixels = 25% of total number of pixels in the image, block size= 4×4 , and maximum chess board distance of a pixel in the considered neighbourhood = 3)

pixel value x and the predicted value x' .

$$PE = x - x' \quad (4.9)$$

Similar to the embedding strategy in the B-tree triangular decomposition based reversible data hiding technique in Chapter 3, the current chapter adopts an adaptive PEE strategy [132]. Either 1 bit or 2 bits of data are embedded depending on the local complexity of the pixel. This section describes the adopted adaptive prediction error expansion strategy, which is similar to the embedding strategy of the technique in Chapter 3. But the discussion mainly highlights the necessary differences between the embedding strategies between the current random distribution of reference pixel based technique and the B-tree triangular

decomposition based technique.

4.2.1 Estimation of Local Complexity

Similar to the technique in Chapter 3, non-reference pixels are visited in an ascending order of their local complexities to carry out the embedding. Local complexity of a pixel is determined as the variance of reference pixel values within a certain distance d from the non-reference pixel. Let p_r be the number of reference pixels within distance d of a non-reference pixel. The local complexity of this non-reference pixel is estimated as following:

$$lc = \frac{1}{p_r - 1} \sum_{i=1}^{p_r} (x_i - \bar{x})^2 \quad (4.10)$$

$$\text{where } \bar{x} = \frac{1}{p_r} \sum_{i=1}^{p_r} x_i$$

where x_i refers to i^{th} reference pixel value. $\bar{x}$ is the mean pixel value of the p_r number of reference pixels. This variance in equation 4.10 is used to determine whether a pixel is located in a smooth or a rough region.

For a non-reference pixel, same set of reference pixels is used for pixel value prediction as well as estimation of local complexity. The estimated local complexity value should be same during embedding and extraction processes. Hence, only the reference pixel values are used for computing the local complexity.

4.2.2 Order of Embedding

It has been previously reported in [69] that embedding in the pixels in ascending order of their local complexities introduces a better trade-off between the payload size and the embedding distortion. Even, the experimental results being reported in the context of B-tree triangular decomposition based reversible data hiding technique (figure 3.6) also establish the same fact. Hence, the proposed work visits the pixels in ascending order of their local complexities. Moreover, pixels having the same local complexity value are visited in their raster scan order.

4.2.3 Adaptive Embedding Based on Prediction Error and Local Complexity

An adaptive embedding strategy was first proposed in [132]. Based on the local complexity of a pixel, the method determines to embed either 1 bit or 2 bits of data. The adopted strategy, here, is exactly same as the embedding strategy for the B-tree triangular decomposition based reversible data hiding technique in previous chapter (Section 3.2.2). Hence, this adaptive embedding strategy is not repeated here. It can be found in Section 3.2.2. Moreover, the compressed overflow/underflow location map and other necessary parameters for extracting the data bits as well as for restoring the cover image are stored as auxiliary information. Even, the components of the auxiliary information and the method of embedding the auxiliary information are exactly same as with those in the B-tree triangular decomposition based reversible data hiding technique. A description of those can be found in Section 3.2.4.

4.3 Extraction

Extraction of hidden data bits and restoration of cover image pixel values are possible with the help of auxiliary information. At first, the set of reference pixels (which are not modified during the embedding stage) is obtained as a side channel information. Then, the auxiliary information is retrieved from the least significant bits (LSBs) of the non-reference pixel values of the top few rows in the image. The compressed overflow/underflow location map is obtained from it. Then, the compressed overflow/underflow location map is decompressed to obtain the original overflow/underflow location map. It specifies the pixels where changes (due to embedding or shifting) have not been taken place. Two threshold values T_{lc} and T_{PE} are also obtained from the auxiliary information.

Every non-reference pixel value is predicted using a set of reference pixel values following the strategy as discussed in Section 4.1.2. The reference pixels are not changed during embedding. Therefore, predicted value for a pixel during extraction is as same as that during embedding (x'). Then, the prediction error

is computed as following:

$$PE^w = x^w - x' \quad (4.11)$$

where x^w is the marked non-reference pixel value.

Based on the prediction error and the local complexity value of a pixel, extraction of the data bits and restoration of the cover image pixel value are carried out. The strategy is exactly similar as the strategy of the B-tree triangular decomposition based reversible data hiding technique (Section 3.3).

4.4 Experimental Results

A set of eight standard test images (figure 1.4) of size 512×512 is considered to evaluate the performance of the proposed random distribution of reference pixel based reversible data hiding technique. The proposed reversible data hiding technique randomly distributes a set of reference pixels in each block depending on the variances of pixel values in all the blocks in an image. Hence, experiments are performed by varying the number of reference pixels (or rather percentage of these reference pixels over total number of pixels) in the image. Number of reference pixels are varied as 25%, 50% and 75% of the total number of pixels in the image. Various block sizes are also considered as 2×2 , 4×4 and 8×8 . Similarly local complexity threshold T_{lc} is also varied as 0, 1, 2, 4, 8 and 16. Various combinations of these parameters - percentage of reference pixels, block size and local complexity threshold - are mentioned in table 4.1. A total of 54 experimental cases exist for each payload size and for each image.

Plots in figure 4.3 exhibit the performance of the proposed technique with various combinations of percentages of reference pixels and block sizes. Peak signal-to-noise ratio (PSNR) values between the cover images and the corresponding marked images are observed by varying the payload size while keeping the local complexity threshold T_{lc} as 0 (i.e., single embedding). The proposed method is based on random distribution of reference pixels. The set of reference pixels changes for every execution of the proposed technique. As a result, the observed PSNR values change too. Hence, each PSNR value for the proposed technique in the plots in figure 4.3 and in all other subsequent plots is an average of the PSNR

Table 4.1: Various parameter combinations for reported experiments

Percentage of Reference Pixels	Block Size	T_{lc}	Number of Experimental Cases
25	2×2	0, 1, 2, 4, 8, 16	6
	4×4	0, 1, 2, 4, 8, 16	6
	8×8	0, 1, 2, 4, 8, 16	6
50	2×2	0, 1, 2, 4, 8, 16	6
	4×4	0, 1, 2, 4, 8, 16	6
	8×8	0, 1, 2, 4, 8, 16	6
75	2×2	0, 1, 2, 4, 8, 16	6
	4×4	0, 1, 2, 4, 8, 16	6
	8×8	0, 1, 2, 4, 8, 16	6

values of ten executions of the proposed technique with the concerned parameter setting. In this context, the value of prediction error threshold T_{PE} is chosen as the required minimum value to embed a payload. It is observed from these plots that more payload can be embedded if the percentage (and hence, the number) of reference pixels is less. Hence, more number of pixels are left as non-reference pixels to embed the payload bits. Therefore, percentage of reference pixel has very prominent influence in the performance of the proposed technique. Among the three considered percentages of reference pixels, the highest amount of embedding is possible in the case of 25% of pixels as reference pixels. The lowest amount of embedding is possible in the cases of 75% of pixels as reference pixels. A comparison of performances among the cases of various block sizes with fixed percentage of reference pixels (25%) reveals the closeness of PSNR values with block sizes of 2×2 and 4×4 . For some of the images (e.g., the Lena and the Lake images), the block size of 4×4 exhibits marginally better performance than the block size of 2×2 . On the other hand, for the Mandrill image, the block size of 2×2 exhibits marginally better performance than the block size of 4×4 . However, the performance of the proposed technique with block size 8×8 is clearly inferior to the performances of the technique with other comparing block sizes.

For the rest of the reported experiments in this chapter, the best combination of percentage of reference pixels and block size is considered for each test image. The best combination of these two parameters can be found from the plots in figure 4.3. For example, in the case of Peppers image, the proposed technique with 25% of pixels in the image as reference pixel and a block size of 4×4 demonstrates better performance than the same technique with other combinations of these two parameters. Superiority of the proposed prediction scheme with the selected parameter values is demonstrated in figure 4.4 using normalized histogram plots of prediction error for the Peppers image. Here, normalized histogram of the prediction error using the proposed technique is compared with the normalized histograms of the prediction errors using (i) a multi-predictor scheme (GMA) combining GAP and MED predictors ([128]) and (ii) B-tree Triangular decomposition based prediction (Section 3.1) with the value of decomposition threshold as 8 (table 3.1). Number of pixels, whose values are predicted, vary across these schemes. Hence, normalized histograms are presented in figure 4.4, where height of each bin is divided using the total number of pixels being predicted. Otherwise, more number of reference pixel may lead to more height of the zero-th bin in a prediction error histogram. It might mislead the analysis. Hence, normalized histograms are used here for comparison. More height of the zero-th bin in a normalized histogram indicates that prediction error is zero for more number of pixels. Hence, it indicates a better prediction.

Two thresholds - namely, the prediction error threshold T_{PE} and the local complexity threshold T_{lc} - control the performance of the proposed random distribution of reference pixel based reversible data hiding technique. The threshold T_{PE} decides the pixels where payload bits are embedded (equations 3.7, 3.8, 3.9 and 3.10). Hence, an increase in the value of T_{PE} also increases the embedding capacity. But to reduce the embedding distortion, the value of T_{PE} is minimally chosen which suffices to embed the payload of a certain length. The local complexity threshold T_{lc} decides whether 1 bit or 2 bits of payload are embedded in a pixel. Thus, it controls the achievable embedding capacity of the proposed technique. Table 4.2 reports the achievable embedding capacity of the proposed technique when only single bit embedding is used (i.e., $T_{lc}=0$). This reported

4.4 Experimental Results

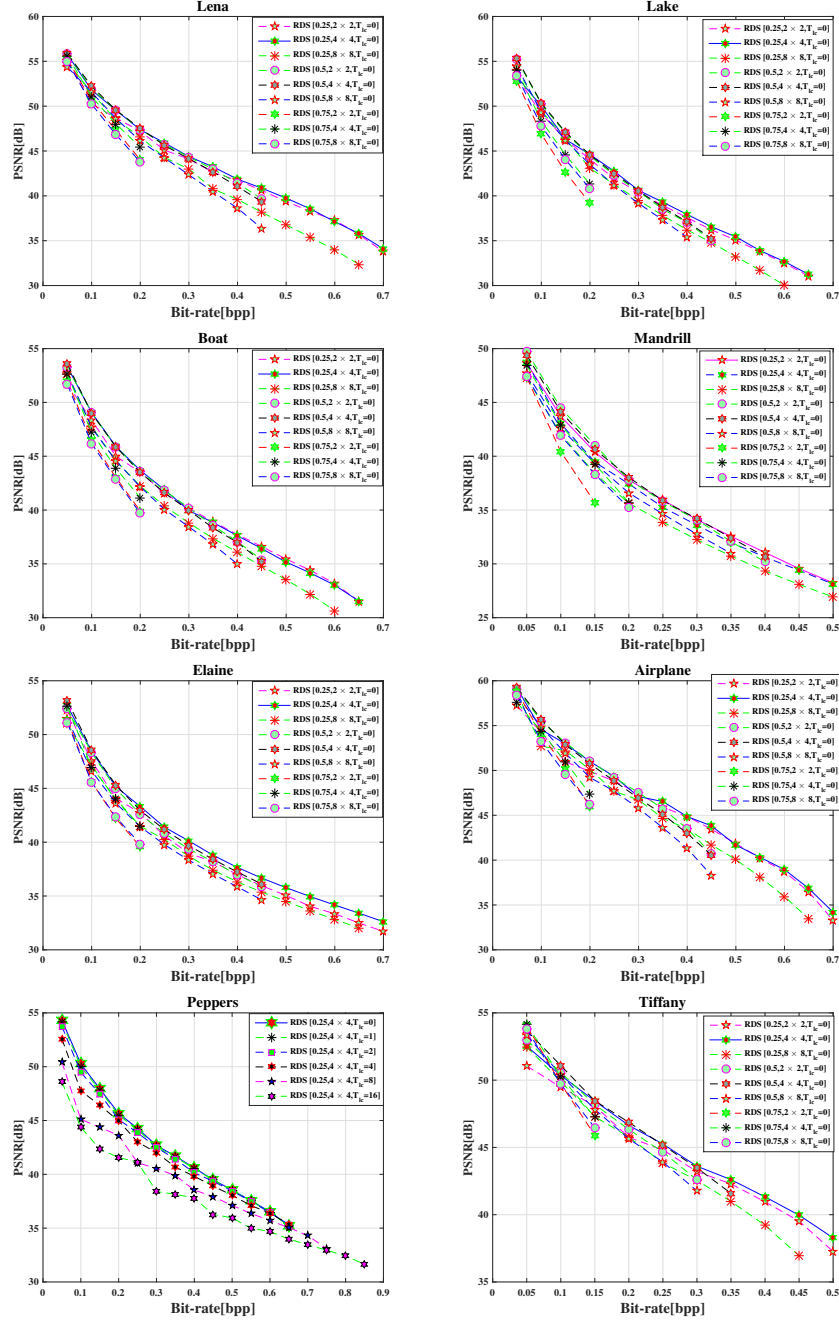


Figure 4.3: Payload size versus PSNR plots for various parameter combinations (percentage of reference pixels and block size) with local complexity threshold $T_{lc} = 0$

4.4 Experimental Results

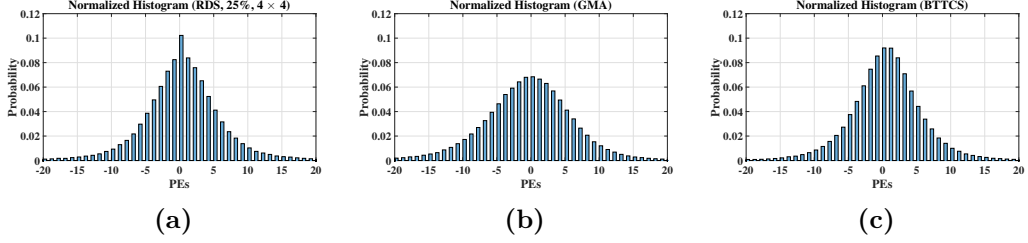


Figure 4.4: Normalized prediction error histograms of various predictors for Peppers image: (a) Proposed predictor using random distribution of reference pixels having 25% of pixels as reference pixels and 4×4 block size, (b) GMA , and (c) B-tree triangular decomposition based prediction with decomposition threshold T_d as 8

Table 4.2: Achievable embedding capacity (in bpp – rounded upto two digits after decimal point) with single embedding ($T_{lc} = 0$)

Image	Reference Pixels(%)	Block Size	T_{PE}	T_{lc}	achievable bpp
Lena	25%	4×4	20	0	0.71
Lake	25%	4×4	20	0	0.67
Boat	25%	4×4	20	0	0.67
Mandrill	25%	2×2	20	0	0.54
Elaine	25%	4×4	20	0	0.71
Airplane	25%	4×4	20	0	0.71
Peppers	25%	4×4	20	0	0.66
Tiffany	25%	4×4	20	0	0.51

achievable embedding capacity corresponds to a maximum T_{PE} value of 20. Embedding capacity is measured as the ratio of the number of bits which can be embedded in an image and the number of pixels in the image. Hence, it is represented as bits per pixel (bpp). For example, a maximum of 0.66 bits per pixel can be embedded in the Peppers image for single embedding with the given parameter setting. Experiments are not carried out with even higher values of T_{PE} as it would introduce more embedding distortion.

To achieve more embedding capacity, the proposed work uses the adaptive embedding strategy. Either 1 bit or 2 bits of payload are embedded in the pre-

diction error of a pixel based on the local complexity threshold T_{lc} . Two bits of payload are embedded in the pixels having local complexity less than a threshold T_{lc} . One bit of payload is embedded in the pixels having local complexity either greater than or equal to the threshold T_{lc} . Hence, usage of adaptive embedding increases the achievable embedding capacity of the proposed technique. Achievable embedding capacities of the proposed technique with the value of T_{lc} as 16 are reported in table 4.3 for the test images. Comparison of achievable embedding capacities between table 4.2 and table 4.3 reveals the increase of embedding capacities due to adaptive embedding for most of the images. For example, the embedding capacity has increased from 0.66 (single bit embedding) to 0.80 (adaptive embedding) for the Peppers image. Similar to the case of B-tree triangular decomposition based reversible data hiding technique (Chapter 3), only exception in this context is noticed for the Tiffany image. For the Tiffany image, embedding capacity of the proposed random distribution of reference pixel based reversible data hiding technique slightly decreases from 0.51 (single bit embedding) to 0.50 (adaptive embedding). This exception has been attributed to increase in size of compressed overflow/underflow location map, which is also being stored in the image as auxiliary information. More lighter pixel in the Tiffany image causes the increase in the size of overflow/underflow location map, as more overflow occurs in this case. More presence of lighter pixels in the Tiffany image as compared to other test images are visibly evident in figure 1.4.

In the reported experiments, value of the local complexity threshold T_{lc} is varied as 0, 1, 2, 4, 8, and 16 to control the adaptive embedding process. The value of T_{lc} as 0 indicates embedding of 1 bit payload in the prediction error of a pixel. A higher value of T_{lc} indicates 2 bits of payload can be embedded in more number of pixels. Hence, embedding capacity increases with an increase in the value of T_{lc} . Simultaneously, PSNR value decreases with an increase in the value of T_{lc} . These findings are obvious from the plots in figure 4.5.

Finally, the performance of the proposed random distribution of reference pixel based reversible data hiding technique is compared with other existing techniques in the literature: (i) a multi-predictor scheme combining GAP, MED and Asymmetric histogram shifting (GMA) [128], (ii) Improved pairwise embedding [82]

4.4 Experimental Results

Table 4.3: Achievable embedding capacity (in bpp – rounded upto two digits after decimal point) with adaptive embedding ($T_{lc} = 16$)

Image	<i>Reference Pixels</i> (%)	<i>Block Size</i>	T_{PE}	T_{lc}	achievable bpp
Lena	25%	4×4	20	16	1.01
Lake	25%	4×4	20	16	0.84
Boat	25%	4×4	20	16	0.80
Mandrill	25%	2×2	20	16	0.65
Elaine	25%	4×4	20	16	0.81
Airplane	25%	4×4	20	16	1.05
Peppers	25%	4×4	20	16	0.80
Tiffany	25%	4×4	20	16	0.50

and (iii) the significant bit expansion [49] and (iv) B-Tree triangular decomposition based technique (Chapter 3). The proposed technique with single embedding ($T_{lc}=0$) has outperformed most of the other comparing methods in majority of the images. Only the improved pairwise embedding technique [82] achieves better PSNR values than the proposed technique across all images (figure 4.6). But the embedding capacity of this improved pairwise technique is low. For example, a payload having 65536 bits (i.e, $\frac{65536}{512 \times 512} = 0.25$ bpp) cannot be embedded in Tiffany image using the improved pairwise embedding technique [82]. Similarly, Mandrill image can not accommodate even a payload of 26214 bits ($\frac{26214}{512 \times 512} = 0.1$ bpp). The improved pairwise technique is the best among the comparing methods in terms of PSNR values, but it is only effective for small payload. Hence, the proposed technique outperforms most of the existing techniques for various payload sizes. For most of the images, the proposed technique marginally outperforms the B-Tree triangular decomposition based reversible data hiding too (Chapter 3).

It has been mentioned earlier that the proposed technique is executed ten times with each parameter setting due to the inherent randomness in the technique. The above plots exhibit the average PSNR values of ten executions. Alongside the average PSNR values, standard deviations of these PSNR values over ten executions are also estimated to understand the variation in the results from one execution to another. These standard deviations are observed for each combination of pa-

4.4 Experimental Results

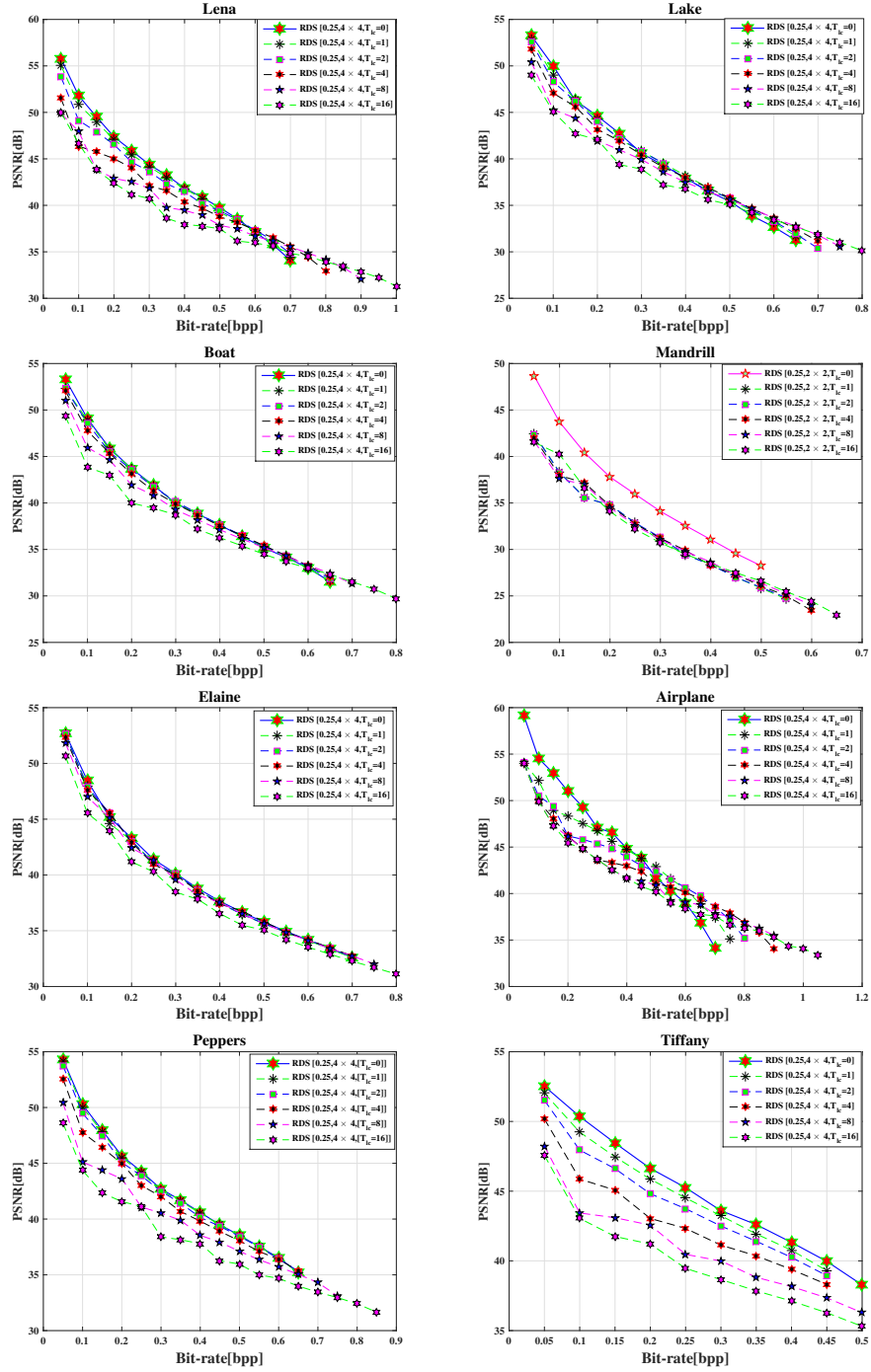


Figure 4.5: Payload size versus PSNR plots for various local complexity threshold values with pre-selected values of percentage of reference pixel and block size

4.4 Experimental Results

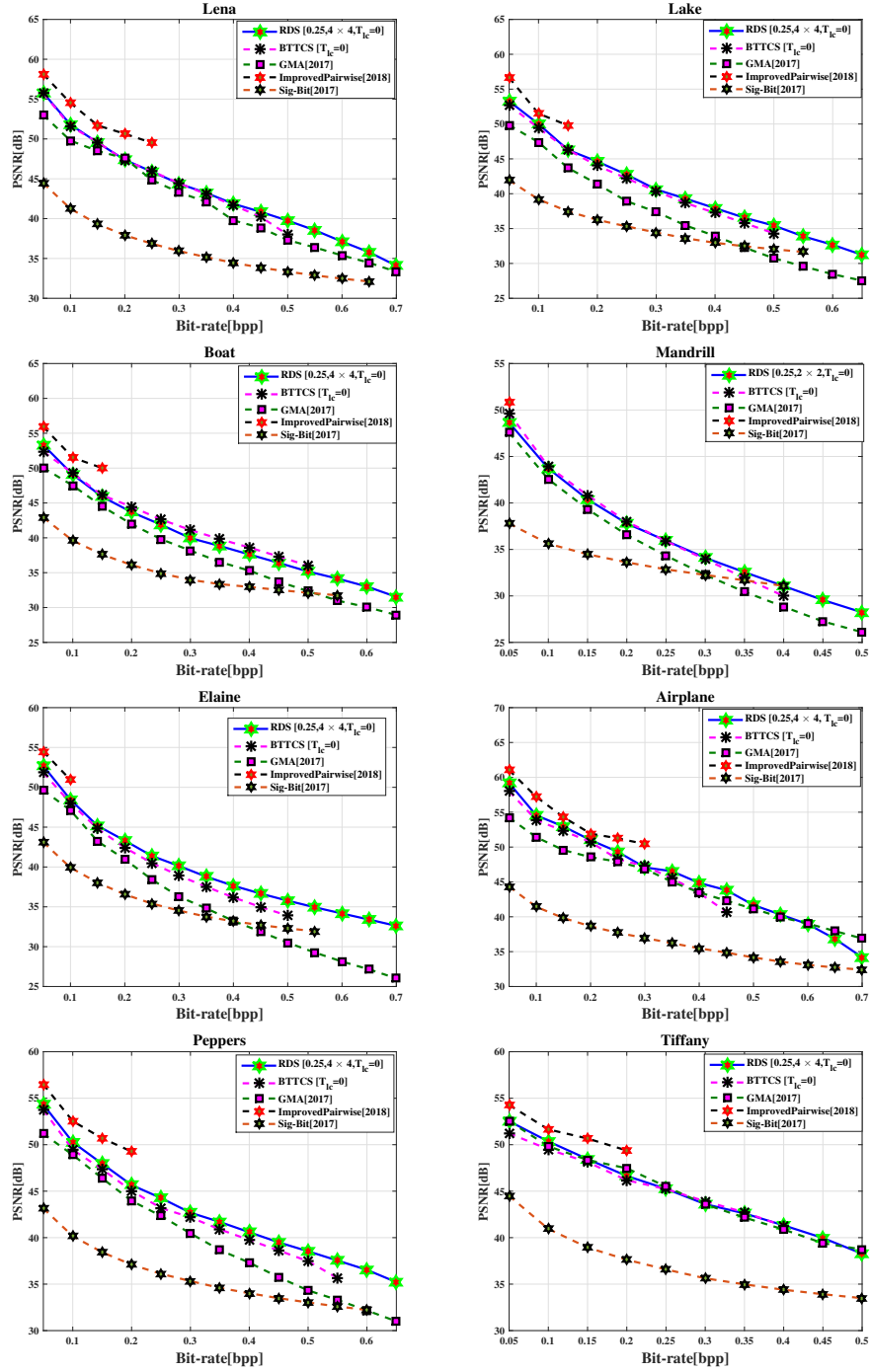


Figure 4.6: Payload size versus PSNR plots for several reversible data hiding techniques

Table 4.4: Minimum, average and maximum of standard deviations of PSNR values over ten executions for each combination of parameter values for various payload sizes

Image	Min	Avg	Max
Lena	0.02	0.03	0.06
Lake	0.03	0.05	0.11
Boat	0.02	0.03	0.06
Mandrill	0.02	0.03	0.05
Elaine	0.01	0.03	0.06
Airplane	0.02	0.04	0.07
Peppers	0.01	0.03	0.09
Tiffany	0.02	0.04	0.12

parameter values (as discussed above) for various payload sizes. Minimum, average and maximum of these standard deviations for each image are tabulated in table 4.4. It can be seen from these simple statistics about the observed standard deviations in PSNR values that there is not much change in the PSNR values across different executions of the proposed technique.

The marked images as per the proposed reversible data hiding technique for various payload size are presented in figure 4.7.

4.5 Summary

In this chapter, a novel prediction error expansion (PEE) based reversible data hiding technique is proposed using the concept of random distribution of reference pixels. In a PEE based reversible data hiding technique, the prediction of a pixel value is an important step. In the proposed technique, a set of reference pixels is used to predict the values in other non-reference pixels. But unlike Delaunay triangulation based method ([119]) and B-tree triangular decomposition based method (Chapter 3), the proposed prediction method distributes the set of reference pixels throughout the image randomly. But the variances of pixel values in the blocks guide this method of distributing the reference pixels. Then,

the value of a non-reference pixel is predicted as weighted median of the values in the nearby reference pixels. The weight being associated with a reference pixel value is inversely proportional to the chess board distance of the reference pixel from the non-reference pixel. This random distribution of reference pixel based prediction is the major contribution of the work in this chapter. Finally, an adaptive prediction error expansion based embedding strategy [132] is utilized for embedding the data into the prediction error of a non-reference pixel. Local complexity of a non-reference pixel is computed as the variance of the values in nearby reference pixels. Either one bit or two bits of data are inserted into the prediction error of a non-reference pixel based on its local complexity.

Block size and percentage of reference pixels among all pixels in an image are two parameters to influence the performance of the proposed reversible data hiding technique. Various values of these parameters are investigated to find a suitable combination of these values. Use of lesser number of reference pixels ensures higher embedding capacity. It can be found that 25% of pixels as reference pixel achieves the best PSNR values (between the cover images and the corresponding marked images) too. Moreover, the better PSNR values are obtained for the block size as either 2×2 or 4×4 . The best suitable block size varies from one image to another. Based on the observed PSNR values, the best combination of values for these two parameters is selected for each image for subsequent experiments.

Local complexity threshold is also varied for the experiments. Higher value of this threshold ensures higher embedding capacity. But the PSNR value decreases with increment of this threshold as 2 bits are embedded in more number of pixels. Hence, single bit embedding ($T_{lc} = 0$) is suitable for small payload size. Adaptive embedding may be adopted for higher payload size. In this chapter, an extensive set of experimental results is reported by varying above parameters of the proposed technique. Finally, the proposed random distribution of reference pixels based reversible data hiding technique outperforms several existing reversible data hiding techniques. The proposed reversible data hiding technique in this chapter marginally outperforms B-tree triangular decomposition based technique (Chapter 3) too for most of the test images.

Like the existing reversible data hiding techniques, the proposed reversible data hiding technique is also fragile. Therefore, robustness of the proposed technique against several attacks on the marked image is not generally studied for these techniques. Unlike robust watermarking schemes, a minor change in the marked pixel value destroys the hidden data in these reversible data hiding techniques. Hence, this enables identification of any tampering of original cover image. Therefore, the proposed reversible data hiding technique can be mainly used for ensuring integrity of the original cover image.



Figure 4.7: Results of the random distribution of reference pixels based reversible data hiding scheme for various test images (row-wise). Columns from left to right: Original image, marked images with 0.1 bpp, 0.2 bpp, 0.3 bpp, and 0.4 bpp.

Chapter 5

Reversible Data Hiding Using Selected Directional Context Based Prediction

Unlike the reference pixel based reversible data hiding techniques in previous two chapters, two novel reversible data hiding techniques using selected directional context based predictors are proposed in this chapter. In these techniques, a pixel value is predicted using pixel values in the neighborhood of the pixel. Both techniques use a combination of linear predictors in a set of selected directions. In the first of these techniques, the directions are selected using the diversities and the averages of pairs of neighboring pixel values in a 3×3 neighborhood in horizontal, vertical, diagonal and anti-diagonal directions. In the second technique, the directions are selected by comparing the gradients across all four directions. A 5×5 neighborhood is used to estimate the gradients in various directions. Moreover, an adaptive prediction error histogram bin shifting method is proposed for embedding either 1 bit or 2 bits of data in the prediction error of a pixel.

The organization of this chapter is mentioned here. The reversible data hiding technique using selected context based prediction with 3×3 neighborhood is proposed in Section 5.1. In another reversible data hiding scheme using selected context based prediction in Section 5.2, an improved gradient based prediction using 5×5 neighborhood is used. This reversible data hiding technique is depicted in Section 5.2. Finally, a summary of the contributions in this chapter is

drawn in Section 5.3.

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

In this section, an eight neighborhood based pixel prediction strategy is proposed. A 3×3 neighborhood is considered for this work. Averages of pair of neighborhood values in horizontal, vertical, diagonal, and anti-diagonal directions provide the predicted values in respective directional context. A novel selection strategy is introduced to select few of these directional contexts based on the diversity between the pair of pixel values in each direction. Average of the predicted values in the selected directional contexts produces the final predicted value. Like the prediction error histogram bin shifting method in [56], data bits are embedded in prediction errors belonging to two adjacent bins of prediction error histogram. As it is evident from previous studies, the peak in a prediction error histogram can be observed in the zero-th bin or any other adjacent bin. Hence, like the work in [56], prediction errors in two histogram bins for prediction error as either 0 or -1 are used for embedding. Questions may be raised as why only these two bins are selected for embedding. Hence, histogram bins for prediction error as either 0 or 1 are also used for embedding as a slight variation in the experiment. Moreover, an adaptive histogram bin shifting is proposed here to increase the embedding capacity of the scheme. Here, either one bit or two bits of data are embedded in the prediction error of a pixel. In summary, the proposed method introduces the following novel concepts : (i) a selected directional context based pixel prediction using pairs of pixels in eight neighborhood of a pixel and (ii) an adaptive prediction error histogram bin shifting as embedding strategy. Experimental results show the superiority of the proposed reversible data hiding technique in comparison with few other existing reversible data hiding techniques. This work is presented in this section in detail.

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

5.1.1 Proposed Selected Context Based Prediction Using Eight Neighborhood

The proposed prediction scheme considers 8-neighborhood of a pixel as it is shown in figure 5.1. Based on directional context, the eight pixels in the neighborhood are divided into four groups containing the pairs of horizontal, vertical, diagonal, and anti-diagonal neighbors. Diversity between the pair of pixels in a group is measured by considering the absolute difference of the pixel values in the pair. Let d_h , d_v , d_d , and d_a denote the diversities of the pixel pairs in horizontal, vertical, diagonal, and anti-diagonal directions, respectively. The diversity computation is given in equation 5.1.

$(m-1, n-1)$	$(m-1, n)$	$(m-1, n+1)$
$(m, n-1)$	(m, n)	$(m, n+1)$
$(m+1, n-1)$	$(m+1, n)$	$(m+1, n+1)$

Figure 5.1: 8-neighborhood of a pixel at coordinate (m, n)

$$\begin{aligned}
 d_h &= |x_{m,n-1} - x_{m,n+1}| \\
 d_v &= |x_{m-1,n} - x_{m+1,n}| \\
 d_d &= |x_{m-1,n-1} - x_{m+1,n+1}| \\
 d_a &= |x_{m-1,n+1} - x_{m+1,n-1}|
 \end{aligned} \tag{5.1}$$

where, in a generalized notation, $x_{i,j}$ denotes the pixel value at coordinate (i, j) . The notation $|\cdot|$ refers to the absolute value of its argument.

Moreover, integer average of the pair of neighboring pixel values in each direction (horizontal, vertical, diagonal, and anti diagonal) is computed as it is stated in equation 5.2.

$$\begin{aligned}
 a_h &= \lfloor \frac{x_{m,n-1} + x_{m,n+1}}{2} \rfloor \\
 a_v &= \lfloor \frac{x_{m-1,n} + x_{m+1,n}}{2} \rfloor \\
 a_d &= \lfloor \frac{x_{m-1,n-1} + x_{m+1,n+1}}{2} \rfloor \\
 a_a &= \lfloor \frac{x_{m-1,n+1} + x_{m+1,n-1}}{2} \rfloor
 \end{aligned} \tag{5.2}$$

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

where a_h , a_v , a_d , and a_a denote the computed integer average values in horizontal, vertical, diagonal, and anti-diagonal directions, respectively. The notation $\lfloor \cdot \rfloor$ refers to the highest integer which is smaller than its argument.

Basically, the average of the pair of neighboring pixel values in a particular direction predicts the center pixel value in the same directional context. Similar idea on average of pair of directional neighbors for prediction can also be found in [76]. But only the horizontal and vertical directions are considered in [76]. In the proposed approach, two other directions (diagonal and anti-diagonal) are also considered.

The proposed approach selects a few of these directional contexts based on the homogeneity (or less diversity) of neighborhood pixels in these directions. Hence, the directional contexts having the least diversity is considered for predicting the current pixel. Let four diversity values in equation 5.1 be sorted in ascending order and let these be denoted as d_1, d_2, d_3 and d_4 (while d_1 is the smallest of these four values). Moreover, the integer averages in these four directions (equation 5.2) are sorted in ascending order of the diversities in respective directions and let the sorted values be a_1, a_2, a_3 and a_4 . Here, the integer average value a_i corresponds to the direction having diversity d_i . Basically, these average values act as a predicted value in their respective directions. At first, the predicted value a_1 according to the least diverse group (with diversity value d_1) is considered to predict the central pixel value. Additionally, the predictions in other directions are considered, only if the predicted (integer average) values in those directions are also close enough to the value a_1 . A threshold T decides the closeness of these average values to the value a_1 . The value of T is assumed to be 1 for the reported experiments. To focus on the directions of less diverse pixel pairs, closeness among these average values is tested iteratively starting with the second least diverse group. This complete algorithm is mentioned in Algorithm 5.1, where the iteration has been broken down using if-else constructs for three other groups (apart from the least diverse group). Ultimately, if predicted (i.e., integer average) values of all four groups are similar enough, then integer average of all four predicted values (i.e., integer averages for individual groups) predicts the center pixel value.

In any prediction error based reversible data hiding technique, the predicted pixel value must be same during embedding and extractions phases. As per

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

Algorithm 5.1 Predicting the center pixel value using neighbors in selected directional context

```

1: Compute the diversities in various directions [ $d_h, d_v, d_d$ , and  $d_a$ ] using equation 5.1.
2: Compute the integer averages of a pair of neighbors in various directions [ $a_h, a_v, a_d$ , and  $a_a$ ] using equation 5.2.
3: Sort the diversity values in ascending order and represent them as [ $d_1, d_2, d_3$ , and  $d_4$ ].
4: Sort the integer average values in ascending order of the diversities in respective directions and store them as [ $a_1, a_2, a_3$ , and  $a_4$ ].
5: // Difference ( $d$ ) between the integer averages in two directions having the two least diversity values is computed.
6:  $d = |a_1 - a_2|$ 
7: if  $d \geq T$  then
8:   // The direction with the second least diversity value is not considered.
9:   // Estimated predicted value is the integer average in the direction having the least diversity value.
10:   $x' = a_1$ 
11: else
12:   // Difference between the integer averages in two directions having the least and the third least diversity values is computed.
13:    $d = |a_1 - a_3|$ 
14:   if  $d \geq T$  then
15:     // The direction having the third least diversity value is not considered, only the directions having the least two diversity values are considered.
16:     // Estimated value is the integer average of respective integer averages in the directions with the least two diversity values.
17:      $x' = \lfloor \frac{a_1 + a_2}{2} \rfloor$ 
18:   else
19:     // Difference between the integer averages in two directions having the least and the fourth least diversity values is computed.
20:      $d = |a_1 - a_4|$ 
21:     if  $d \geq T$  then
22:       // The direction having the fourth least diversity value is not considered, only the directions having the least three diversity values are considered.
23:       // Estimated value is the integer average of respective integer averages in the directions with the least three diversity values.
24:        $x' = \lfloor \frac{a_1 + a_2 + a_3}{3} \rfloor$ 
25:     else
26:       // Estimated value is the integer average of respective integer averages in all directions.
27:        $x' = \lfloor \frac{a_1 + a_2 + a_3 + a_4}{4} \rfloor$ 
28:     end if
29:   end if
30: end if

```

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

above discussion, the predicted value depends on the 8 values of neighboring pixels. Hence, the same 8 values must be available during extraction too. But as embedding data bits in a pixel changes the pixel values, original cover pixel values for all 8 neighbors cannot be availed during extraction. Therefore, following strategy is adopted to maintain same set of 8 values for these neighboring pixels during embedding and extraction. Embedding is carried out in raster scan order of the pixels in an image. Hence, for a current pixel at coordinate (m, n) , the four neighboring pixels $(m-1, n-1)$, $(m-1, n)$, $(m-1, n+1)$, and $(m, n-1)$ already contain the marked values. Remaining four neighboring pixels at locations $(m, n+1)$, $(m+1, n-1)$, $(m+1, n)$ and $(m+1, n+1)$ are not visited till then. Hence, these four pixels contain the original values. These context values which are available during prediction of the pixel value at coordinate (m, n) are shown in figure 5.2. Similarly, the extraction of data bits and the restoration of the original pixel values are carried out in the opposite of raster scan order. Hence, during extraction phase, the pixels at coordinates $(m, n+1)$, $(m+1, n-1)$, $(m+1, n)$ and $(m+1, n+1)$ have been already traversed and their original values have been restored back. The pixels at coordinates $(m-1, n-1)$, $(m-1, n)$, $(m-1, n+1)$, and $(m, n-1)$ still contain the marked values. Thus, same set of context values (figure 5.2) is available to the prediction algorithm during both embedding and extraction.

$x'_{m-1, n-1}$	$x'_{m-1, n}$	$x'_{m-1, n+1}$
$x'_{m, n-1}$		$x_{m, n+1}$
$x_{m+1, n-1}$	$x_{m+1, n}$	$x_{m+1, n+1}$

Figure 5.2: Context pixel values of 8-neighbors of pixel (m, n) during prediction

5.1.2 Embedding Using Adaptive Histogram Bin Shifting

Unlike the reversible data hiding techniques in Chapter 3 and Chapter 4, the prediction error expansion based embedding strategy (Section 3.2) cannot be adopted for this work. An overflow/underflow map is needed in the prediction error expansion strategy in order to avoid overflow/underflow conditions during

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

embedding. Construction of an overflow/underflow map in this strategy requires the predicted values for every pixel in the image. In the previous two techniques in Chapter 3 and Chapter 4, the predicted values for every non-reference pixel are computed apriori using the reference pixel values. Embedding is not carried out in these reference pixels. Hence, the prediction error expansion based strategy can be used for the reversible data hiding techniques in the previous chapters. But in the present selected directional context based reversible data hiding technique, the prediction of a pixel value and embedding in the pixel are carried out simultaneously. This is because the proposed prediction scheme in Section 5.1.1 uses 8-neighborhood of a pixel for predicting its value. Few of these neighborhood pixels contain marked values (figure 5.2). Hence, the predicted values of all the pixels in the image cannot be known before the actual embedding starts. Therefore, the overflow/underflow map cannot be obtained in the manner in which it was done in the previous two reference pixel based techniques. Hence, the prediction error expansion strategy (Section 3.2) cannot be adopted here for embedding. The required overflow/underflow map for the present technique is derived in a different way, which is mentioned in the due course of the discussion.

The proposed embedding scheme, in this section, is based on a concept of adaptive prediction error histogram bin shifting. Hence, at first, a pixel value is predicted using the proposed approach in Section 5.1.1. Subsequently, prediction error for a pixel is computed as the difference between the cover image pixel value and its predicted value using equation 5.3.

$$PE = x - x' \quad (5.3)$$

According to the prediction error histogram bin shifting method in [56], a single data bit is embedded in the prediction error of a pixel belonging to two adjacent histogram bins having prediction error as either 0 or -1. For the pixels in the remaining histogram bins, the prediction errors are shifted in appropriate direction. The proposed adaptive scheme extends this idea to embed either 1 bit or 2 bits of data in the prediction error of a pixel based on an estimated local complexity of the pixel. Each component of the proposed adaptive prediction error histogram bin shifting method is presented in this section.

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

5.1.2.1 Estimation of Local Complexity

Local complexity (lc) of the current pixel approximates the roughness of the pixel. The estimation of the local complexity of the current pixel is computed as the variance of four pixel values at bottom and right neighbors ($x_{m+1,n-1}$, $x_{m+1,n}$, $x_{m+1,n+1}$ and $x_{m,n+1}$). As these pixels maintain the original values during embedding and extraction processes, only these pixel values are used to estimate the roughness around the center pixel at coordinate (m,n) . The top and left neighboring pixels have marked (i.e., modified) values. Therefore, the four neighboring pixels at the top and at the left are not used to estimate local complexity.

5.1.2.2 Adaptive Embedding

As a generalization of the prediction error histogram bin modification in [56], in this section, two consecutive histogram bins are considered for embedding where prediction errors are either PE_l or PE_r ($PE_r = PE_l + 1$). The peak bin in a prediction error histogram normally corresponds to the zero-th bin or a nearby bin. Hence, unlike [56], separate experiments are carried out by embedding in two pairs of histogram bins: (i) histogram bins corresponding to prediction error as either -1 and 0 ($PE_l = -1$ and $PE_r = 0$), and (ii) histogram bins corresponding to prediction error as either 0 and 1 ($PE_l = 0$ and $PE_r = 1$). The rest of the discussion refers the general representation of PE_l and PE_r , where $PE_r = PE_l + 1$. The prediction errors in the bins having prediction error other than PE_l and PE_r are shifted to identify the histogram bins where embedding data is located.

Adaptive embedding strategy provides a flexibility of either embedding one bit or two bits of data in the prediction error of a pixel. It is discussed below.

Embedding Two Bits in a Pixel

Local complexity (lc) of a pixel is compared against a threshold value (T_{lc}) to determine whether the pixel is in smooth region or rough region. If the local complexity value (lc) is less than the threshold T_{lc} , then the pixel is assumed to be in a smooth region. Two bits of data are embedded in the prediction error of such a pixel, if the prediction error PE of the pixel is equal to either PE_l or PE_r .

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

(equation 5.4).

$$x^w = \begin{cases} x' + PE - w & , \quad \text{if } PE = PE_l \\ x' + PE + w & , \quad \text{if } PE = PE_r \end{cases} \quad (5.4)$$

where $w \in \{0, 1, 2, 3\}$ indicates two bits of data. x and x^w represent the pixel values before and after embedding the data, respectively. To separate the bins where data bits are embedded from all other bins, the prediction errors in the remaining bins are shifted. Otherwise, extraction of the data bits and restoration of the pixel values are not possible. Shifting of the predictions errors follows equation 5.5.

$$x^w = \begin{cases} x' + PE - 3 & , \quad \text{if } PE < PE_l \\ x' + PE + 3 & , \quad \text{if } PE > PE_r \end{cases} \quad (5.5)$$

Embedding Single Bit in a Pixel

If local complexity value (lc) of a pixel is greater than or equal to the threshold T_{lc} , then the pixel is assumed to be in a rough region. One bit of data is embedded in the prediction error of such a pixel, if the prediction error PE of the pixel is equal to either PE_l or PE_r (equation 5.6).

$$x^w = \begin{cases} x' + PE - w & , \quad \text{if } PE = PE_l \\ x' + PE + w & , \quad \text{if } PE = PE_r \end{cases} \quad (5.6)$$

where $w \in \{0, 1\}$ indicates one bit of data. x and x^w denote the pixel values before and after embedding the data, respectively.

To separate the bins where data bits are embedded from all other bins, the prediction errors in the remaining bins are shifted. Otherwise, extraction of the data bits and restoration of the pixel values are not possible. Shifting of the prediction errors follows equation 5.7.

$$x^w = \begin{cases} x' + PE - 1 & , \quad \text{if } PE < PE_l \\ x' + PE + 1 & , \quad \text{if } PE > PE_r \end{cases} \quad (5.7)$$

5.1.2.3 Auxiliary Information

According to the proposed adaptive prediction error histogram bin shifting method, the maximum modification to a pixel value is ± 3 . Hence, before applying the proposed reversible data hiding technique, the image pixel values 255,

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

254, and 253 are converted to 252, 251 and 250, respectively, to avoid overflow condition (i.e., pixel value after modification should not be higher than 255). Similarly, pixel values 0, 1, and 2 are converted to 3, 4, and 5, respectively, to avoid underflow condition (i.e., pixel value after modification should not be less than 0). A binary location map is constructed to specify these overflow and underflow pixel locations in the image. The modified locations are marked with 1 in the binary location map, which is also known as overflow/underflow map. This overflow/underflow map is compressed using a lossless image compression technique. The proposed work uses the arithmetic coding algorithm for lossless compression. This compressed location map is stored as part of the auxiliary information, which is useful for complete extraction of the data as well as accurate restoration of the image pixels. Auxiliary information also contains few additional information. Here, the complete set of information of auxiliary information is stated: (i) coordinates of the last embedded pixel, (ii) whether 1 bit or 2 bits are embedded in the last embedded pixel, (iii) the value of local complexity threshold T_{lc} , (iv) whether there is at least one pixel causing overflow/underflow conditions, (v) length of the compressed overflow/underflow map, and (vi) the compressed overflow/underflow location map. The items in (v) and (vi) are required only at the occurrence of overflow/underflow condition.

The length of the auxiliary information can be estimated by considering summation of number of bits being used to represent each component of the auxiliary information. (i) The number of bits representing the coordinate of last embedded pixel is 18 for an image of size 512×512 (as $2 \times \log_2(512) = 18$). (ii) 1 bit is used to represent whether 1 or 2 bits are embedded in the last embedded pixel. (iii) 5 bits are used to represent the local complexity threshold T_{lc} . (iv) 1 bit is used to indicate whether overflow/underflow may arise for any of the pixels in the image. (v) Length L (in bits) of the compressed overflow/underflow map is represented using 18 bits for an image of size 512×512 (as $2 \times \log_2(512) = 18$). (vi) The compressed location map length is L bits. Thus, the total number of bits being used to encode the auxiliary information is: $(18 + 1 + 5 + 1 + 18 + L) = 43 + L$.

At first, the auxiliary information is embedded into the pixels of first few rows (in raster scan sequence) in the cover image using LSB (least significant bit) replacement technique. Then, the LSBs of those first few pixels are appended

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

at the end of the data stream to be embedded. But as this step is carried out before the actual embedding of payload, the position of the last pixel (part (i) of auxiliary information), where embedding is carried out, is not known apriori. Hence, the LSBs for the first 18 pixels (in raster scan sequence) are reserved for storing the coordinate of the last embedded pixel. The number of bits representing the coordinate of last embedded pixel is 18 for an image of size 512×512 (as $2 \times \log_2(512) = 18$). The coordinate of the last embedded pixel will be stored at these reserved bits after the embedding of the payload.

It is to be noted that the auxiliary information may occupy the first few rows in the image. Hence, those few rows are not used to embed the payload. Embedding of the payload starts from the immediate next row using a raster scan order traversal of the pixels. Embedding process is described in Section 5.1.2.2. Once the embedding of the entire payload is completed, the position of the last embedded pixel is stored as part (i) of auxiliary information at the bits which were reserved for storing this information.

5.1.3 Extraction

The secret data is extracted and the original cover image pixel values are restored from the pixel values of the marked image. Knowledge about the auxiliary information is necessity in this phase. The auxiliary information is derived from the LSBs of the pixels in the first few rows of the marked image. As part of auxiliary information, the coordinates of the last embedded pixel is known. Moreover, it is also known from auxiliary information whether 1 bit or 2 bits of payload are embedded in the last embedded pixel. From this pixel onward, the pixels are traversed in the reverse of raster scan order for simultaneous extraction of payload bits and restoration of the original pixel values. Visiting these pixels in the reverse order (of embedding) is necessity to achieve the same context pixel values around a pixel (m, n) . Because at the time of visiting this pixel, four other pixels in the neighborhood $((m, n+1), (m+1, n-1), (m+1, n)$ and $(m+1, n+1))$ have been already traversed and the original pixel values in those locations have been restored. On the contrary, four other pixels in the neighborhood (at coordinates $(m-1, n-1), (m-1, n), (m-1, n+1)$ and $(m, n-1))$ have not been visited.

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

Those pixels contain the marked values. Hence, the same context as embedding (figure 5.2) is achieved. Obtaining the same context pixel values ensures that the predicted value during extraction remains same as the predicted value during embedding.

During extraction phase too, a pixel value is predicted using the proposed method of selected context based prediction using 8-neighborhood (Section 5.1.1). Subsequently, the prediction error (PE^w) is calculated by considering the difference between the marked pixel value x^w and its predicted value x' using equation 5.8.

$$PE^w = x^w - x' \quad (5.8)$$

Similarly, the local complexity value of a pixel is calculated using the restored pixel values in the neighborhood as mentioned in Section 5.1.2.1. Hence, the obtained local complexity value is also same as the local complexity value during embedding. Then, it can be known whether 1 bit or 2 bits of payload need to be extracted from the current pixel (m, n) by comparing the obtained local complexity value with the local complexity threshold T_{lc} . It is to be noted that the value of the local complexity threshold T_{lc} is known from auxiliary information.

5.1.3.1 Extraction of Two Bits

If local complexity of the current pixel is less than T_{lc} and prediction error falls within the range $[PE_l-3, PE_r+3]$, then two bits of data (w) are extracted using equation 5.9.

$$w = \begin{cases} 0 = 00 & \text{if } PPE^w = PE_l \text{ or } PE_r \\ 1 = 01 & \text{if } PE^w = PE_l - 1 \text{ or } PE_r + 1 \\ 2 = 10 & \text{if } PE^w = PE_l - 2 \text{ or } PE_r + 2 \\ 3 = 11 & \text{if } PE^w = PE_l - 3 \text{ or } PE_r + 3 \end{cases} \quad (5.9)$$

Subsequently, the original pixel value is restored using equation 5.10.

$$x = \begin{cases} x' + PE^w + w, & \text{if } PE^w \in [PE_l - 3, PE_l] \\ x' + PE^w - w, & \text{if } PE^w \in [PE_r, PE_r + 3] \end{cases} \quad (5.10)$$

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

If local complexity of a pixel is less than T_{lc} and prediction error PE^w is either less than $PE_l - 3$ or greater than $PE_r + 3$, then shifting back of the marked pixel value to original value is carried out using equation 5.11.

$$x = \begin{cases} x' + PE^w + 3 & , \quad \text{if} \quad PE^w < PE_l - 3 \\ x' + PE^w - 3 & , \quad \text{if} \quad PE^w > PE_r + 3 \end{cases} \quad (5.11)$$

5.1.3.2 Extraction of a Single Bit

If local complexity of the current pixel is either greater than or equal to T_{lc} and prediction error falls within the range $[PE_l - 1, PE_r + 1]$, then one bit of data (w) is extracted using equation 5.12.

$$w = \begin{cases} 0, & \text{if} \quad PE^w = PE_l \text{ or } PE_r \\ 1, & \text{if} \quad PE^w = PE_l - 1 \text{ or } PE_r + 1 \end{cases} \quad (5.12)$$

Subsequently, the original pixel value is restored using equation 5.13.

$$x = \begin{cases} x' + PE^w + w, & \text{if} \quad PE^w \in [PE_l - 1, PE_l] \\ x' + PE^w - w, & \text{if} \quad PE^w \in [PE_r, PE_r + 1] \end{cases} \quad (5.13)$$

If local complexity of a pixel is either greater than or equal to T_{lc} and prediction error PE^w is either less than $PE_l - 1$ or greater than $PE_r + 1$, then shifting back of the marked pixel value to original value is carried out using equation 5.14.

$$x = \begin{cases} x' + PE^w + 1 & , \quad \text{if} \quad PE^w < PE_l - 1 \\ x' + PE^w - 1 & , \quad \text{if} \quad PE^w > PE_r + 1 \end{cases} \quad (5.14)$$

5.1.4 Experimental Results

Experimental results involving the proposed selected context based prediction scheme and adaptive prediction error histogram bin shifting are reported in this section. Performance of the proposed scheme has been tested using the standard test images as shown in figure 1.4. Embedding payload bits in the peak bins of the histogram is desired to achieve higher embedding capacity. Normally, in a prediction error histogram of an image, peak bins are observed in the zero-th bin and its nearby bins. Therefore, separate experiments are carried out by embedding

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

Table 5.1: Achievable embedding capacities with histogram bin pairs $(-1, 0)$ and $(0, 1)$ using single bit embedding ($T_{lc} = 0$) for various test images.

Pair of bins	$(-1, 0)$			$(0, 1)$	
Image	T_{lc}	Embedding Capacity (Bits)	PSNR	Embedding Capacity (Bits)	PSNR
Lena	0	58389	48.68	58394	48.68
Lake	0	36798	48.48	37182	48.49
Boat	0	36817	48.49	37197	48.49
Mandrill	0	19236	48.34	19464	48.34
Elaine	0	33826	48.46	34165	48.46
Airplane	0	83189	48.92	81714	48.91
Peppers	0	41801	48.53	42191	48.54
Tiffany	0	48468	48.85	48629	48.80

Table 5.2: Achievable embedding capacities with histogram bin pairs $(-1, 0)$ and $(0, 1)$ using adaptive embedding ($T_{lc} = 16$) for various test images.

Pair of bins	$(-1, 0)$			$(0, 1)$	
Image	T_{lc}	Embedding Capacity (Bits)	PSNR	Embedding Capacity (Bits)	PSNR
Lena	16	81057	41.31	81383	41.31
Lake	16	49663	42.81	50344	42.82
Boat	16	47939	42.67	48785	42.68
Mandrill	16	22339	45.47	22391	45.47
Elaine	16	43671	43.26	43913	43.27
Airplane	16	118340	40.98	116266	40.96
Peppers	16	59232	41.56	59188	41.60
Tiffany	16	64515	41.30	64756	41.23

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

in two pairs of histogram bins: (i) histogram bins corresponding to prediction error as -1 and 0 ($PE_l = -1$ and $PE_r = 0$), and (ii) histogram bins corresponding to prediction error as 0 and 1 ($PE_l = 0$ and $PE_r = 1$). Embedding capacity (in terms of number bits) are mentioned in table 5.1 for the above two cases for each image. These results correspond to single bit embedding in a pixel ($T_{lc} = 0$). It can be seen in table 5.1 that the embedding capacities are almost similar in these two experimental cases. The PSNR values (at the highest embedding capacity for each image) between the cover images and their corresponding marked images are also similar in these two cases. Similarly, embedding capacities in above two experimental cases are reported in table 5.2 for adaptive embedding (with $T_{lc} = 16$) for each image. The embedding capacities as well as corresponding PSNR values between the cover images and their corresponding marked images are observed to be similar for adaptive embedding too irrespective of the selection of histogram bin pairs as either (-1, 0) or (0, 1). Though embedding capacities for adaptive embedding ($T_{lc} = 16$) are higher than those with single embedding ($T_{lc} = 0$). It justifies the use of adaptive embedding (either 1 bit or 2 bits in a pixel) over the use of single bit embedding in a pixel. But significant difference in performance is not observed between the histogram bin pairs being (-1, 0) or (0, 1). Hence, any one pair among these two pairs of histogram bins can be used for embedding.

Experiments are also carried out by varying the value of local complexity threshold T_{lc} as 0,1,2,4,8 and 16. The value of T_{lc} as 0 indicates that a single bit is embedded in a pixel. Increase of the value of T_{lc} implies that increasingly more pixels are considered for 2-bit embedding. These are the cases for adaptive embedding (either 1 bit or 2 bits in a pixel) depending on the local complexity of the pixel. PSNR values between the cover images and the corresponding marked images are plotted in figure 5.3 with various payload sizes for the test images. It can be seen from these plots that the PSNR values are the highest in the case of $T_{lc} = 0$, i.e., single embedding. With increment in the value of T_{lc} , the PSNR values decrease. But higher embedding capacity is achieved with higher value of T_{lc} . These plots in figure 5.3 correspond to the experiments using (-1, 0) histogram bin pairs. Similar trend in decrease of PSNR values and increase of embedding capacities with increased local complexity threshold value T_{lc} can be observed using (0, 1) histogram bin pairs too.

5.1 Reversible Data Hiding with Selected Directional Context Based Prediction Using Eight Neighborhood

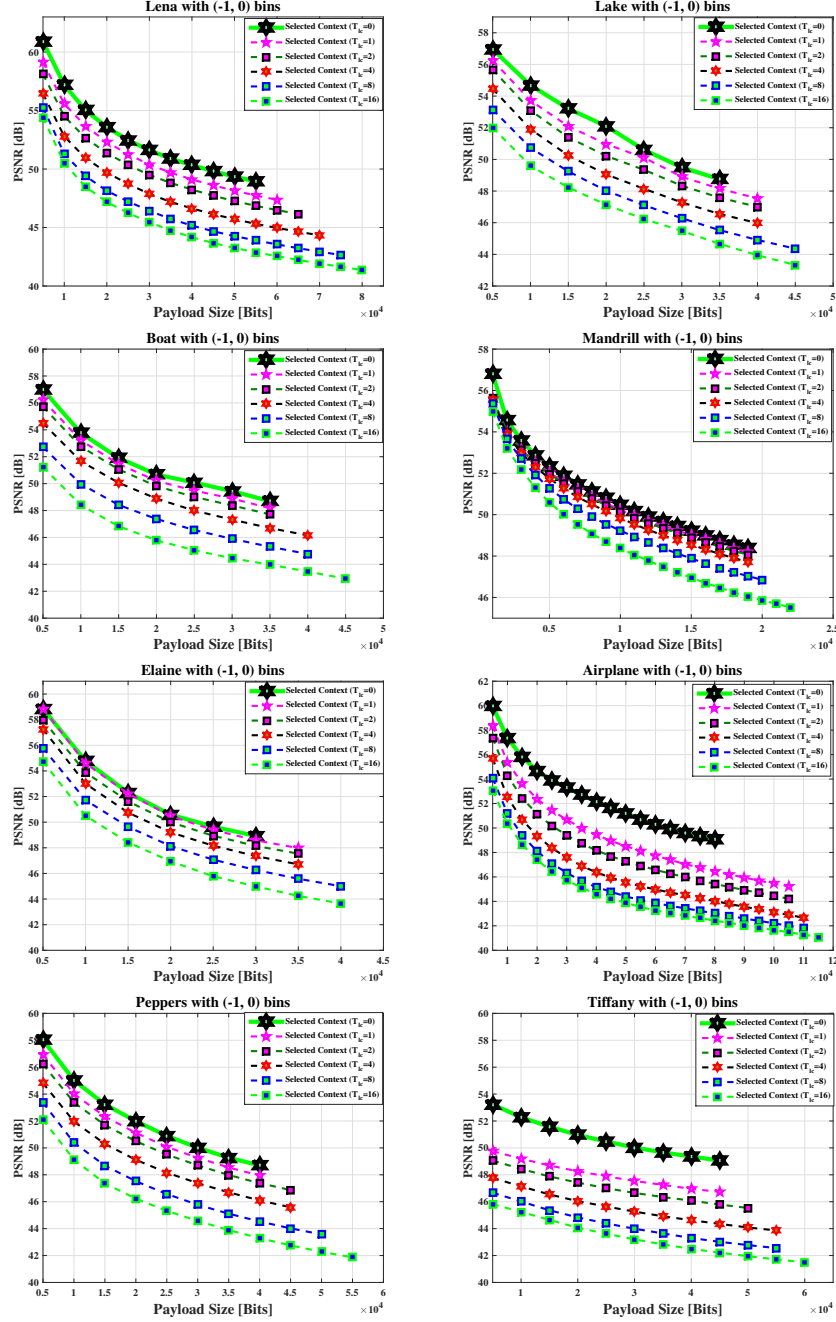


Figure 5.3: Payload size versus PSNR plots using various values of local complexity threshold T_{lc} for various images.

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation

The performance of the proposed technique is compared against several existing techniques like improved MED and improved SGAP [58], rhombus average [131], EGBSW [25], Delaunay triangulation [119], higher-significant bit-plane expansion [49], B-tree triangular decomposition (Chapter 3) and random distribution of reference pixels (Chapter 4) based reversible data hiding. PSNR values between the cover images and the corresponding marked images for various payload sizes are shown in figure 5.4 for all the comparing methods. It can be observed that the proposed method outperforms the existing comparing methods. It can also be observed that the performance of the proposed method in both experimental cases - (i) histogram bins corresponding to prediction error as -1 and 0 ($PE_l = -1$ and $PE_r = 0$), and (ii) histogram bins corresponding to prediction error as 0 and 1 ($PE_l = 0$ and $PE_r = 1$) - are similar. In these plots (figure 5.4), the legend 'Selected Context (0)' refers to the proposed selected context based prediction with adaptive histogram bin shifting using left bin PE_l as 0 and right bin PE_r as 1. Similarly, the legend 'Selected Context (-1)' refers to the proposed selected context based prediction with adaptive histogram bin shifting using left bin PE_l as -1 and right bin PE_r as 0.

The marked images as per the proposed reversible data hiding technique for various payload sizes are presented in figure 5.5.

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation

Unlike the prediction strategy in the previous section, this section adopts a gradient based pixel prediction scheme. Several gradient based prediction schemes are available in literature. One of the existing gradient based prediction schemes, namely Extended Gradient Based Selective Weighting (EGBSW) [25]), uses a 4×4 neighborhood for estimating gradients in horizontal, vertical, diagonal, and anti-diagonal directions. The considered neighborhood is asymmetric with respect to the location of the pixel being predicted. In the proposed work, a larger neighborhood of 5×5 is considered for estimating these gradients. The objective

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation

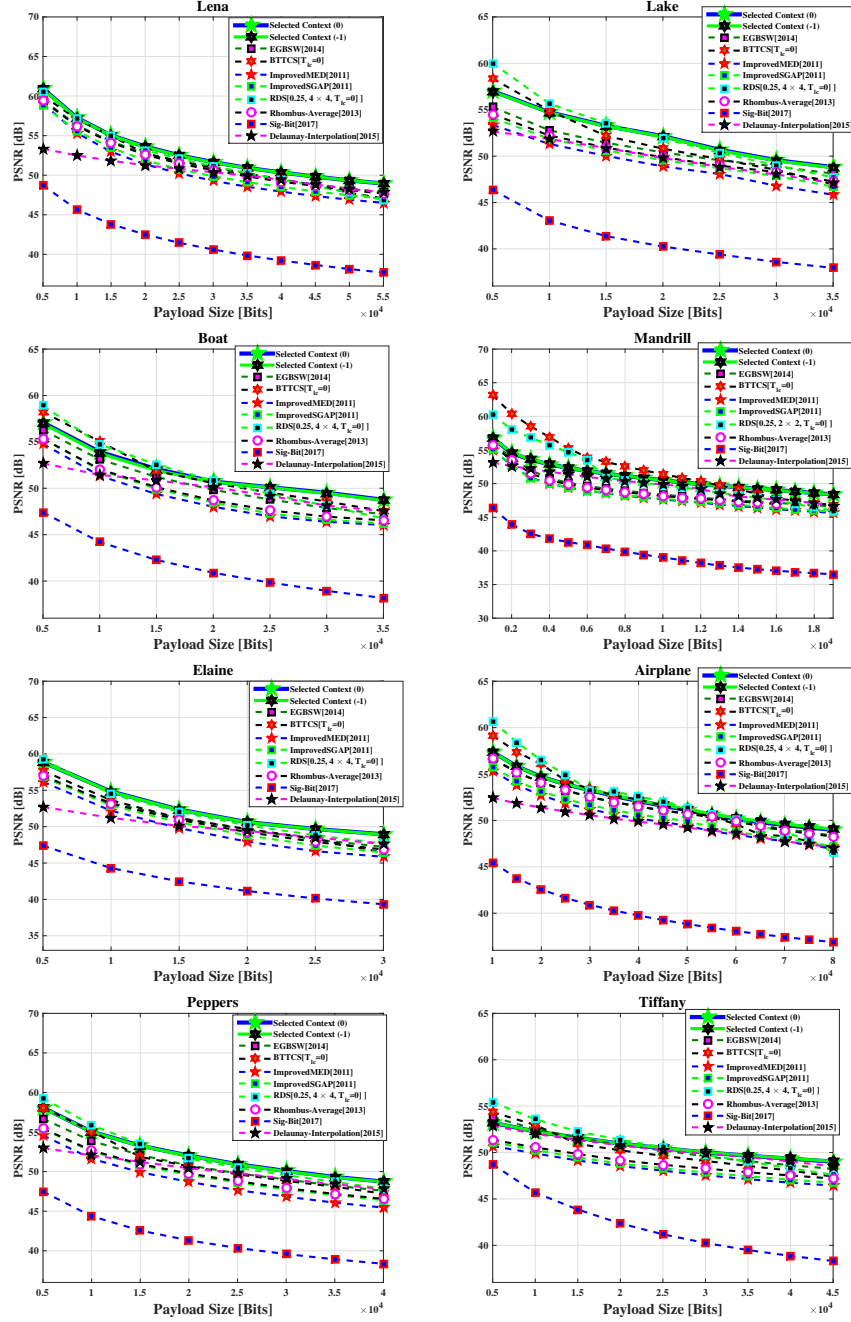


Figure 5.4: Comparison of PSNR values between the cover images and the marked images with various payload sizes among several reversible data hiding techniques

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation



Figure 5.5: Results of the selected directional context using 8-neighborhood based reversible data hiding technique for various test images (row-wise). Columns from left to right: Original image, marked images with 5000 bits, 10000 bits, and 15000 bits.

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation

of the proposed work is to explore whether a larger neighborhood captures better contextual information for gradient estimation and hence, a better pixel prediction is achieved. Moreover, an adaptive prediction error histogram bin shifting method is considered here for embedding data bits. Traditional prediction error histogram bin shifting method [56] embeds one bit of data in the prediction error of a pixel. The proposed adaptive method embeds either 1 bit or 2 bits of data in the prediction error of a pixel based on local complexity (i.e., roughness) around a pixel. At first, the existing EGBSW predictor is described here, as it provides the basis of the proposed improved gradient based predictor. Then, the proposed technique along with the experimental results are presented in this section.

5.2.1 Extended Gradient Based Selective Weighting (EGBSW) Predictor

The extended gradient based selective weighting (EGBSW) predictor was introduced in [25]. This section narrates this existing EGBSW predictor. This predictor estimates the pixel value based on the context of 15 pixels in a 4×4 neighborhood (figure 5.6).

<i>NW</i>	<i>N</i>	<i>NE</i>	<i>NEE</i>
<i>W</i>	<i>x(m,n)</i>	<i>E</i>	<i>EE</i>
<i>SW</i>	<i>S</i>	<i>SE</i>	<i>SEE</i>
<i>SSW</i>	<i>SS</i>	<i>SSE</i>	<i>SSEE</i>

Figure 5.6: 4×4 neighborhood of pixel (m, n) in EGBSW

At first, four different predicted values x'_h , x'_v , x'_d and x'_a are obtained using linear prediction in each of horizontal, vertical, diagonal and anti-diagonal directions, respectively. These linear predictors with respect to their directions are computed as following:

$$x'_h = \lfloor \frac{W + E}{2} + 0.5 \rfloor \quad (5.15)$$

$$x'_v = \lfloor \frac{N + S}{2} + 0.5 \rfloor \quad (5.16)$$

$$x'_d = \lfloor \frac{2(NW + SE) + N + S + W + E}{8} + 0.5 \rfloor \quad (5.17)$$

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation

$$x'_a = \lfloor \frac{2(NE + SW) + N + S + W + E}{8} + 0.5 \rfloor. \quad (5.18)$$

where $\lfloor \cdot \rfloor$ operator returns the highest integer which is less than its argument.

Next, a gradient based strategy selects two of these four directions. Weighted average of the two predicted values in selected directions is considered as the final predicted value for the pixel.

In EGBSW, the 4×4 pixel neighborhood is used to compute the gradients in each of horizontal, vertical, diagonal, and anti-diagonal directions. The gradient in each direction is estimated as an average of weighted summation of differences of neighboring pixel values in corresponding direction. Let the weighted summations of differences of neighboring pixel values in horizontal, vertical, diagonal, and anti-diagonal directions be represented as D_h , D_v , D_d , and D_a , respectively. These values are estimated as follows:

$$D_h = 2|W - E| + 2|E - EE| + |NW - N| + |N - NE| + |NE - NEE| + |SW - S| + |S - SE| + |SE - SEE| \quad (5.19)$$

$$D_v = 2|N - S| + 2|S - SS| + |NW - W| + |W - SW| + |SW - SSW| + |NE - E| + |E - SE| + |SE - SSE| \quad (5.20)$$

$$D_d = 2|NW - SE| + 2|SE - SSE| + |W - S| + |S - SSE| + |N - E| + |E - SEE| + |SW - SS| + |NE - EE| \quad (5.21)$$

$$D_a = 2|NE - SW| + |W - N| + |S - E| + |SS - SE| + |SE - EE| + |SSE - SEE|. \quad (5.22)$$

where $|\cdot|$ operator returns the absolute value of its argument.

The gradients g_h , g_v , g_d and g_a in these four directions are computed from the above summations of differences in respective directions using equation 5.23.

$$g_h = \lfloor \frac{D_h}{10} + 0.5 \rfloor, g_v = \lfloor \frac{D_v}{10} + 0.5 \rfloor \\ g_d = \lfloor \frac{D_d}{10} + 0.5 \rfloor, g_a = \lfloor \frac{D_a}{7} + 0.5 \rfloor \quad (5.23)$$

The directions with the two smallest gradient magnitudes are considered for final prediction of the pixel. Let g_0 and g_1 be the two smallest gradient magnitudes. Let corresponding predicted values in respective directions be x'_0 and x'_1 . Then the predicted value is finally computed as:

$$x' = \lfloor \frac{g_0 x'_1 + g_1 x'_0}{g_0 + g_1} + 0.5 \rfloor \quad (5.24)$$

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation

5.2.2 Proposed Gradient Based Predictor Using 5×5 Neighborhood

The existing EGBSW predictor [25] uses a 4×4 neighborhood. The reported work, in this section, investigates the effectiveness of a 5×5 neighborhood based gradient estimation for a better prediction, and hence, a better reversible data hiding scheme, through capturing more information about the neighborhood. The proposed predictor extends the concept of EGBSW predictor for a context of 5×5 neighborhood as shown in figure 5.7.

<i>NW</i>	<i>N</i>	<i>NE</i>	<i>NEE</i>	<i>NEEE</i>
<i>W</i>	$\mathbf{x(m,n)}$	<i>E</i>	<i>EE</i>	<i>EEE</i>
<i>SW</i>	<i>S</i>	<i>SE</i>	<i>SEE</i>	<i>SEEE</i>
<i>SSW</i>	<i>SS</i>	<i>SSE</i>	<i>SSEE</i>	<i>SSEEE</i>
<i>SSSW</i>	<i>SSS</i>	<i>SSSE</i>	<i>SSSEE</i>	<i>SSSEEE</i>

Figure 5.7: 5×5 neighborhood of pixel (m, n) as being considered in the proposed predictor

Like the EGBSW predictor [25], the proposed approach also computes four predicted values x'_h , x'_v , x'_d and x'_a using linear prediction in each of horizontal, vertical, diagonal and anti-diagonal directions, respectively. These linear predictors with respect to their directions are computed using equations 5.15, 5.16, 5.17 and 5.18. It can be seen from these equations that average of only either horizontal neighbors or vertical neighbors is considered in the linear prediction in the concerned direction, as they predict a pixel value better than a rhombus average (average of 4-neighbors of a pixel) [25]. Moreover, the average of pair of diagonal or anti-diagonal neighbors does not provide good prediction. Hence, additional consideration of rhombus neighborhood along with the diagonal or anti-diagonal neighbors strengthens the prediction [25].

The proposed method differs from the EGBSW predictor [25] in gradient estimation. To estimate the gradients, the weighted summations of pixel differences in horizontal, vertical, diagonal and anti-diagonal directions are computed as fol-

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation

lowing:

$$\begin{aligned}
D_h = & 2|W - E| + 2|E - EE| + 2|EE - EEE| + |NW - N| \\
& + |N - NE| + |NE - NEE| + |NEE - NEEE| \\
& + |SW - S| + |S - SE| + |SE - SEE| + |SEE - SEEE| \\
& + |SSW - SS| + |SS - SSE| + |SSE - SSEE| \\
& + |SSEE - SSEEE|
\end{aligned} \tag{5.25}$$

$$\begin{aligned}
D_v = & 2|N - S| + 2|S - SS| + 2|SS - SSS| + |NW - W| \\
& + |W - SW| + |SW - SSW| + |SSW - SSSW| \\
& + |NE - E| + |E - SE| + |SE - SSE| + |SSE - SSSE| \\
& + |NEE - EE| + |EE - SEE| + |SEE - SSEE| \\
& + |SSEE - SSSEE|
\end{aligned} \tag{5.26}$$

$$\begin{aligned}
D_d = & 2|NW - SE| + 2|SE - SSEE| + 2|SSEE - SSSEEE| \\
& + |W - S| + |S - SSE| + |SSE - SSSEE| + |SW - SS| \\
& + |SS - SSSE| + |SSW - SSS| + |N - E| \\
& + |E - SEE| + |SEE - SSEEE| + |NE - EE| \\
& + |EE - SEEE| + |NEE - EEE|
\end{aligned} \tag{5.27}$$

$$\begin{aligned}
D_a = & 2|NE - SW| + |W - N| + |SSW - S| + |S - E| \\
& + |E - NEE| + |SSSW - SS| + |SS - SE| + |SE - EE| \\
& + |EE - NEEE| + |SSS - SSE| + |SSE - SEE| \\
& + |SEE - EEE| + |SSSE - SSEE| + |SSEE - SEEE| \\
& + |SSSEE - SSEEE|
\end{aligned} \tag{5.28}$$

The gradients g_h , g_v , g_d and g_a in these four directions are computed from the summations of neighborhood pixel differences in respective directions as following:

$$\begin{aligned}
g_h = & \lfloor \frac{D_h}{18} + 0.5 \rfloor, g_v = \lfloor \frac{D_v}{18} + 0.5 \rfloor. \\
g_d = & \lfloor \frac{D_d}{18} + 0.5 \rfloor, g_a = \lfloor \frac{D_a}{16} + 0.5 \rfloor.
\end{aligned} \tag{5.29}$$

Finally, similar to EGBSW predictor [25], the directions corresponding to the two smallest gradient magnitudes are considered to estimate the final predicted value. Small gradient value indicates less change in pixel values in the corresponding direction. Therefore, the linear predictor in that direction generates

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation

more accurate predicted value. Hence, the directions with the two smallest gradient magnitudes are selected. The final predicted value x' is computed as the weighted average of the predicted values in those two directions as in equation 5.24. Each of the smallest two gradients g_0 and g_1 is normalized by the summation of these two gradients (i.e., $g_0 + g_1$). These two normalized gradient magnitudes act as the weights for the weighted summation in equation 5.24.

The major difference between the EGBSW predictor [25] and the proposed predictor is that the estimations of gradients are carried out using an extended neighborhood pixel values in a 5×5 window to capture more contextual information.

It is to be noted that the predicted values must be same during embedding and extraction processes. As per the above discussion, the predicted values are computed using the 24 neighborhood pixel values in the 5×5 window. Hence, same set of 24 values must be available during embedding and extraction. But as embedding data bits in a pixel changes the pixel value, original cover pixel values for all 24 neighbors cannot be availed during extraction. Therefore, following strategy is adopted to maintain the same set of 24 values for the neighboring pixels while embedding and extraction. Embedding is performed in a raster scan order traversal of image pixels. Hence, for a current pixel at coordinate (m, n) , the six neighborhood pixels at locations $NW, N, NE, NEE, NEEE$ and W already contain the marked values. Remaining eighteen neighborhood pixels are not visited till then. Hence, these eighteen pixels contain the original values. Similarly, the extraction of data bits and the restoration of the original pixel values are performed in the opposite order of raster scan traversal. Hence, during extraction process, the pixels at these eighteen coordinates have been already traversed and their original values have been restored back. The pixels at neighboring locations $NW, N, NE, NEE, NEEE$ and W still contain the marked values. Thus, same set of context values are available to the prediction algorithm during both embedding and extraction.

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation

5.2.3 Embedding and Extraction

The proposed reversible data hiding scheme uses the same adaptive prediction error histogram bin shifting method as with the previously described reversible data hiding technique (section 5.1) for embedding of either 1 bit or 2 bits of data in the prediction error of a pixel. Section 5.1.2 can be referred here for the detailed description of this embedding method. Even the estimation of local complexity and handling of auxiliary information are carried out exactly the same way as those of the previous technique in Section 5.1.

Similarly, extraction of data bits and restoration of cover image pixel values are carried out using the method as described in Section 5.1.3.

5.2.4 Experimental Results

This section presents the experimental results of the proposed improved gradient based prediction and adaptive prediction error histogram bin shifting. Performance of the proposed scheme has been tested using a set of standard test images as shown in figure 1.4. Normally, in a prediction error histogram of an image, tall bins are observed in zero-th bin and in its nearby bins. So, experiments are carried out by varying the two pairs of histogram bins: (i) histogram bins corresponding to prediction error as either -1 or 0 ($PE_l = -1$ and $PE_r = 0$), and (ii) histogram bins corresponding to prediction error as either 0 or 1 ($PE_l = 0$ and $PE_r = 1$). The embedding capacities (in number of bits) for these two cases have been reported in table 5.3. These embedding capacities mentioned in the table 5.3 are for single embedding ($T_{lc} = 0$). Significant difference in the embedding capacities is not observed for the above two cases. Peak Signal-to-Noise Ratio (PSNR) values (at the highest embedding capacity for each image) between the cover images and the corresponding marked images are also similar in these two cases. Similarly, embedding capacities in above two experimental cases are reported in table 5.4 for adaptive embedding (with $T_{lc} = 16$) for each image. Here, the adaptive embedding is carried out by considering the local complexity threshold T_{lc} as 16. Again, there is not present any significant difference either in embedding capacities or in PSNR values for adaptive embedding too irrespective of the selection of histogram bin pairs as either (-1, 0) or (0, 1). Though embedding capacities for

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation

Table 5.3: Achievable embedding capacities with histogram bin pairs $(-1, 0)$ and $(0, 1)$ using single bit embedding ($T_{lc} = 0$) for various test images.

Pair of bins		$(-1, 0)$		$(0, 1)$	
Image	T_{lc}	Embedding Capacity (Bits)	PSNR	Embedding Capacity (Bits)	PSNR
Lena	0	67460	48.81	64819	48.78
Lake	0	41119	48.56	40563	48.56
Boat	0	43423	48.58	42968	48.58
Mandrill	0	22432	48.40	22369	48.40
Elaine	0	37801	48.54	37677	48.54
Airplane	0	96040	49.12	90253	49.06
Peppers	0	47290	48.62	46662	48.61
Tiffany	0	56107	48.97	54231	48.94

adaptive embedding ($T_{lc} = 16$) are higher than those with single bit embedding ($T_{lc} = 0$). It justifies the use of adaptive embedding (either 1 bit or 2 bits in a pixel) over the use of single bit embedding in a pixel. But significant difference in performance is not observed between the histogram bin pairs being either $(-1, 0)$ or $(0, 1)$. Hence, any one pair among these two pairs of histogram bins can be considered for embedding.

The experiments are carried out by varying the value of local complexity threshold T_{lc} as 0, 1, 2, 4, 8 and 16. The value of T_{lc} being 0 indicates a single bit embedding. Higher the value of T_{lc} is, more data can be embedded. PSNR values between the cover images and the corresponding marked images are plotted in 5.8 with various payload sizes for each test image. It can be seen from the plots that the PSNR values are the highest in the case of $T_{lc} = 0$, i.e., single bit embedding. With increment in the value of T_{lc} , the PSNR values decrease. But higher embedding capacity is achieved with higher value of T_{lc} . These plots in figure 5.8 correspond to the experiments using $(-1, 0)$ histogram bin pairs. Similar trend in decrease of PSNR values and increase of embedding capacities with increased local complexity threshold value T_{lc} can be observed using $(0, 1)$

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation

Table 5.4: Achievable embedding capacities with histogram bin pairs $(-1, 0)$ and $(0, 1)$ using adaptive embedding ($T_{lc} = 16$) for various test images.

Pair of bins		$(-1, 0)$		$(0, 1)$	
Image	T_{lc}	Embedding Capacity (Bits)	PSNR	Embedding Capacity (Bits)	PSNR
Lena	16	104196	41.55	101774	41.54
Lake	16	59663	42.95	59719	42.95
Boat	16	60663	42.83	59872	42.82
Mandrill	16	26699	45.56	26699	45.56
Elaine	16	51358	43.40	51566	43.40
Airplane	16	156178	41.36	153141	41.33
Peppers	16	70263	41.69	70122	41.69
Tiffany	16	87988	41.56	86660	41.51

histogram bin pairs too.

Performance of the proposed method is compared against the performances of several existing methods like EGBSW [25], improved MED and improved SGAP [58], rhombus average [131], higher-significant bit-plane [49], Delaunay triangulation [119], B-tree triangular decomposition (chapter 3) and random distribution of reference pixels (chapter 4) based reversible data hiding techniques. PSNR values between the cover images and the corresponding marked images for various payload sizes are shown in figure 5.9 for all the comparing techniques. It can be observed that the proposed technique outperforms the existing comparing techniques. It can also be observed that the performance of the proposed technique in both experimental cases - (i) histogram bins corresponding to prediction error as either -1 and 0 ($PE_l = -1$ and $PE_r = 0$), and (ii) histogram bins corresponding to prediction error as either 0 and 1 ($PE_l = 0$ and $PE_r = 1$) - are similar. In these plots (figure 5.9), the legend 'Improved Gradient (0)' refers to the proposed improved gradient based prediction with adaptive histogram bin shifting using left bin PE_l as 0 and right bin PE_r as 1. Similarly, the legend 'Improved Gradient (-1)' refers to the proposed improved gradient based prediction with adaptive

5.2 Reversible Data Hiding with Selected Directional Context Based Prediction Using an Improved Gradient Estimation

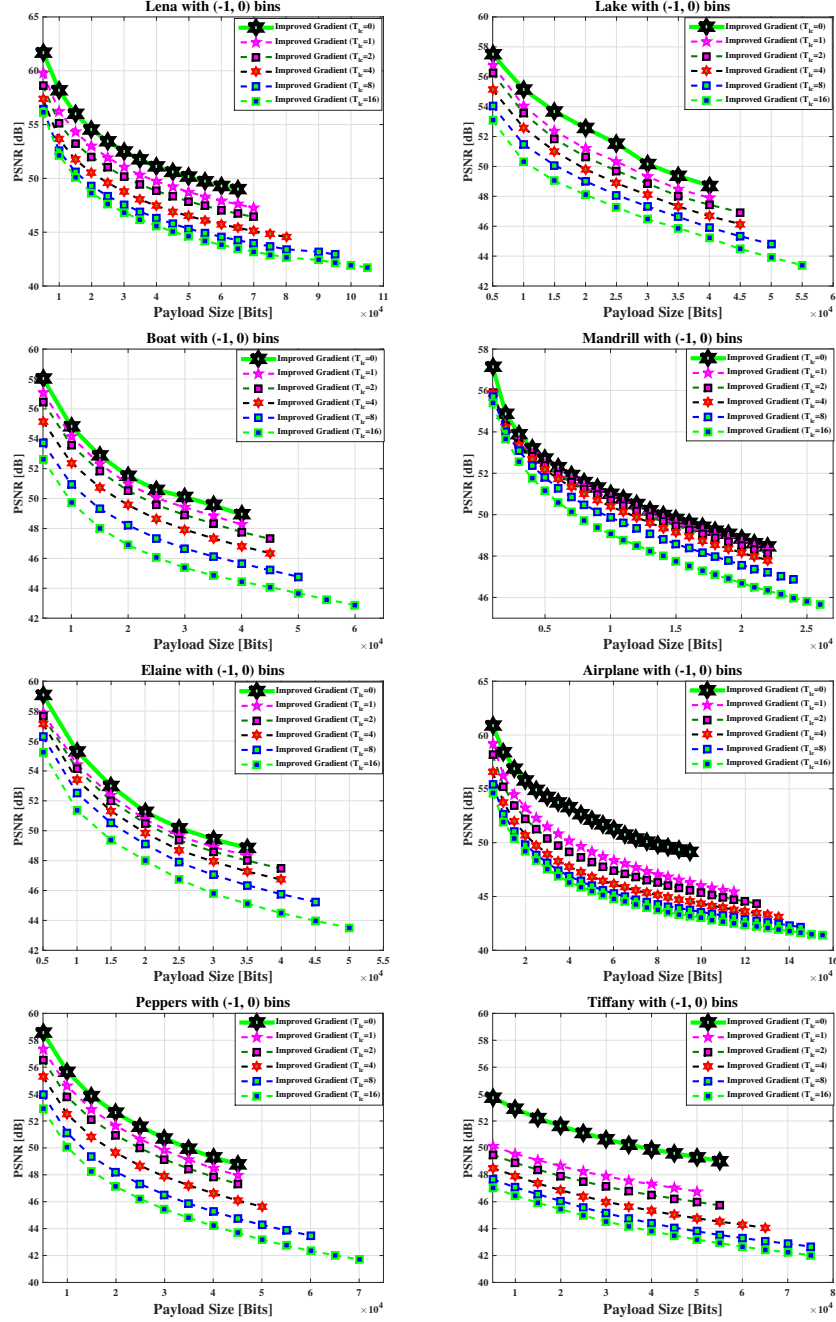


Figure 5.8: Payload size versus PSNR plots using various values of local complexity threshold T_{Lc} for various images.

histogram bin shifting using left bin PE_l as -1 and right bin PE_r as 0.

Finally, the performances of the two techniques in this chapter are compared. PSNR values between the cover images and the corresponding marked images for these two techniques are plotted in figure 5.10 for various payload sizes. The legend as 'Selected Context' refers to the reversible data hiding technique with selected directional context based prediction using 3×3 neighborhood (Section 5.1). The legend as 'Improved Gradient' refers to the reversible data hiding technique with an improved gradient based prediction using 5×5 neighborhood. In both the cases, '-1' as part of the legend refers to the technique with adaptive histogram bin shifting using left bin PE_l as -1 and right bin PE_r as 0. Similarly, '0' as part of the legend refers to the technique with adaptive histogram bin shifting using left bin PE_l as 0 and right bin PE_r as 1. It can be observed from the plots in figure 5.10 that the reversible data hiding technique with an improved gradient based prediction performs better than the technique with 3×3 neighborhood.

The marked images as per the proposed reversible data hiding scheme for various payload size are presented in figure 5.11.

5.3 Summary

In this chapter, two novel reversible data hiding techniques are proposed. These two techniques differ in how a pixel value is predicted. The first of these two techniques is based on selected directional context based prediction, where diversities and averages of pixel value pairs select the directions. A context based on 3×3 neighborhood of a pixel is considered for predicting a pixel value. Integer average of pair of neighborhood values in a direction is used to predict the pixel value in the concerned directional context. The diversities between the neighborhood pixel values and the predicted values in four directions (horizontal, vertical, diagonal, and anti-diagonal) are used to select a set of directional contexts. The directions with low diversities are selected. The integer average of the predicted values in the selected directional contexts determines the final predicted value for a pixel. The second technique is based on an improved gradient estimation. A

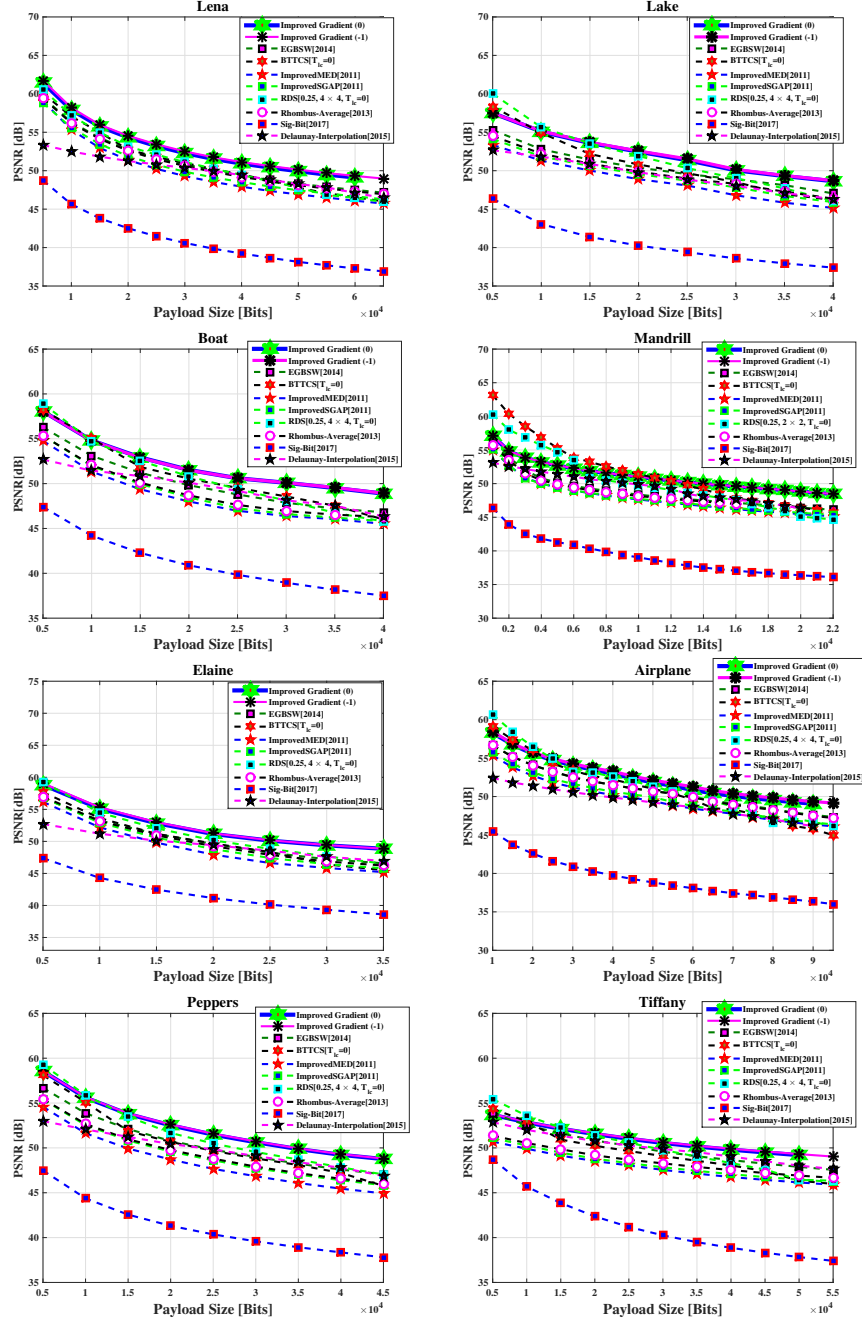


Figure 5.9: Comparison of PSNR values between the cover images and the marked images with various payload sizes among several reversible data hiding techniques

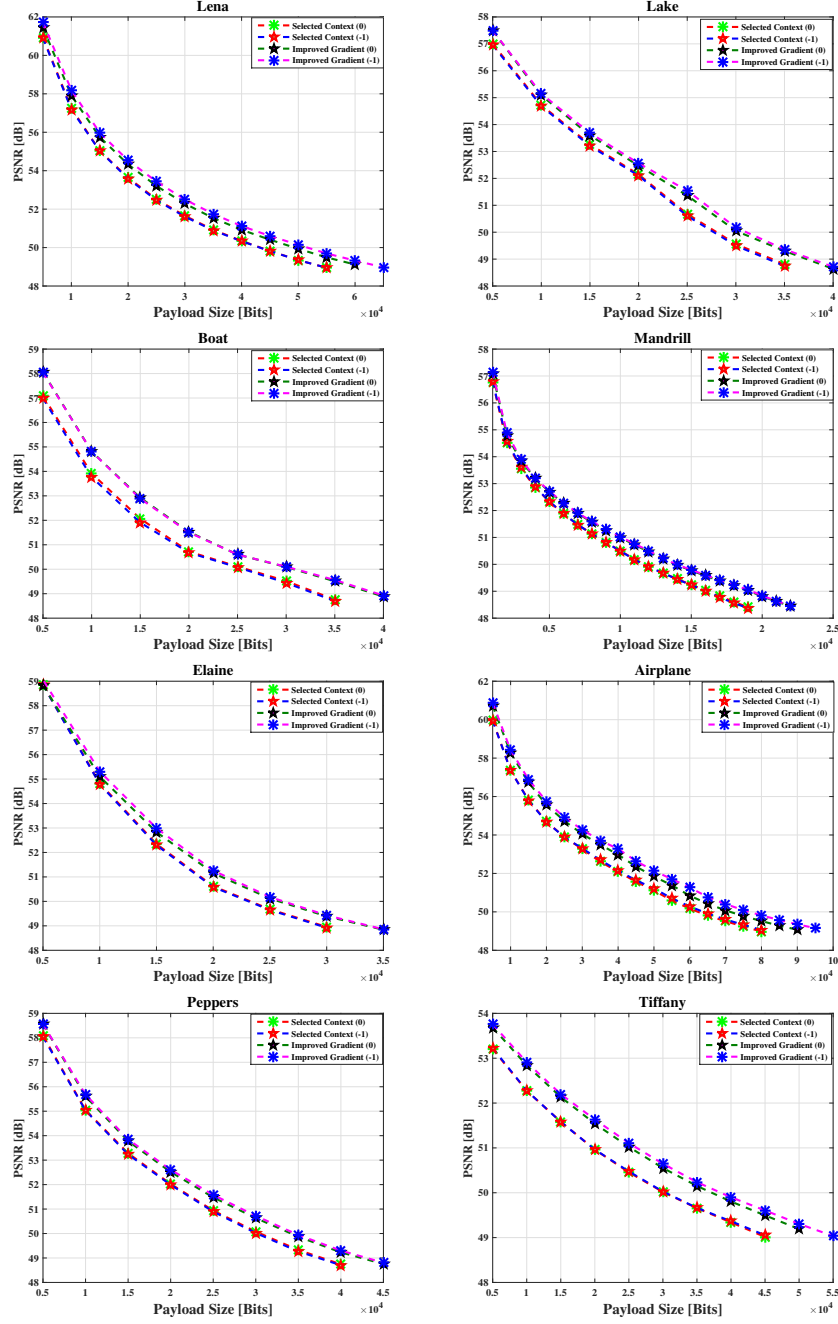


Figure 5.10: Performance comparison between the proposed reversible data hiding techniques in Section 5.1 and Section 5.2



Figure 5.11: Results of the Improved Gradient based reversible data hiding scheme for various test images (row-wise). Columns from left to right: Original image, marked images with 5000 bits, 10000 bits, 15000 bits and 20000 bits

novel gradient estimation is proposed using 5×5 neighborhood. The pixel prediction strategy uses the new gradient estimation. Basically, the estimated gradient values are used to select the directions of linear predictors.

An adaptive prediction error histogram bin shifting strategy is introduced in this chapter for embedding of data bits in the prediction error of a pixel. Both of the proposed reversible data hiding techniques use this embedding strategy. According to this strategy, either 1 bit or 2 bits of data are embedded in the prediction error of a pixel based on local complexity of the pixel. Increment in local complexity threshold value helps in achieving higher embedding capacity. Experiments are conducted by embedding data bits in pixels belonging to a pair of prediction error histogram bins. Experiments reveal that significant difference is not observed for the selection of histogram bin pair as either $(-1, 0)$ or $(0, 1)$.

It is to be noted that the proposed reversible data hiding technique, as well as other reversible data hiding techniques in the literature, are fragile in nature. These techniques are not robust at all against any kind of tampering of the marked image. Hence, experiments demonstrating the robustness analysis of the proposed techniques have not been carried out. Normally, these fragile reversible data hiding techniques are designed for ensuring the integrity of an image.

Chapter 6

Reversible Data Hiding Using Multiple Predictors

Literature of prediction error expansion based reversible data hiding techniques is full of various pixel prediction strategies. Researchers have put considerable efforts to come up with good pixel prediction schemes. A good pixel prediction scheme leads to a good reversible data hiding technique. Several of these prediction schemes predict a pixel value using the values of its neighboring pixels. Either simple computation (such as average [69] and weighted median [91]) or gradient estimation [55] [62] involving the neighboring pixel values predict the center pixel value. Two such schemes have also been proposed in Chapter 5. There also exist reference pixel based prediction schemes [118] [119]. Two such schemes have also been proposed in Chapter 3 and Chapter 4. But the question arises whether a single pixel prediction strategy provides good prediction accuracy in different neighborhood contexts. Assumption here is that such pixel predictor does not exist to provide accurate prediction in varied neighborhood contexts. Therefore, multi-predictor strategies combining the outcomes of multiple predictors also exist in literature [128] and [127]. In this chapter, performances of several existing predictors (including few of the proposed predictors) are examined in the light of varied neighborhood contexts in an image. Experimental observations lead to a conclusion that there does not exist any single good predictor for varied neighborhood contexts. This conclusion strengthens the need of a multi-predictor strategy. Several simple strategies for combining the outcomes of multiple predictors are

experimentally investigated. Like the techniques in Chapter 5, an adaptive prediction error histogram bin shifting method is adopted for this task.

The organization of this chapter is mentioned here. Performances of multiple predictors in the context of varied neighborhood contexts in images are investigated in Section 6.1. Multiple simple strategies for combining the outcomes of multiple predictors are discussed in Section 6.2. Embedding and Extraction processes are mentioned in Section 6.3. Experimental results are discussed in Section 6.4. The summary of contributions in this chapter is presented in Section 6.5.

6.1 Performance Comparison of Pixel Predictors

Several pixel prediction strategies exist in literature in the context of prediction-error expansion or histogram bin shifting based reversible data hiding. Performances of sixteen such predictors are investigated for predicting a pixel value. These sixteen predictors are listed here. The first nine of these predictors use simple computation involving few of the neighborhood pixels. These are:

1. Median Edge Detector (MED) predictor [23], which considers the presence of an edge either at the top or at the left of a pixel.
2. Rhombus Average (RA) predictor [69], which is a simple average of 4-neighbors of a pixel.
3. Improved Rhombus (IR) [74], which considers the average of either two vertical neighbors, two horizontal neighbors or 4-neighbors in a pixel.
4. Selected directional context (SDC) based predictor using 8-neighborhood of a pixel (Section 5.1.1).
5. Average of the pixel values in 4-neighborhood (AV) as it is computed in [70].

6.1 Performance Comparison of Pixel Predictors

6. Predictor based on horizontal grouping (HG) using a weighted average of six neighboring pixels (top-left, top, top-right, bottom-left, bottom, and bottom-right neighbors) [163].
7. Median of the pixel values (ME) in the 8-neighborhood of a pixel.
8. Maximum (MAX) pixel value among top, left and top-left neighbors [146].
9. Minimum (MIN) pixel value among top, left and top-left neighbors [146].

Similarly, several gradient based predictors are also used for the experiment here. These gradient based predictors vary in the estimation of gradients and in the neighborhood being considered. These gradient based predictors are listed as following:

10. Gradient Adjusted Predictor (GAP) [61]
11. Simplified Gradient Adjusted Predictor (SGAP) [58]
12. Gradient Based Selective Weighting (GBSW) [62]
13. Extended Gradient Based Selective Weighting (EGBSW) [25]
14. Improved Gradient Based Selective Weighting (IGBSW) (Section 5.2.2)
15. Accurate Gradient Selective Prediction (AGSP) [127]
16. Threshold Controlled Gradient Adaptive Planar Prediction (TGAPP)[63]

The above predictors are applied to predict pixel values in the set of eight test images in figure 1.4. For each of the above predictors, pixels in the few rows and columns at the image boundary are not predicted due to unavailability of the complete set of neighborhood pixels. For example, in the case of MED predictor [23], pixels at the top most row and left most column are left out. In the case of IGBSW predictor in Section 5.2.2, numbers of rows being left out from upper and lower sides at the boundary are 1 and 3, respectively. Similarly, number of columns being left out from left and right sides at the boundary are 1 and 3, respectively, in IGBSW. Finally, a set of pixels can be identified, where all of

6.1 Performance Comparison of Pixel Predictors

the above predictors are applied. This common set of pixels for all predictors excludes 2 rows and 3 rows from the upper and the lower sides at the boundary, respectively. Similarly, this common set of pixels for all predictors excludes 2 columns and 3 columns from the left and right sides at the image boundary, respectively. Three rows at the lower side and three columns at the right side are excluded because of IGBSW predictor. Similarly, the exclusions of top two rows and left most two columns are required due to the neighborhood of AGSP predictor [127].

For each pixel in the common set of considered pixels, sixteen prediction errors are obtained by using the above predictors. A prediction scheme is said to produce the best performance for a pixel if the prediction scheme causes the lowest value among the absolute values of sixteen prediction errors. More than one prediction scheme may produce the lowest absolute value of the prediction error for a pixel. In such cases, multiple such prediction schemes are noted down. Thus for each predictor, percentage of pixels (among total number of considered pixels), where the predictor performs the best, is estimated. These image-wise percentage values are tabulated in tables 6.1 and 6.2 for each predictor. These values are presented in two tables due to space constraint. As an example, in case of Airplane image, MED predictor produces the best performance among all predictors in 25.36% of the considered pixels. Similarly, in case of Boat image, IGBSW predictor produces the best performance among all predictors in 16.26% of the considered pixels. For each test image, the best performing predictor is highlighted in bold font. Here, performance of a predictor is judged based on the percentage of pixels in which it causes the best performance. For example, the selected directional context (SDC) with 8-neighborhood of a pixel performs the best among all sixteen comparing predictors in the case of Boat image. But the performance of the improved rhombus (IR) predictor is also not far behind to the performance of the SDC predictor. Moreover, IR predictor performs the best for the Airplane image. Hence, there does not exist a single predictor which performs the best for all images.

Average of the percentages of pixels in which a predictor performs the best, is computed over all images. These average percentage values are reported at the last row of tables 6.1 and 6.2. It can be seen that SDC predictor performs the

6.1 Performance Comparison of Pixel Predictors

Table 6.1: Percentage of pixels for which each predictor produces the lowest absolute value of the prediction error

Image	MED	RA	IR	SDC	AV	HG	ME	MAX
Lena	18.74	20.45	23.84	24.18	10.12	21.27	21.29	18.31
Lake	13.66	13.33	16.59	18.83	7.53	17.67	16.36	17.4
Boat	15.15	13.64	19.65	19.97	7.31	18.59	15.4	16.98
Mandrill	13.11	9.84	17.2	16.42	6.23	10.6	9.71	14.55
Elaine	13.07	12.71	15.28	18.55	7.18	15.96	15.35	18.71
Airplane	25.36	27.01	29.76	28.84	13.39	26.52	28.08	22.65
Peppers	13.05	13.49	16.28	20.37	6.99	21.02	18.81	18.35
Tiffany	20.62	21.16	25.07	26.36	10.29	22.28	22.84	21.22
Average	16.6	16.45	20.46	21.69	8.63	19.24	18.48	18.52

best in 21.69% of the considered pixels. This is the best performance among the performances of all sixteen predictors. But this performance cannot be considered as impressive because it provides the best performance in only 21.69% of the considered pixels. There exist another 78.31% of pixels where other predictors perform better than SDC predictor. This observation highlights the need of using multiple predictors.

But a question arises on how many predictors can be considered in such a multi-predictor scheme. More importantly, there is a need to identify a set of those good predictors which can be used in the multi-predictor scheme. To find an answer for this question, an average of these average percentage values over all sixteen predictors is estimated as 17.32%. The predictors which perform better than this average performance (i.e., whose average percentage is higher than 17.32%), are finally considered for a multi-predictor scheme. Therefore, the selected predictors are IR, SDC, HG, ME, MAX, MIN, EGBSW, IGBSW, AGSP, and TGAPP. The average percentage values for each of these selected predictors are highlighted using bold font in last row (for the average percentages) of the tables 6.1 and 6.2. Other six predictors are not considered in the context of a multi-predictor scheme as their performances are worse than the average performance of all sixteen predictors. Using the set of selected predictors, a multi-predictor scheme is developed. Subsequent sections in this chapter consider only

6.2 Strategies for Combining Multiple Predicted Values

Table 6.2: Percentage of pixels for which each predictor produces the lowest absolute value of the prediction error (continuation of table 6.1)

Image	MIN	GAP	SG AP	GB SW	EGB SW	IGB SW	AG SP	TGA PP
Lena	17.87	15.39	18.45	18.52	21.71	22.08	18.82	19.4
Lake	17.6	11.53	13.69	16.06	14.75	15.35	17.28	15.12
Boat	17.34	10.60	13.00	14.71	15.78	16.26	16.02	18.1
Mandrill	14.79	9.42	11.15	10.75	11.37	12.65	11.03	15.4
Elaine	18.25	11.41	13.46	14.61	13.55	14.16	16.14	15.63
Airplane	20.5	23.41	24.4	21.47	27.94	28.59	23.13	24.17
Peppers	18.42	12.08	13.90	17.79	16.09	16.33	19.19	14.89
Tiffany	19.62	17.72	19.57	19.52	23.28	24.1	20.4	21.28
Average	18.05	13.94	15.95	16.68	18.06	18.69	17.75	17.99

the selected set of predictors.

6.2 Strategies for Combining Multiple Predicted Values

This section presents several ways of combining multiple predicted values in a multi-predictor scheme. Few such strategies can be found in the literature of reversible data hiding. The maximum and the minimum of a set of predicted values are considered in 1st round and in 2nd round, respectively, of the multi-predictor scheme in [128]. In [127], a pixel value is compared with the maximum and the minimum of several predicted values. Any of the following four cases may arise for a pixel: (1) If the maximum and the minimum of the predicted values are same (i.e., all predicted values are same), the maximum (or the minimum) value is considered as the final predicted value for the pixel. As all predicted values are same, a single predicted value can easily be obtained for this pixel in both embedding and extraction phases. (2) If the pixel value is greater than or equal to the maximum of the predicted values, the maximum predicted value is considered as the final predicted value for the pixel. Embedding of data bits in this pixel

6.2 Strategies for Combining Multiple Predicted Values

causes the marked pixel value to remain as greater than or equal to the maximum predicted value. (3) On the contrary, if the pixel value is lesser than or equal to the minimum of the predicted values, the minimum predicted value is considered as the final predicted value for the pixel. Embedding of data bits in this pixel causes the marked pixel value to remain as lesser than or equal to the minimum predicted value. In both cases (2) and (3), obtaining the final predicted value is easy as the original pixel value and the marked pixel value remain at the same side of the maximum and the minimum of the predicted values. (4) If the pixel value falls in between the maximum and the minimum of the predicted values, final predicted value is not decided for the pixel. Hence, embedding of data bits is not carried out in this pixel. This is because the directions of the original pixel value and the marked pixel value with respect to both of the maximum and the minimum predicted values may not remain the same. Hence, obtaining the same predicted value for both embedding and extraction may not be possible for every pixel. Due to the occurrence of case (4), embedding capacity of the multi-predictor scheme [127] is less. Embedding may not be possible in many of the pixels. This condition arises for more number of pixels if the difference between the maximum and the minimum of the predicted values is high. Moreover, a high value for the said difference may be observed if number of predictors is also high.

Rather, several simple strategies for combining multiple predicted values are discussed in this section. Let p predictors produce predicted values as $x'_1, x'_2, \dots, x'_p$. Then, these predicted values can be combined in one of the following ways:

- Minimum of the predicted values:

$$x'_{min} = x'_i, \text{ where } x'_i \leq x'_j, \text{ for all values of } j \text{ from } 1 \text{ to } p \quad (6.1)$$

- Maximum of the predicted values

$$x'_{max} = x'_i, \text{ where } x'_i \geq x'_j, \text{ for all values of } j \text{ from } 1 \text{ to } p \quad (6.2)$$

- Average of the predicted values

$$x'_{mean1} = \frac{1}{p} \sum_{i=1}^p x'_i \quad (6.3)$$

6.2 Strategies for Combining Multiple Predicted Values

- Average of the maximum and the minimum values

$$x'_{mean2} = \frac{x'_{max} + x'_{min}}{2} \quad (6.4)$$

- Median of the predicted values: To compute the median of a set of predicted values $x'_1, x'_2, \dots, x'_p$, these values are sorted in ascending order as $x'_{sort_1}, x'_{sort_2}, \dots, x'_{sort_p}$, where $x'_{sort_1} \leq x'_{sort_2} \leq \dots \leq x'_{sort_p}$. Then, the median is computed as:

$$\begin{aligned} x'_{median} &= median(x'_{sort_1}, x'_{sort_2}, \dots, x'_{sort_p}) \\ &= x'_{sort_{\frac{p+1}{2}}} \quad \text{if } p \text{ is odd} \\ &= \frac{x'_{sort_{\frac{p}{2}}} + x'_{sort_{\frac{p}{2}+1}}}{2} \quad \text{if } p \text{ is even} \end{aligned} \quad (6.5)$$

It is to be noted that the predicted values must be same during embedding and extraction processes. As per the above discussion, the final predicted values in a multi-predictor scheme depends on the set of p predicted values. Hence, each of these p predictors must generate the same predicted value for embedding and extraction processes. It is possible if the same set of neighborhood pixel values are available during embedding and extraction. But as embedding data bits in a pixel changes the pixel value, original cover pixel values for all neighboring pixels cannot be availed during extraction. Therefore, following strategy is adopted to maintain the same set of neighboring pixel values for embedding and extraction. Embedding is performed in a raster scan order traversal of image pixels. Hence, the pixels, which have been visited prior to the visit of current pixel (m, n) , already contain the marked values. Remaining neighborhood pixels are not visited till then. Hence, these pixels contain the original values. Similarly, the extraction of data bits and the restoration of the original pixel values are performed in the opposite order of raster scan traversal. Hence, during extraction process, the later set of pixels have been already traversed and their original values have been restored back. The first set of neighboring pixels still contain the marked values. Thus, for any of the p prediction algorithms, the same context values are available during both embedding and extraction.

6.3 Embedding and Extraction

The proposed reversible data hiding technique uses the same adaptive prediction error histogram bin shifting method as with the previously described reversible data hiding technique (Section 5.1) for embedding of either 1 bit or 2 bits of data in the prediction error of a pixel. Section 5.1.2 can be referred here for the detailed description of this embedding method. Even the estimation of local complexity and handling of auxiliary information are carried out exactly the same way as those of the previous technique in Section 5.1.

Similarly, extraction of data bits and restoration of cover image pixel values are carried out using the method as described in Section 5.1.3.

6.4 Experimental Results

This section presents experimental results based on the multi-predictor scheme using adaptive prediction error histogram bin shifting strategy. Selection of ten predictors (IR, SDC, HG, ME, MAX, MIN, EGBSW, IGBSW, AGSP, and TGAPP) for the proposed multi-predictor scheme has already been explained in Section 6.1. Five simple strategies for combining the outcomes of these ten predictors are experimentally studied. Performances of these multi-predictor strategies are tested using the standard test images in figure 1.4. The performances are evaluated using the peak signal-to-noise ratio (PSNR) between the cover images and the corresponding marked images by varying the payload size. These PSNR values for various multi-predictor strategies are plotted in figure 6.1 against various payload sizes for each test image. Normally, in a prediction error histogram of an image, tall bins are observed in zero-th bin and in its nearby bins. So, experiments are carried out by varying the two pairs of histogram bins: (i) histogram bins corresponding to prediction error as either -1 or 0 ($PE_l = -1$ and $PE_r = 0$), and (ii) histogram bins corresponding to prediction error as either 0 or 1 ($PE_l = 0$ and $PE_r = 1$). In these plots, the legends for combination strategies with a '(0)' refer to the concerned multi-predictor combination schemes with adaptive histogram bin shifting using left bin PE_l as 0 and right bin PE_r as 1. Similarly,

the legends for combination strategies with a '(-1)' refer to the concerned multi-predictor combination strategies with adaptive histogram bin shifting using left bin PE_l as -1 and right bin PE_r as 0. Moreover, the reported results in these plots correspond to single bit embedding in a pixel ($T_{lc} = 0$). It can be observed from the plots in figure 6.1 that the multi-predictor scheme using median of the predicted values produces better performance as compared against other combination strategies of multiple predicted values. Therefore, subsequent discussions in this section refer to the median based combination strategy (equation 6.5) for the multi-predictor scheme.

Additionally, for any multi-predictor combination strategy, significant difference in results cannot be noticed from the plots in figure 6.1 between the two experimental cases with pairs of histogram bins corresponding to (i) prediction error as either -1 or 0 ($PE_l = -1$ and $PE_r = 0$), and (ii) prediction error as either 0 or 1 ($PE_l = 0$ and $PE_r = 1$). This observation is further strengthened by the reported embedding capacities (in number of bits) for these two cases in table 6.3 and table 6.4. These embedding capacities in table 6.3 are for single embedding ($T_{lc} = 0$). Significant difference in the embedding capacities is not observed for the above two selections of histogram bin pairs. Peak signal-to-noise ratio (PSNR) values (at the highest embedding capacity for each image) between the cover images and the corresponding marked images are also similar in these two cases for each image. Similarly, embedding capacities in above two experimental cases are reported in table 6.4 for adaptive embedding (with $T_{lc} = 16$) for each image. The adaptive embedding is carried out by considering the local complexity threshold $T_{lc} = 16$. Again, there is not present any significant difference either in embedding capacities or in PSNR values for adaptive embedding too irrespective of the selection of histogram bin pairs as (-1, 0) or (0, 1). Though embedding capacities for adaptive embedding ($T_{lc} = 16$) are higher than those with single embedding ($T_{lc} = 0$). It justifies the use of adaptive embedding (either 1 bit or 2 bits in a pixel) over the use of single bit embedding in a pixel. But significant difference in performance is not observed between the histogram bin pairs being (-1, 0) or (0, 1). Hence, anyone pair among these two pairs of histogram bins can be considered for embedding.

6.4 Experimental Results

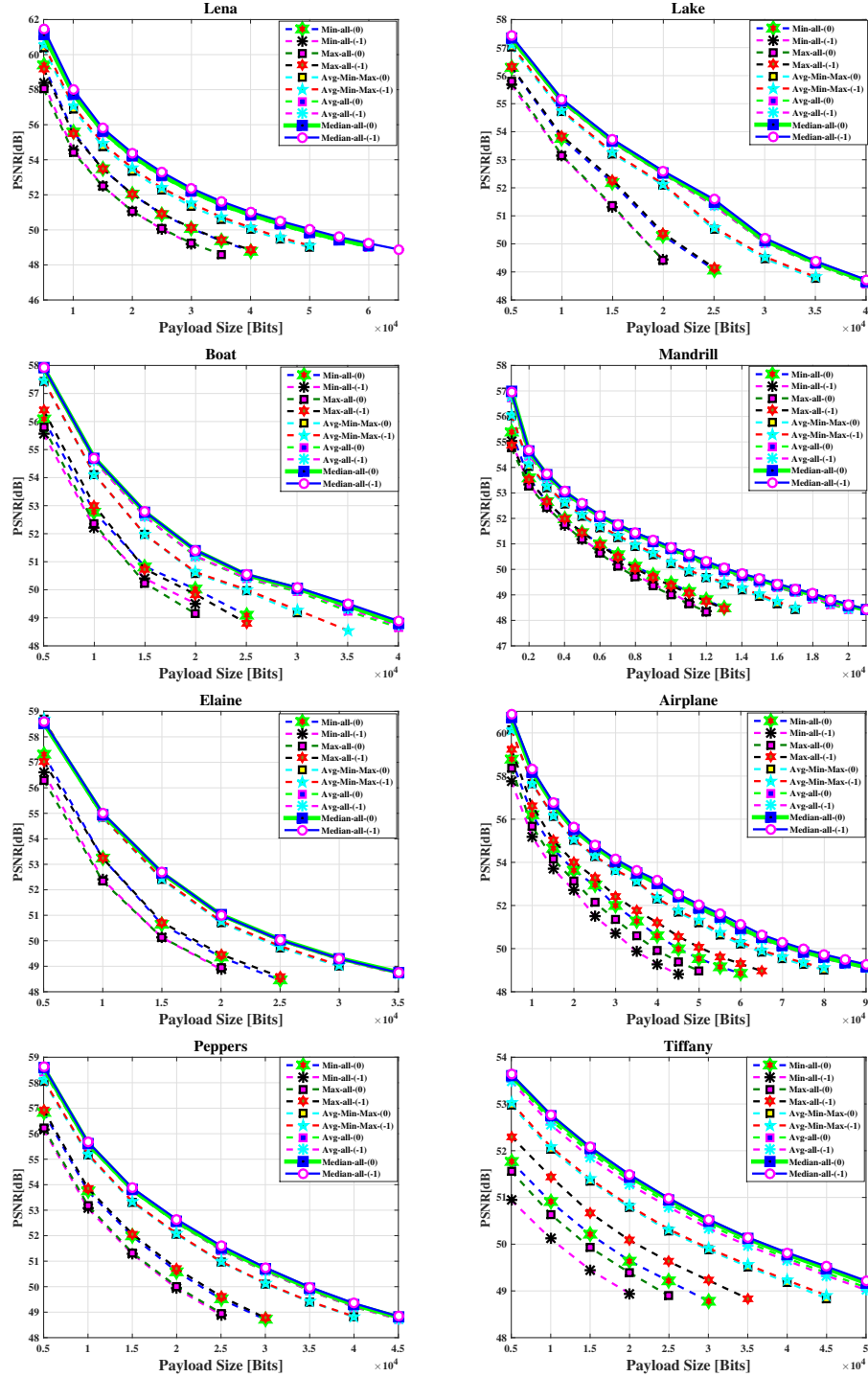


Figure 6.1: Performances of various combination strategies for the multi-predictor scheme

6.4 Experimental Results

Table 6.3: Achievable embedding capacities and corresponding PSNR values with histogram bin pairs $(-1, 0)$ and $(0, 1)$ using single bit embedding ($T_{lc} = 0$) for various test images.

Pair of bins		$(-1, 0)$		$(0, 1)$	
Image	T_{lc}	Embedding Capacity	PSNR	Embedding Capacity	PSNR
Lena	0	66066	48.81	64142	48.80
Lake	0	41064	48.58	40633	48.58
Boat	0	42785	48.60	42165	48.59
Mandrill	0	21329	48.40	21115	48.40
Elaine	0	36722	48.55	36863	48.55
Airplane	0	93623	49.14	91034	49.11
Peppers	0	47459	48.64	47171	48.64
Tiffany	0	54468	48.97	53628	48.96

Table 6.4: Achievable embedding capacities and corresponding PSNR values with histogram bin pairs $(-1, 0)$ and $(0, 1)$ using adaptive embedding ($T_{lc} = 16$) for various test images.

Pair of bins		$(-1, 0)$		$(0, 1)$	
Image	T_{lc}	Embedding Capacity	PSNR	Embedding Capacity	PSNR
Lena	16	103995	41.59	100786	41.56
Lake	16	60185	42.98	59605	42.97
Boat	16	60372	42.86	59390	42.84
Mandrill	16	25517	45.56	25359	45.56
Elaine	16	50371	43.43	50510	43.44
Airplane	16	162733	41.44	157949	41.39
Peppers	16	71684	41.72	70986	41.71
Tiffany	16	89327	41.58	87125	41.52

The experiments are carried out by varying the value of local complexity threshold T_{lc} as 0, 1, 2, 4, 8, and 16. The value of T_{lc} being 0 indicates a single bit embedding. Higher the value of T_{lc} is, more data can be embedded. PSNR values between the cover images and the corresponding marked images are plotted in figure 6.2 with various payload sizes for each test image. It can be seen from these plots that PSNR values are the highest in case of $T_{lc} = 0$, i.e., single embedding. With increase in the value of T_{lc} , the PSNR values decrease. But higher embedding capacity is achieved with higher value of T_{lc} . These plots in figure 6.2 correspond to the experiments using (-1, 0) histogram bin pairs. Similar trend in decrease of PSNR values and increase of embedding capacities with increased local complexity threshold value T_{lc} can be observed using (0, 1) histogram bin pairs too.

The marked images as per the proposed reversible data hiding technique for various payload sizes are presented in figure 6.3. These marked images correspond to single embedding (i.e., $T_{lc} = 0$). Finally, performance of the proposed multi-predictor based reversible data hiding technique using median based predictor combination strategy is compared against the performances of several existing methods like EGBSW [25], improved MED and improved SGAP [58], rhombus average [131], higher-significant bit-plane [49], Delaunay triangulation [119], B-tree triangular decomposition (Chapter 3) and random distribution of reference pixels (Chapter 4) based reversible data hiding techniques. PSNR values between the cover and the marked images for various payload sizes have been shown in figure 6.4 for all the comparing methods. Reported PSNR values of the proposed multi-predictor scheme in these plots correspond to single embedding (i.e., $T_{lc} = 0$). It can be observed that the proposed method outperforms the existing comparing methods. It can also be observed that the performance of the proposed method in both experimental cases - (i) histogram bins correspond to prediction error as -1 and 0 ($PE_l = -1$ and $PE_r = 0$), and (ii) histogram bins correspond to prediction error as 0 and 1 ($PE_l = 0$ and $PE_r = 1$) - are similar. In these plots (figure 6.4, the legend 'Median-all-(0)' refers to the proposed median based multi-predictor scheme with adaptive histogram bin shifting using left bin PE_l as 0 and right bin PE_r as 1. Similarly, the legend 'Median-all-(-1)' refers to the proposed median

6.4 Experimental Results

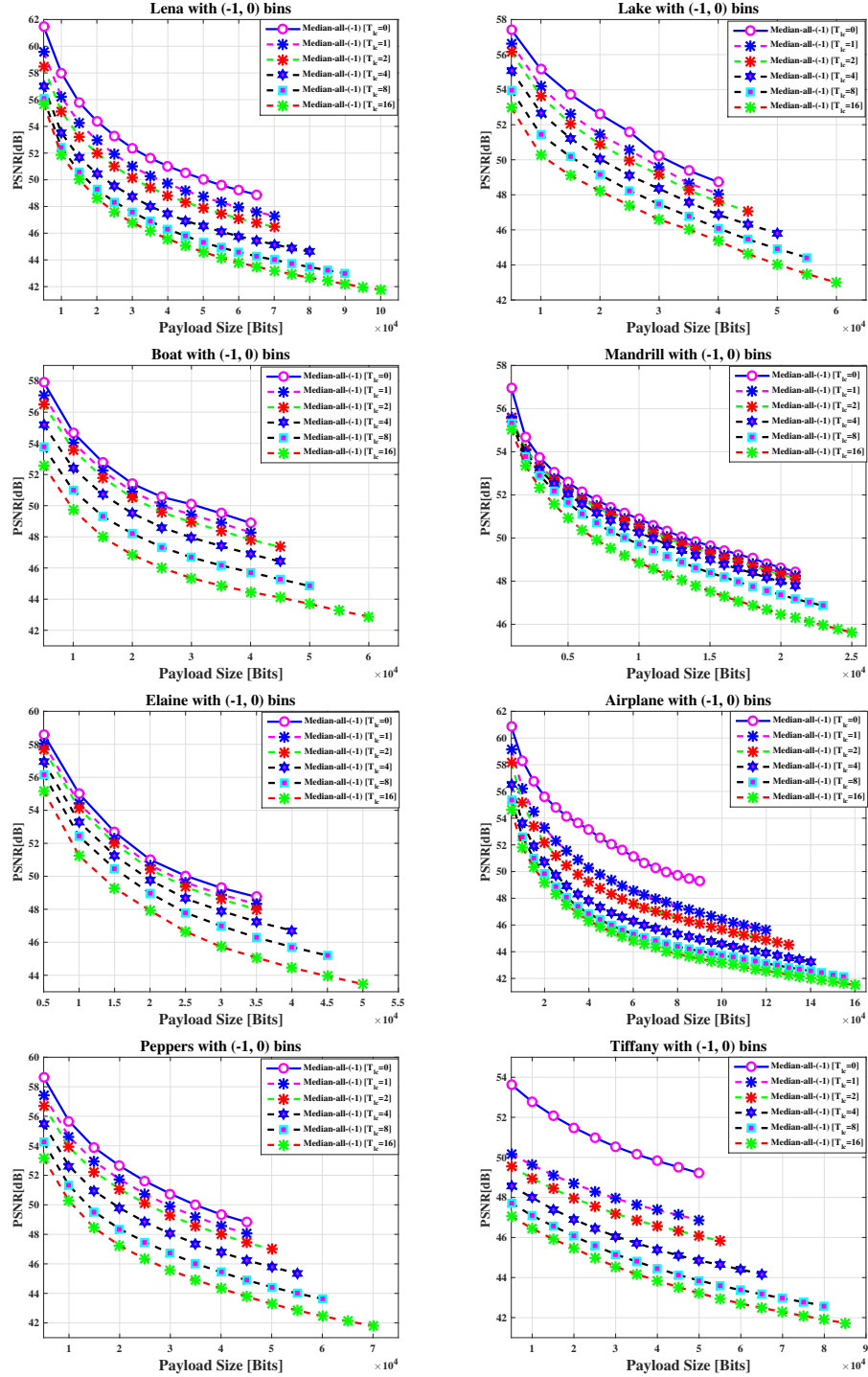


Figure 6.2: Embedding capacity versus PSNR plots using various values of local complexity threshold T_{lc} for various images.

based multi-predictor scheme with adaptive histogram bin shifting using left bin PE_l as -1 and right bin PE_r as 0.

Finally, the performance of the proposed multi-predictor reversible data hiding technique with median based combination strategy is compared with the performances of the two reversible data hiding techniques in previous chapter (Chapter 5). PSNR values between the cover images and the corresponding marked images for these three techniques (with single embedding) are plotted in figure 6.5 for various payload sizes. The legend as 'Selected Context' refers to the reversible data hiding technique with selected directional context based prediction using 3×3 neighborhood (Section 5.1). The legend as 'Improved Gradient' refers to the reversible data hiding technique with an improved gradient based prediction using 5×5 neighborhood (Section 5.2). The legend as 'Median-all' refers to the proposed reversible data hiding technique with multi-predictor scheme using median based combination strategy as mentioned in this chapter. For each of these techniques, '-1' as part of the legend refers to the technique with adaptive histogram bin shifting using left bin PE_l as -1 and right bin PE_r as 0. Similarly, '0' as part of the legend refers to the technique with adaptive histogram bin shifting using left bin PE_l as 0 and right bin PE_r as 1. It can be observed from the plots in figure 6.5 that the reversible data hiding technique with multi-predictor scheme using median as combination strategy performs better than the reversible data hiding technique using selected directional context with 3×3 neighborhood (Section 5.1). Although performance of the reversible data hiding technique using improved gradient based prediction (Section 5.2) is comparable with that of the proposed multi-predictor based reversible data hiding technique with median as combination strategy.

6.5 Summary

Major contributions of this chapter are manifold. At first, existences of several prediction strategies in the context of prediction error expansion/histogram bin shifting based reversible data hiding technique are recognized. Performance analysis of sixteen such predictors concludes that a single predictor does not always generate the lowest absolute value for prediction error. Hence, it justifies the need



Figure 6.3: Results of the proposed multi-predictor scheme based reversible data hiding scheme for various test images (row-wise). Columns from left to right: Original image, marked images with 5000 bits, 10000 bits, 15000 bits and 20000 bits

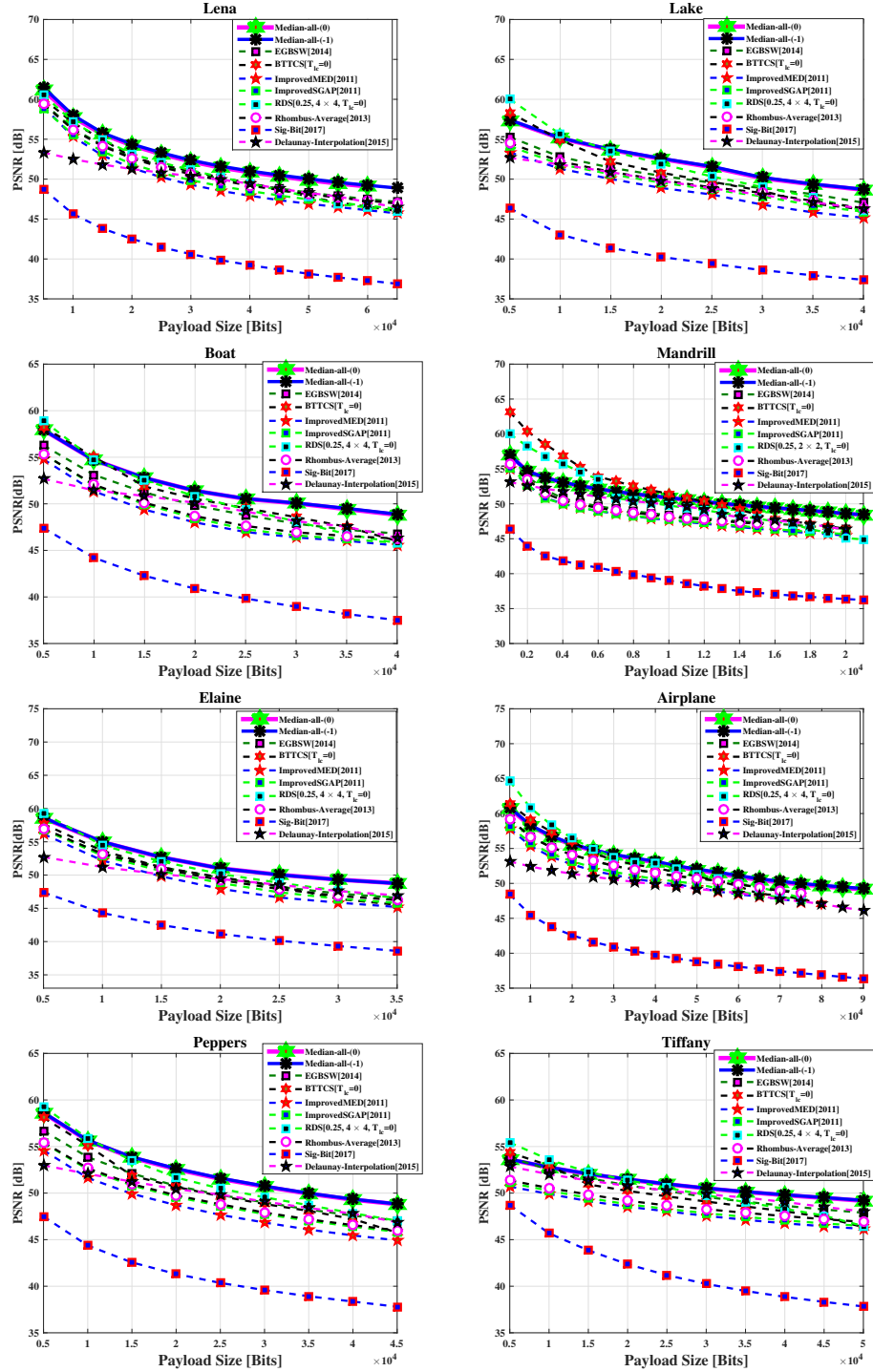


Figure 6.4: Comparison of PSNR values between the cover images and the marked images with various payload sizes among several reversible data hiding techniques

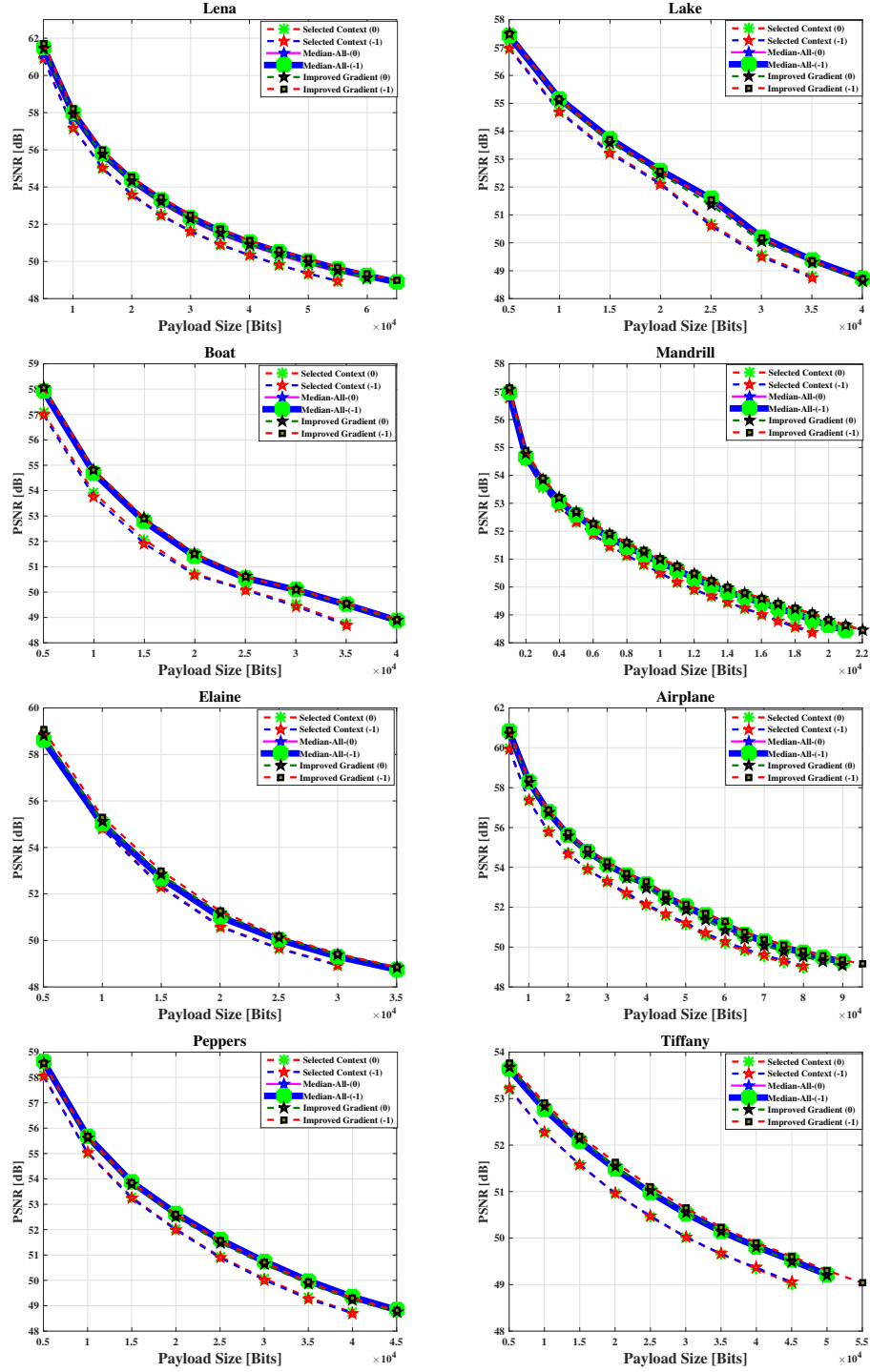


Figure 6.5: Performance comparison of the proposed multi-predictor based reversible data hiding technique with the techniques using selected directional context with 3×3 neighborhood (Section 5.1) and improved gradient based prediction (Section 5.2)

of using multiple predictors in the context of reversible data hiding. Moreover, the performance analysis points to ten of these sixteen predictors as their performances are better than the average performance of all sixteen predictors. Then these ten predictors (being denoted as IR, SDC, HG, ME, MAX, MIN, EGBSW, IGBSW, AGSP, and TGAPP) are combined in a multi-predictor based reversible data hiding technique.

Secondly, several simple strategies for combining multiple predictors are experimentally studied. The median of ten predicted values for selected predictors exhibits better performance as compared against other combination strategies for multiple predictors. Hence, the proposed reversible data hiding technique using median based multi-predictor strategy performs better than several existing reversible data hiding techniques.

Similar to the reversible data hiding techniques in Chapter 5, an adaptive prediction error histogram bin shifting strategy is used in the proposed multi-predictor based reversible data hiding technique for embedding of data bits in the prediction error of a pixel. According to this strategy, either 1 bit or 2 bits of data are embedded in the prediction error of a pixel based on local complexity of the pixel. Increase in local complexity threshold value helps in achieving higher embedding capacity. Experiments are conducted by embedding data bits in pixels belonging to two pairs of prediction error histogram bins. Experiments reveal that significant difference in performance is not observed for the selection of histogram bin pair as either $(-1, 0)$ or $(0, 1)$.

It is to be noted that the proposed reversible data hiding technique, as well as other reversible data hiding techniques in literature, are fragile in nature. These techniques are not robust at all against any kind of tampering of the marked image. Hence, experiments demonstrating the robustness analysis of the proposed technique have not been carried out. This technique can be used in satellite, medical, military and banking applications where recovery of the original images are very important. Normally, these fragile reversible data hiding techniques are designed for ensuring the integrity of an image.

Chapter 7

Application: Prediction Error Expansion Based Reversible Data Hiding for Detecting Tampered Cheque Image

Reversible data hiding is useful in an application where recovery of the original cover image is required along with extraction of the hidden data. Being fragile in nature, such a technique is mainly used for ensuring integrity of the image. Moreover, recovery of the cover image helps in accurate automated analysis of the original cover image. Hence, reversible data hiding technique is used to detect whether a bank check image has been tampered or not. Few such attempts can be found in [11, 12] in the context of an image based cheque clearing system, namely cheque truncation system (CTS). In [11], a basic difference expansion based reversible data hiding technique [28] is applied to ensure integrity of a bank cheque image. Overflow/underflow problem of the difference expansion based technique is addressed in [12] by suggesting a particular way of pairing of pixel values.

Traditionally, prediction error expansion (PEE) based reversible data hiding techniques perform better than difference expansion based techniques. Few such PEE based techniques are proposed in previous chapters of this thesis. Hence, in this chapter, the proposed reversible data hiding techniques (in Chapters 4, 5

and 6) are applied on a bank cheque image for ensuring integrity of the cheque image. Remainder of this chapter is organized as following: A brief introduction about cheque truncation system (CTS) is presented in Section 7.1. A method for detecting tampered cheque images using PEE based reversible data hiding techniques is explored in Section 7.2. Experimental results are presented in Section 7.3. Finally, this work is summarized in Section 7.4.

7.1 Brief Introduction of Cheque Truncation System

Cheque truncation system (CTS) [164] is an image based cheque clearing environment involving multiple banks. This system of image based cheque clearing is prevalent in India. A broad overview of CTS environment for cheque clearing is presented in this section. Before proceeding further, let few related terminologies be clarified. A presenting bank refers to a bank where a customer presents the physical cheque leaf with a request for clearing the cheque. A drawee bank refers to a bank where the issuer of the cheque maintains her account. A clearing house refers to the intermediary which facilitates the cheque clearing among the presenting bank and the drawee bank. Now, the steps of cheque clearing in CTS environment are presented as following:

1. A customer presents a physical cheque leaf at a counter of the presenting bank.
2. At the presenting bank, the physical cheque leaf is scanned to capture its image in gray scale. Movement of the physical cheque is truncated at the presenting bank. Hence, this system is termed as cheque truncation system.
3. The scanned cheque image is sent to the clearing house interface at the presenting bank.
4. Through the clearing house interface, the cheque image is forwarded to the clearing house. Subsequently, the image is forwarded to the appropriate

7.1 Brief Introduction of Cheque Truncation System

drawee bank, where the signature of the issuer and sufficiency of the amount in the issuer's bank account are verified.

5. Upon successful verification of the above information, the account of the issuer is debited and the positive response is returned to the presenting bank through the clearing house.
6. Once the presenting bank receives the positive response from the drawee bank, the customer is paid the amount (or her account is credited).

In CTS environment, communication of a cheque image from the clearing house interface of the presenting bank to the drawee bank (through the clearing house) is secured using a public key-enabled infrastructure. But it lacks end-to-end security from the scanner at the presenting bank to the verification module at the payee bank. Hence, an unencrypted original cheque image is available at various processing nodes at the presenting bank before it is put to the clearing house interface. This leaves the cheque image susceptible to image tampering attacks. One malicious insider can edit the image to alter the payee name and/or to inflate the amount. Such alteration in the cheque image can be detected at the drawee bank through a fragile and reversible data hiding technique. A secret data is inserted into the cheque image at the presenting bank at the time of scanning the physical cheque. The drawee bank extracts the secret data. If the extracted data matches perfectly with the inserted data, then the cheque image at the drawee bank can be considered as genuine. Otherwise, the cheque image can be considered as tampered. Moreover, reversible nature of the data hiding technique ensures restoration of the original cheque image at the drawee bank. Automated analysis of the cheque image for detecting the security features of a cheque leaf is possible in such a scenario.

Ensuring of integrity of a cheque image based on reversible data hiding is also essential for another use case. In this use case, a customer takes a photo of the physical cheque leaf using her mobile and sends it to her bank with a request for clearing. Moreover, the issuance of a physical cheque can be abolished and a customer can download an image of a cheque from her online banking service [165]. In this case too, ensuring integrity of the cheque image is paramount.

7.2 Prediction Error Expansion Based Reversible Data Hiding Techniques for Detecting Cheque Image Tampering

Reversible data hiding techniques, which are traditionally fragile in nature, are mainly used for ensuring integrity of the original images. Hence, reversible data hiding techniques can be used for ensuring integrity of the bank cheque images [11, 12]. Several novel prediction error expansion/shifting based reversible data hiding techniques have been proposed in the previous chapters of this thesis. In this section, following of these proposed techniques are applied for detecting tampered cheque images:

1. Random distribution of reference pixel based reversible data hiding (Chapter 4) with block size 4×4 and 25% of pixels in an important portion as reference pixels. This technique is executed using a threshold on prediction error T_{PE} as 1.
2. Reversible data hiding with selected directional context based prediction using eight neighborhood (Section 5.1) with prediction error histogram bins having prediction errors as either -1 or 0 for embedding.
3. Reversible data hiding with selected directional context based prediction using improved gradient (Section 5.2) with prediction error histogram bins having prediction errors as either -1 or 0 for embedding.
4. Reversible data hiding using multiple predictors (Chapter 6) with prediction error histogram bins having prediction errors as either -1 or 0 for embedding.

Moreover, these above techniques are used with single bit embedding (i.e., local complexity threshold $T_{lc} = 0$). Increasing embedding capacity is not an objective for this task. Hence, single bit embedding is carried out for the above listed reversible data hiding techniques. In this work, similar to the techniques in [11, 12], an attempt is made to detect tampering of seven important portions of a cheque image. These seven important portions contain: (i) date, (ii) payee name, (iii) legal amount, (iv) courtesy amount, (v) account number, (vi) signature and

7.2 Prediction Error Expansion Based Reversible Data Hiding Techniques for Detecting Cheque Image Tampering

(vii) MICR code. At first, a template based approach is adopted for identifying these rectangle-shaped important portions. Then, the above proposed reversible data hiding techniques are applied to embed a secret binary data string to each of these important portions. An auxiliary data string is generated as part of embedding, which is also inserted in the pixels of the unimportant portions of the cheque image. These auxiliary data string carries necessary information for accurate extraction of inserted secret data and recovery of original pixel values in the important portions. This embedding of the secret data as well as auxiliary information is carried out during acquisition of the cheque image from a physical cheque leaf. At the drawee bank, the embedded secret data is extracted from the marked cheque image. Accurate extraction of the secret data ensures that the important portions of the cheque image have not been tampered. Otherwise, it can be understood by the drawee bank that the important portions of the image have been tampered. Then, the cheque image is discarded and the drawee bank notifies the presenting bank about tampering of the presented cheque image. Moreover, reversible nature of the proposed data hiding techniques ensures restoration of the original cheque image pixel values at the important portions. This may help in automated analysis of the cheque image at the drawee bank.

It is to be noted that the important portions in the bank cheque image are not square in shape. Hence, the proposed B-tree triangular decomposition based reversible data hiding technique (Chapter 3) has not been applied for ensuring the integrity of the cheque images.

7.2.1 Template Based Identification of Important Portions

As it has been already discussed above, template based identification of the important portions is carried out for both of embedding and extraction processes. A template is conceived as a binary mask of same size of the cheque image. The locations in the binary mask corresponding to important portions have pixels with value 1. Other locations in the binary mask denoting unimportant portions have pixels with value 0. A sample template for the cheque image in figure 7.1 is presented in figure 7.2.

7.2 Prediction Error Expansion Based Reversible Data Hiding Techniques for Detecting Cheque Image Tampering

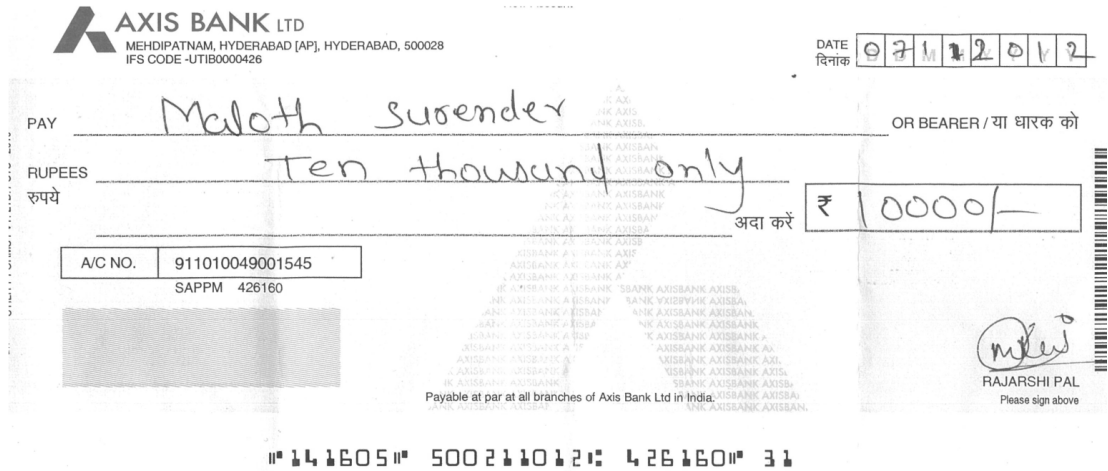


Figure 7.1: An original cheque image

Every bank may not follow the exact same placement of above mentioned important portions in its cheque. Therefore, different templates are used for different banks for identifying these important regions. For example, another template for the cheque image in figure 7.3 is presented in figure 7.4.

7.2.2 Embedding

Embedding process of the secret data into the cheque image is described in this section. This process comprises of following steps:

1. Each rectangular important portion is treated as an image. The data bits are inserted into the prediction errors in the pixels of important portions based on the corresponding embedding procedures for each of the above listed reversible data hiding techniques. Prediction and embedding procedures are carried out based on the procedures mentioned in the corresponding chapters of this thesis. Let maximum N_i bits can be embedded in the i -th important portion of a cheque image using a reversible data hiding technique. Then, total number of bits being embedded in all seven important portions is $N = \sum_{i=1}^7 N_i$. It is to be noted that embedding is carried out in pixels, where overflow/underflow has not occurred.

7.2 Prediction Error Expansion Based Reversible Data Hiding Techniques for Detecting Cheque Image Tampering

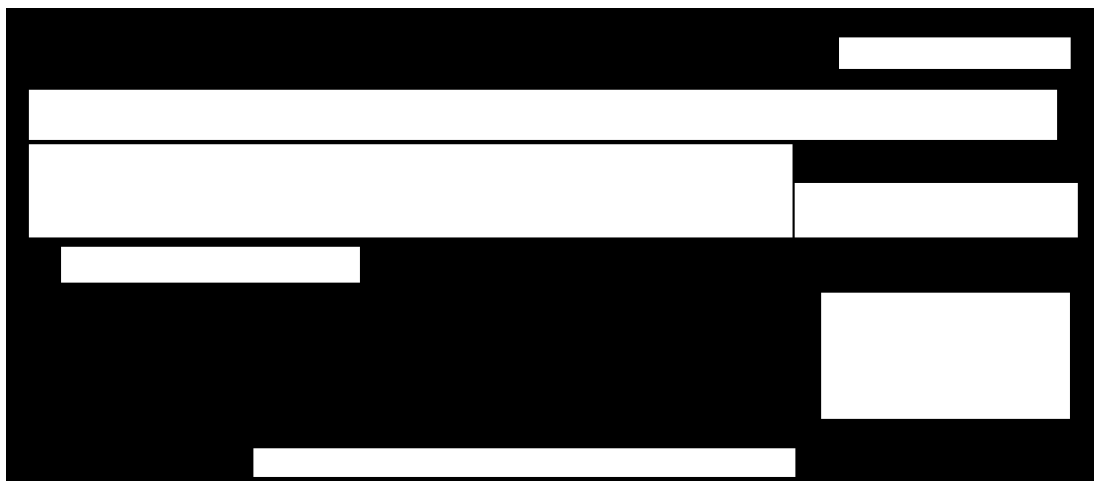


Figure 7.2: Template for the cheque image in figure 7.1



Figure 7.3: An original cheque image

7.2 Prediction Error Expansion Based Reversible Data Hiding Techniques for Detecting Cheque Image Tampering

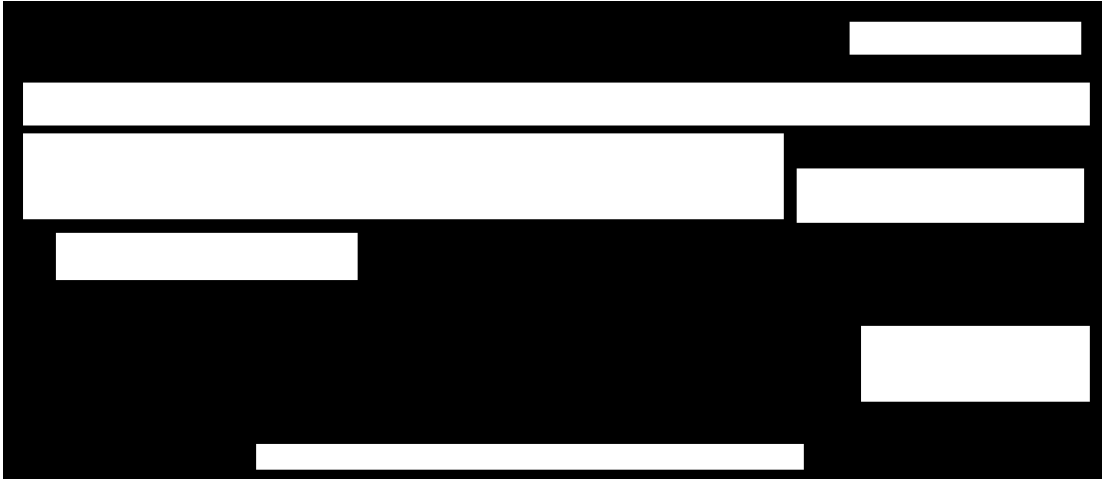


Figure 7.4: Template for the cheque image in figure 7.3

2. Auxiliary information is required for accurate extraction of the secret data and restoration of the original pixel values at the important portions of the image. Details of auxiliary information can be found in Section 3.2.4 and Section 5.1.2.3. It mainly consists of parameters associated with the proposed reversible data hiding technique and a compressed overflow/underflow location map. Location of the last embedded pixel is not stored as embedding process covers every pixel in an important portion. Auxiliary information is generated from each of the seven important portions. Then, auxiliary information strings from all of these important portions are concatenated one after another to derive an overall auxiliary information string. This auxiliary information is inserted into the pixels of the unimportant regions of cheque image by following a raster scan ordering of these pixels. Least significant bit (LSB) substitution method is used to embed the bits of the auxiliary information in the LSBs of these pixel values. But unlike the descriptions in Section 3.2.4 and Section 5.1.2.3, the actual LSBs of these pixels are not appended at the end of the secret data. This is because the length of the derived auxiliary information is relatively high as compared to the achieved embedding capacity in these important portions. As a result, the actual LSBs of these pixels in the unimportant portions are lost forever. Hence, recovery of the original pixel values in the unimportant portions is

not possible.

7.2.3 Extraction

Initially, the auxiliary information is extracted from the LSBs of the pixel values in unimportant portions by visiting them in raster scan sequence until the entire auxiliary information is extracted. Auxiliary information for each of seven important portions are segregated. Using the auxiliary information for an important portion, the secret data bits are extracted from the pixels in the important portion. Simultaneously, the original pixel values at those important portions are restored back. The extraction process of each of the listed reversible data hiding schemes are described in the corresponding chapters of this thesis. Only exception, in this context, is that the pixel values at the unimportant regions can not be obtained back.

The extracted secret data is compared with the embedded secret data. If it is extracted accurately, i.e., without an error in a single bit, it is concluded that the important portions of the cheque image have not been tampered. Otherwise, it is concluded that at least one important portion in the cheque image has been tampered. This conclusion is possible because modification of a marked image destroys the hidden data as per a fragile data hiding technique.

7.3 Experimental Results

Ten bank cheque images are used in the reported experiments in this section. These cheque images are shown in figure 7.5. Each image is of size 668×1537 .

To compare the performances of the used reversible data hiding techniques (as listed in Section 7.2), each of these techniques is applied to embed a secret data into each of the test cheque images. Length of the embedded secret bit string is equal to the achievable embedding capacity of the concerned reversible data hiding technique on each cheque image. In each case, a randomly generated secret bit string having the necessary length is used as secret data. In each case, maximum possible amount of secret data is embedded in order to cover the entirety of seven

important portions in the cheque image. Four marked cheque images for the original image in figure 7.1 are presented in figure 7.6 for the applied reversible data hiding techniques.

For each test cheque image and each applied reversible data hiding technique, a secret data of length equaling to the achievable embedding capacity is embedded in the pixels at the important portions. It is to be noted that pixels, where underflow/overflow might occur, have been excluded from this embedding process. Peak signal-to-noise ration (PSNR) between a cover image and its corresponding marked image is noted for each test cheque image using each of the comparing prediction error expansion based reversible data hiding techniques. These PSNR values are presented in table 7.1. Based on the reported PSNR values, it can be observed that random distribution of reference pixel based reversible data hiding technique (Chapter 4) performs better than other three comparing techniques in majority of the cheque images. But it can also be observed based on the average PSNR values (over 10 test images) that selected directional context using 8-neighborhood based reversible data hiding technique (Section 5.1) also performs almost as good as random distribution of reference pixel based technique (Chapter 4) on cheque images. Moreover, the lengths of the secret data string for these cases are also presented in table 7.2. PSNR values signify the similarity between the original cheque images and their corresponding marked cheque images. Hence, a reversible data hiding technique, which provides the highest PSNR value (table 7.1), draws the attention here. Length of secret data string is considered to cover the entirety of seven important portions in each case. Increase the embedding capacity is not an objective here. Hence, the lengths in table 7.2 are mentioned just for reporting purpose.

It is also experimentally observed that accurate extraction of the embedded data string is not possible from a tampered marked image. In order to report these experiments, several tampered marked images are created from the actual marked images. For illustration, four such tampered marked images are presented in figure 7.7. These four tampered marked images correspond to the four marked images in figure 7.6 (one-to-one correspondence). In the top-left image in figure 7.7, payee name is altered. In the top-right image in figure 7.7, amount is modified. In the

Table 7.1: PSNR values between a cover image and corresponding marked image at maximum achievable embedding capacity for the comparing techniques

Image	Random Distribution Based	Eight Neighborhood Based	Improved Gradient Based	Multi Predictor Based
Cheque 1	53.54	51.88	50.96	51.11
Cheque 2	53.33	51.85	50.96	51.09
Cheque 3	53.52	52.02	51.02	51.11
Cheque 4	53.50	51.99	51.05	51.15
Cheque 5	51.10	53.12	53.00	53.16
Cheque 6	50.00	52.32	52.11	52.17
Cheque 7	49.95	52.34	52.14	52.21
Cheque 8	53.59	52.10	51.56	51.65
Cheque 9	54.08	52.23	51.46	51.53
Cheque 10	50.49	53.06	53.40	53.41
Average	52.31	52.29	51.77	51.86

bottom-left image in figure 7.7, name and amount are modified. In the bottom-right image in figure 7.7, name and date are modified. As accurate extraction of the embedded data is not possible from such images, it can be concluded that these marked images have been tampered.

7.4 Summary

In this chapter, ensuring integrity of bank cheque image in the context of cheque truncation system (CTS) is considered as an application of reversible data hiding. Four prediction error expansion based reversible data hiding techniques (which are proposed in Chapters 4, 5 and 6) are applied in this context. Similar usage of these proposed reversible data hiding techniques can be perceived in the context of mobile cheque deposit and issuance of digital cheque.

Being fragile in nature, these proposed techniques are instrumental in detection of tampering in the marked cheque images. Moreover, reversibility of these

Table 7.2: Length of secret data string (achievable embedding capacity) for various test images using each comparing techniques

Image	Random Distribution Based	Eight Neighborhood Based	Improved Gradient Based	Multi Predictor Based
Cheque 1	30316	283085	81047	72512
Cheque 2	30913	285298	77549	68566
Cheque 3	28114	291425	65086	58438
Cheque 4	31730	286494	74635	65562
Cheque 5	31620	112533	77268	88052
Cheque 6	17958	87995	31194	30170
Cheque 7	13504	79889	23618	23718
Cheque 8	63010	283244	67579	61108
Cheque 9	47509	302003	51867	46470
Cheque 10	74358	102646	131631	121578
Average	36903	211461	68147	63617

techniques restore the original pixel values at the important portions of the cheque images.

Based on the reported PSNR values, it can be observed that random distribution of reference pixel based reversible data hiding technique (Chapter 4) performs better than other three comparing techniques in majority of the cheque images. But it can also be observed based on the average PSNR values (over 10 test images) that selected directional context using 8-neighborhood based reversible data hiding technique (Section 5.1) also performs almost as good as random distribution of reference pixel based technique (Chapter 4) on cheque images.

7.4 Summary



Figure 7.5: Original bank cheque images (Cheques numbered 1 to 10 in row-major order)

7.4 Summary



Figure 7.6: Marked images for cheque image in figure 7.1 using various reversible data hiding techniques: random distribution of reference pixel based (top-left), eight neighborhood based (top-right), improved gradient based (bottom-left), and multi-predictor based (bottom-right)



Figure 7.7: Tampered images from the marked images in figure 7.6

Chapter 8

Conclusion and Future Work

In this chapter, concluding remarks are presented on the research works being presented in the thesis. This chapter is organized in following two sections: A summary of the contributions of the works in this thesis is presented in Section 8.1. Finally, a few glimpses on future research directions are placed in Section 8.2.

8.1 Summary of Contributions

Reversible data hiding is a special kind of data hiding, where the cover image can be restored from the marked image along with extraction of the hidden data. Over past two decades, several reversible data hiding techniques have been introduced by the researchers. Among these techniques, prediction error expansion based techniques stand out prominently due to their superior performances in comparison to other genres of reversible data hiding techniques. In a prediction error expansion based technique, a pixel value is predicted. The prediction error of a pixel is expanded to embed secret data bits. Scope of the research work, in this thesis, is prediction error expansion based techniques. Several novel prediction error expansion/prediction error histogram bin shifting based reversible data hiding techniques are presented in this thesis. A summary of these works is presented in this section.

Inspired by the works using a set of reference pixels for predicting values of other pixels (and more specifically, due to the Delaunay triangulation (DT) based

technique [119]), a novel reversible data hiding technique based on a B-tree triangular decomposition based method is proposed in this work. The original image is recursively decomposed into a set of right-angled triangles using B-tree triangular decomposition method. Further decomposition of a triangle is stopped if (i) the difference between the maximum and the minimum pixel values in a triangle (also considering the sides of the triangle) is within a threshold or (ii) the triangle is very small in size. The triangle division is stored in a B-tree structure. Finally, values of the pixels, which fall inside or on the three sides of a triangle, are predicted using the pixel values at the vertices of the triangle as the reference pixels using an interpolation technique. An adaptive embedding strategy is used for embedding either 1-bit or 2-bit data into the expanded prediction error of a pixel based on an estimated local complexity of a triangle. Local complexity of the pixels in a triangle is approximated by considering the differences among the vertex pixel values in the triangle. As these vertex pixel values are not modified during embedding, the same vertex pixel values are available during extraction phase too. Hence, local complexity of the pixels in a triangle is approximated using these vertex pixel values. Moreover, the proposed work visits the triangles (local regions) in the increasing order of their local complexities. Experimental results demonstrate superiority of the proposed technique (as measured using the peak signal-to-noise ratio between a cover image and the corresponding marked image) as compared to several existing techniques, including another reference pixel based method in [119]. In the reversible data hiding technique in [119], the reference pixels are obtained by partitioning the image into a set of Delaunay triangles. These triangles are constructed by following the Delaunay triangulation property, where the circumcircle of a triangle does not include any other vertex. On the contrary, the proposed B-tree triangular decomposition method in Chapter 3 partitions the image into a set of triangles based on the homogeneity of pixel values in the triangle. Hence, the pixel values in such a triangle are more homogeneous as compared to a Delaunay triangle. This justifies the superior performance of the proposed B-tree triangular decomposition based reversible data hiding technique as compared to Delaunay triangulation based technique [119]. Moreover, the performance of the proposed technique is also noted by varying the local complexity threshold in the context of adaptive embedding. The adaptive

embedding (based on local complexity of a pixel) is useful for embedding large payload. Single bit embedding can be used for small payload, but it achieves less distortion as compared to adaptive embedding.

It is further investigated in Chapter 4 whether a structured method (like the above mentioned B-tree triangular decomposition) can be avoided to decide a set of reference pixels. In this context, a novel distribution mechanism of reference pixels is proposed. According to the proposed reference pixel distribution strategy, an image is partitioned into non-overlapping square blocks of same size. A fixed number of reference pixels are distributed among these blocks. Number of reference pixels being distributed in a block is kept proportional to an estimation of roughness of the pixels in the block. In this context, variance of the pixel values in a block is used to estimate the roughness of the block. Once the number of reference pixels in a block is obtained, those many reference pixels are randomly distributed in the block. Then, the values of these reference pixels within a certain distance from a non-reference pixel are used for predicting the non-reference pixel value using a weighted median based approach. An adaptive embedding strategy is used for embedding either 1-bit or 2-bit data into the expanded prediction error of a pixel based on an estimated local complexity of a non-reference pixel. Performance of the proposed technique is experimentally observed for different percentages of reference pixels (25%, 50% and 75% of total number of pixels in an image) and for different block sizes (2×2 , 4×4 and 8×8). Based on these experiments, it can be observed that the best results are obtained using 25% of pixels in the image as reference pixels and a block size of either 2×2 or 4×4 .

A novel reversible data hiding technique is also proposed in Section 5.1 based on a selected directional context based pixel prediction scheme using 8-neighborhood of a pixel. Averages of a pair of neighboring pixel values (in a 3×3 neighborhood) in each of horizontal, vertical, diagonal and anti-diagonal directions are estimated. These average values attempt to predict the center pixel value from the context of respective directions. The proposed work selects few of these four directions which provide the context for pixel prediction. For this purpose, the absolute differences between values of a pair of neighboring pixels in horizontal, vertical, diagonal and anti-diagonal directions are computed. Then, a

novel context selection strategy is proposed by considering the computed differences. This strategy considers the direction (i.e., context) with the least difference value and incrementally checks the possibility of adding few more directions in the ascending order of these difference values. Finally, the pixel value is predicted using the previously computed average values in the selected contexts. An adaptive prediction error histogram bin shifting based embedding strategy is used here. The histogram bin shifting based strategy is extended to adaptively embed either one bit or two bits of data depending on local complexity of the pixel. The local complexity is estimated by considering the variance of few surrounding pixels. The adaptive histogram bin shifting strategy is applied on pairs of histogram bins corresponding to prediction error pairs as either $(-1, 0)$ or $(0, 1)$. Experiments reveal that significant difference is not observed for the selection of histogram bin pair as either $(-1, 0)$ or $(0, 1)$.

A novel reversible data hiding technique is also proposed in Section 5.2 using a gradient based pixel prediction strategy. Here, the gradient around a pixel is estimated using a 5×5 neighborhood. This gradient-based prediction is an extension of the gradient based prediction technique in [25], which used a 4×4 neighborhood. In this proposed work, gradients in horizontal, vertical, diagonal and anti-diagonal directions are computed using a 5×5 neighborhood. Moreover, linear predictors using a set of neighborhood pixels are considered for each of the above four directions. Then, a weighted average of two predicted values in the directions of the least two gradients estimates the final predicted value. Finally, a novel prediction error histogram bin shifting based reversible data hiding technique is proposed using this gradient based prediction. The histogram bin shifting based strategy is extended to adaptively embed either one bit or two bits of data depending on local complexity of the pixel. Pair of histogram bins corresponding to prediction errors as either $(-1, 0)$ or $(0, 1)$ are used in this work. Experiments reveal that significant difference is not observed for the selection of histogram bin pairs corresponding to prediction errors as either $(-1, 0)$ or $(0, 1)$.

Next, an experimental study is conducted to assess the performance of several existing as well as newly proposed (as above) neighborhood based and gradient based predictors (Chapter 6). The experiment reveals that none of the predictors are perfect in predicting the pixel values in varied neighborhood contexts.

But some of these predictors perform better than rest of the others. Then, a multi-predictor based strategy is proposed by considering a set of good predictors, which have been identified by the experiment. Finally, similar to previous two approaches, adaptive prediction error histogram bin shifting based strategy is adopted to consider variable number of bits for embedding in a pixel. In the context of combining outputs of multiple predictors, following strategies are experimentally evaluated to find out the best way of combining multiple predicted values: (i) the minimum of the predicted values, (ii) the maximum of the predicted values, (iii) mean of the minimum and the maximum predicted values, (iv) mean of all considered predicted values, and (v) median of all considered predicted values. Finally, experimental results indicate that adaptive prediction error histogram bin shifting with median of all considered predicted values as the final predicted value demonstrates better results than other multi-predictor combination strategies.

At the end, it is demonstrated in Chapter 7 how the proposed reversible data hiding techniques can be used to ensure integrity of bank cheque images in an image based cheque presentation and clearing environment. A predetermined secret data is embedded into a select set of portions in the cheque image. At the drawee bank, the successful extraction of the secret data ensures the integrity of these important portions of the cheque image. The original pixel values are also restored due to the reversible nature of the scheme. Recovery of the original pixel values may be important for automated analysis of the cheque image. The above proposed reversible data hiding techniques (except the B-tree triangular decomposition based reversible data hiding in Chapter 3) are applied to achieve the stated objective in this work. B-tree triangular decomposition can only be applied on a square image. Hence, this B-tree triangular decomposition based reversible data hiding technique cannot be used for this task. But performances of other four proposed techniques are experimentally observed for this task.

It is to be noted that the proposed reversible data hiding techniques, as well as majority of other reversible data hiding techniques in the literature, are fragile in nature. These techniques are not robust at all against any kind of tampering of the marked image. Hence, experiments demonstrating the robustness analysis of the proposed techniques have not been carried out. Normally, these fragile

reversible data hiding techniques are designed for ensuring the integrity of an image.

8.2 Future Research Directions

In this section, a few points are stated as a continuation of this research work in future. These research directions are listed here:

- Similar to B-tree triangular decomposition of an image (Chapter 3) for obtaining a set of reference pixels, other partitioning schemes of the image into non-overlapping regular-shaped regions can be used for pixel prediction and subsequent reversible data hiding. For example, quad-tree decomposition [166] is widely used to recursively partitioning an image into square regions by following a quad-tree structure. Another triangular decomposition scheme is also recently proposed in [167]. Performance of these decomposition schemes for pixel prediction and subsequent reversible data hiding can be studied.
- Machine learning is widely used for prediction across varied disciplines. Similarly, machine learning models may be trained to predict a pixel value. Few papers on using machine learning models in the context of reversible data hiding exist in literature [122, 123]. But this direction of research is not explored much yet. Hence, machine learning based pixel value prediction in the context of reversible data hiding can be studied at depth.
- Traditionally, reversible data hiding techniques (as have been discussed so far in this thesis) are fragile in nature. Slight modification to a pixel value destroys the hidden data. Hence, extraction of hidden data as well as recovery of original pixel values are not possible in such a case. This leads to application of these reversible data hiding techniques for ensuring integrity of a cover media. But similar to traditional watermarking methods, an attempt can be made to develop a robust reversible data hiding technique. Very few attempts can be found in this direction [168].

- Reversible data hiding techniques can be extended to color images too. In this context, the questions are being asked on selection of one or more color channels for data hiding, consideration of all color channels for better prediction, maintaining the intensity in a marked pixel as same as the intensity on the original pixel, etc. Few works are already reported for color images [169, 170, 171]. In future, the domain of reversible data hiding for color images can be investigated further.
- Reversible data hiding for encrypted images is another emerging direction of research [172, 173, 174, 175, 176]. Further exploration of this direction is also required.
- Extension of reversible data hiding research from images to videos appears to be natural. Few works on reversible data hiding in videos have been carried out recently [177]. This area may be explored further, because video is being increasingly used in society for varied purposes.

List of Publications

Journals

1. R. Uyyala, R. Pal and M. V. N. K Prasad. **Reversible data hiding using B-tree triangular decomposition based prediction.** *IET Image Processing*, **13**(11):1986-1997, October 2019. **Impact Factor: 2.004.** **Indexing: Scopus, DBLP, and SCI.**
2. R. Uyyala and R. Pal. **Reversible data hiding based on random distribution of reference pixels and adaptive prediction error expansion** (To be communicated).
3. R. Uyyala and R. Pal. **Reversible data hiding using multiple predictors and adaptive histogram bin shifting** (To be communicated).
4. R. Uyyala and R. Pal. **Reversible data hiding : A Literature Survey** (To be communicated).

Conferences

5. R. Uyyala, R. Pal and M. V. N. K Prasad. **Gradient dependent reversible watermarking with low embedding impact.** *In Proceedings of 3rd International Conference on Signal Processing and Integrated Networks*, pages 184-189, Noida, India, February 2016. **Indexing: Scopus.** **Publisher: IEEE.**

6. R. Uyyala, M. V. N. K Prasad and R. Pal. **Selected context dependent prediction for reversible watermarking with optimal embedding.** *In Proceedings of 1st International Conference on Computer Vision and Image Processing*, pages 35-46, Roorkee, India, December 2016. **Indexing: Scopus and DBLP. Publisher: Springer.**
7. R. Uyyala and R. Pal. **Reversible data hiding based on random distribution of reference pixels.** *In Proceedings of The IEEE Region 10 Symposium*, pages 225-230, Sydney, Australia, July 2018. **Indexing: Scopus. Publisher: IEEE.**
8. R. Uyyala and R. Pal. **Reversible data hiding using improved gradient based prediction and adaptive histogram bin shifting.** *In Proceedings of 7th International Conference on Signal Processing and Integrated Networks*, Noida, India, February 2020. **Indexing: Scopus. Publisher: IEEE.**
9. R. Uyyala and R. Pal. **Reversible data hiding technique using selected context based prediction and adaptive histogram bin shifting** (To be communicated).

References

- [1] F. A. P. PETITCOLAS, R. J. ANDERSON, AND M. G. KUHN. **Information hiding - a survey**. *In proceedings of the IEEE*, **87**(7):1062–1078, July 1999.
- [2] A. KHAN, A. SIDDIQA, S. MUNIB, AND S. A. MALIK. **A recent survey of reversible watermarking techniques**. *Information Sciences*, **279**:251 – 272, September 2014.
- [3] F. HARTUNG AND M. KUTTER. **Multimedia watermarking techniques**. *In proceedings of the IEEE*, **87**(7):1079–1107, July 1999.
- [4] N. F. JOHNSON AND S. JAJODIA. **Exploring steganography: seeing the unseen**. *Computer*, **31**(2):26–34, February 1998.
- [5] D. HOU, W. ZHANG, J. LIU, S. ZHOU, D. CHEN, AND N. YU. **Emerging applications of reversible data hiding**. *In Proceedings of 2nd International Conference on Image and Graphics Processing*, pages 105–109, New York, USA, February 2019.
- [6] H.-T. WU, J. HUANG, AND Y.-Q. SHI. **A reversible data hiding method with contrast enhancement for medical images**. *Journal of Visual Communication and Image Representation*, **31**:146 – 153, August 2015.
- [7] Y. YANG, W. ZHANG, D. LIANG, AND N. YU. **Reversible data hiding in medical images with enhanced contrast in texture area**. *Digital Signal Processing*, **52**:13 – 24, May 2016.
- [8] Y. LIU, X. QU, AND G. XIN. **A ROI-based reversible data hiding scheme in encrypted medical images**. *Journal of Visual Communication and Image Representation*, **39**:51 – 57, August 2016.

- [9] G. GAO, X. WAN, S. YAO, Z. CUI, C. ZHOU, AND X. SUN. **Reversible data hiding with contrast enhancement and tamper localization for medical images.** *Information Sciences*, **385-386**:250 – 265, April 2017.
- [10] B. CARPENTIERI, A. CASTIGLIONE, A. D. SANTIS, F. PALMIERI, AND R. PIZZOLANTE. **One-pass lossless data hiding and compression of remote sensing data.** *Future Generation Computer Systems*, **90**:222 – 239, January 2019.
- [11] S. KOTA AND R. PAL. **Detecting tampered cheque images in cheque truncation system using difference expansion based watermarking.** In *Proceedings of IEEE International Advance Computing Conference*, pages 1041–1047, Gurgaon, India, February 2014.
- [12] M. TEJAWAT AND R. PAL. **Detecting tampered cheque images using difference expansion based watermarking with intelligent pairing of pixels.** In *Proceedings of International Conference on Advanced Computing, Networking and Informatics*, pages 631–641, Bhubaneswar, India, June 2015.
- [13] X. WANG, C. SHAO, X. XU, AND X. NIU. **Reversible data-hiding scheme for 2-D vector maps based on difference expansion.** *IEEE Transactions on Information Forensics and Security*, **2**(3):311–320, September 2007.
- [14] F. PENG, Y. LIU, AND M. LONG. **Reversible watermarking for 2D CAD engineering graphics based on improved histogram shifting.** *Computer-Aided Design*, **49**:42 – 50, April 2014.
- [15] Z. ZHANG AND W. ZHANG. **Reversible steganography: data hiding for covert storage.** In *Proceedings of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pages 753–756, Hong Kong, China, December 2015.
- [16] X. DUAN, K. JIA, B. LI, D. GUO, E. ZHANG, AND C. QIN. **Reversible image steganography scheme based on a U-net structure.** *IEEE Access*, **7**:9314–9323, January 2019.

- [17] D. COLTUC. **On stereo embedding by reversible watermarking.** In *Proceedings of International Symposium on Signals, Circuits and Systems*, pages 1–4, July 2007.
- [18] K.-L. CHUNG, Y.-H. HUANG, P.-C. CHANG, AND H.-Y. M. LIAO. **Reversible data hiding-based approach for intra-frame error concealment in H.264/AVC.** *IEEE Transactions on Circuits and Systems for Video Technology*, **20**(11):1643–1647, November 2010.
- [19] J. LIU, D. HOU, W. ZHANG, AND N. YU. **Reversible Adversarial Examples.** *arXiv*, November 2018.
- [20] D. HOU, W. ZHANG, Z. ZHAN, R. JIANG, Y. YANG, AND N. YU. **Reversible image processing via reversible data hiding.** In *Proceedings of IEEE International Conference on Digital Signal Processing*, pages 427–431, Beijing, China, October 2016.
- [21] Y. XIA AND H.-T. WU. **Reversible data hiding in medical image for contrast enhancement of ROI.** In *Proceedings of 12th International Conference on Computational Intelligence and Security*, pages 633–637, Wuxi, China, December 2016.
- [22] C. W. HONSINGER, P. W. JONES, M. RABBANI, AND J. C. STOFFEL. **Lossless recovery of an original image containing embedded data.** *U. S. Patent, US 6278791B1*, August 2001.
- [23] D. M. THODI AND J. J. RODRIGUEZ. **Expansion embedding techniques for reversible watermarking.** *IEEE Transactions on Image Processing*, **16**(3):721–730, March 2007.
- [24] R. DISTASI, M. NAPPI, AND S. VITULANO. **Image compression by B-tree triangular coding.** *IEEE Transactions on Communications*, **45**(9):1095–1100, September 1997.

- [25] I.-C. DRAGOI, D. COLTUC, AND I. CACIULA. **Gradient based prediction for reversible watermarking by difference expansion.** In *Proceedings of the 2nd ACM Workshop on Information Hiding and Multimedia Security*, pages 35–40, New York, USA, June 2014.
- [26] Y.-Q. SHI, X. LI, X. ZHANG, H.-T. WU, AND B. MA. **Reversible data hiding: advances in the past two decades.** *IEEE Access*, 4:3210–3237, May 2016.
- [27] J. TIAN. **Reversible watermarking by difference expansion.** In *Proceedings of Workshop on Multimedia Security at ACM Multimedia*, December 2002.
- [28] J. TIAN. **Reversible data embedding using a difference expansion.** *IEEE Transactions on Circuits and Systems for Video Technology*, 13(8):890–896, August 2003.
- [29] H.-W. TSENG AND C.-C. CHANG. **An extended difference expansion algorithm for reversible watermarking.** *Image and Vision Computing*, 26(8):1148 – 1153, August 2008.
- [30] A. M. ALATTAR. **Reversible watermark using difference expansion of triplets.** In *Proceedings of International Conference on Image Processing*, pages 501–504, Barcelona, Spain, September 2003.
- [31] C.-C. LIN AND N.-L. HSUEH. **A lossless data hiding scheme based on three-pixel block differences.** *Pattern Recognition*, 41(4):1415 – 1425, April 2008.
- [32] A. M. ALATTAR. **Reversible watermark using the difference expansion of a generalized integer transform.** *IEEE Transactions on Image Processing*, 13(8):1147–1156, August 2004.
- [33] Y. QIU, Z. QIAN, AND L. YU. **Adaptive reversible data hiding by extending the generalized integer transformation.** *IEEE Signal Processing Letters*, 23(1):130–134, January 2016.

- [34] A. M. ALATTAR. **Reversible watermark using difference expansion of quads.** In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 377–380, Montreal, Canada, May 2004.
- [35] S. WENG, Y. ZHAO, J.-S. PAN, AND R. NI. **A novel reversible watermarking based on an integer transform.** In *Proceedings of IEEE International Conference on Image Processing*, pages 241–244, September 2007.
- [36] O. M. AL-QERSHI AND B. E. KHOO. **High capacity data hiding schemes for medical images based on difference expansion.** *Journal of Systems and Software*, **84**(1):105 – 112, January 2011.
- [37] J.-Y. HSIAO, K.-F. CHAN, AND J. M. CHANG. **Block-based reversible data embedding.** *Signal Processing*, **89**(4):556 – 569, April 2009.
- [38] H. J. KIM, V. SACHNEV, Y. Q. SHI, J. NAM, AND H.-G. CHOO. **A novel difference expansion transform for reversible data embedding.** *IEEE Transactions on Information Forensics and Security*, **3**(3):456–465, September 2008.
- [39] O. M. AL-QERSHI AND B. E. KHOO. **Two-dimensional difference expansion (2D-DE) scheme with a characteristics-based threshold.** *Signal Processing*, **93**(1):154 – 162, January 2013.
- [40] O. M. AL-QERSHI AND B. E. KHOO. **Controlling hiding capacity using image characteristics with a 2D-DE data hiding scheme.** *International Journal of Electronics and Communications*, **68**(4):346 – 350, April 2014.
- [41] Z. NI, Y.-Q. SHI, N. ANSARI, AND W. SU. **Reversible data hiding.** *IEEE Transactions on Circuits and Systems for Video Technology*, **16**(3):354–362, March 2006.
- [42] K.-L. CHUNG, Y.-H. HUANG, W.-M. YAN, AND W.-C. TENG. **Distortion reduction for histogram modification-based reversible data hiding.** *Applied Mathematics and Computation*, **218**(9):5819 – 5826, January 2012.

- [43] Z. PAN, S. HU, X. MA, AND L. WANG. **Reversible data hiding based on local histogram shifting with multilayer embedding.** *Journal of Visual Communication and Image Representation*, **31**:64 – 74, August 2015.
- [44] P. NAGARJU, R. NASKAR, AND R. S. CHAKRABORTY. **Improved histogram bin shifting based reversible watermarking.** In *Proceedings of International Conference on Intelligent Systems and Signal Processing*, pages 62–65, Gujarat, India, March 2013.
- [45] Z. ZHAO, H. LUO, Z.-M. LU, AND J.-S. PAN. **Reversible data hiding based on multilevel histogram modification and sequential recovery.** *International Journal of Electronics and Communications*, **65**(10):814 – 826, October 2011.
- [46] Y.-Y. TSAI, D.-S. TSAI, AND C.-L. LIU. **Reversible data hiding scheme based on neighboring pixel differences.** *Digital Signal Processing*, **23**(3):919 – 927, May 2013.
- [47] X. ZENG, Z. LI, AND L. PING. **Reversible data hiding scheme using reference pixel and multi-layer embedding.** *International Journal of Electronics and Communications*, **66**(7):532 – 539, July 2012.
- [48] W.-L. TAI, C.-M. YEH, AND C.-C. CHANG. **Reversible data hiding based on histogram modification of pixel differences.** *IEEE Transactions on Circuits and Systems for Video Technology*, **19**(6):906–910, June 2009.
- [49] W. WANG, J. YE, T. WANG, AND W. WANG. **Reversible data hiding scheme based on significant-bit-difference expansion.** *IET Image Processing*, **11**(11):1002–1014, November 2017.
- [50] H. NYEEM. **Reversible data hiding with image bit-plane slicing.** In *Proceedings of 20th International Conference of Computer and Information Technology*, Dhaka, Bangladesh, December 2017.

- [51] L.-C. HUANG, L.-Y. TSENG, AND M.-S. HWANG. **A reversible data hiding method by histogram shifting in high quality medical images.** *Journal of Systems and Software*, **86**(3):716 – 727, March 2013.
- [52] E. FATIMA AND S. ISLAM. **A new high capacity and reversible data hiding technique for images.** In *Proceedings of 15th International Conference on Information Systems Security*, pages 290–304, Hyderabad, India, December 2019.
- [53] X. LI, B. LI, B. YANG, AND T. ZENG. **General framework to histogram-shifting-based reversible data hiding.** *IEEE Transactions on Image Processing*, **22**(6):2181–2191, June 2013.
- [54] D. M. THODI AND J. J. RODRIGUEZ. **Reversible watermarking by prediction-error expansion.** In *Proceedings of 6th IEEE Southwest Symposium on Image Analysis and Interpretation*, pages 21–25, Lake Tahoe, USA, March 2004.
- [55] M. J. WEINBERGER, G. SEROUSSI, AND G. SAPIRO. **The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS.** *IEEE Transactions on Image Processing*, **9**(8):1309–1324, August 2000.
- [56] W. HONG, T.-S. CHEN, AND C.-W. SHIU. **Reversible data hiding for high quality images using modification of prediction errors.** *Journal of Systems and Software*, **82**(11):1833 – 1842, November 2009.
- [57] Y. HU, H.-K. LEE, AND J. LI. **DE-based reversible data hiding with improved overflow location map.** *IEEE Transactions on Circuits and Systems for Video Technology*, **19**(2):250–260, February 2009.
- [58] D. COLTUC. **Improved embedding for prediction-based reversible watermarking.** *IEEE Transactions on Information Forensics and Security*, **6**(3):873–882, September 2011.

- [59] C.-C. CHANG, Y.-H. HUANG, H.-Y. TSAI, AND C. QIN. **Prediction-based reversible data hiding using the difference of neighboring pixels.** *International Journal of Electronics and Communications*, **66**(9):758 – 766, September 2012.
- [60] X. GUI, X. LI, AND B. YANG. **A high capacity reversible data hiding scheme based on generalized prediction-error expansion and adaptive embedding.** *Signal Processing*, **98**:370 – 380, May 2014.
- [61] X. WU AND N. MEMON. **Context-based, adaptive, lossless image coding.** *IEEE Transactions on Communications*, **45**(4):437–444, April 1997.
- [62] J. KNEZOVIC AND M. KOVAC. **Gradient based selective weighting of neighboring pixels for predictive lossless image coding.** In *Proceedings of the 25th International Conference on Information Technology Interfaces*, pages 483–488, Cavtat, Croatia, June 2003.
- [63] S. S. KAMATH, P. APARNA, AND A. ANTONY. **Gradient-oriented directional predictor for HEVC planar and angular intra prediction modes to enhance lossless compression.** *International Journal of Electronics and Communications*, **95**:73 – 81, October 2018.
- [64] Y. CHEN, D. HUANG, G. MA, AND J. WANG. **Gradient-based directional predictor for reversible data hiding.** In *Proceedings of 13th IEEE Conference on Industrial Electronics and Applications*, pages 1553–1556, Wuhan, China, May 2018.
- [65] C.-H. YANG AND M.-H. TSAI. **Improving histogram-based reversible data hiding by interleaving predictions.** *IET Image Processing*, **4**(4):223–234, August 2010.
- [66] H.-W. TSENG AND C.-P. HSIEH. **Prediction-based reversible data hiding.** *Information Sciences*, **179**(14):2460 – 2469, June 2009.
- [67] C.-F. LEE, H.-L. CHEN, AND H.-K. TSO. **Embedding capacity raising in reversible data hiding based on prediction of difference expansion.** *Journal of Systems and Software*, **83**(10):1864 – 1872, October 2010.

- [68] C. QIN, C.-C. CHANG, AND L.-T. LIAO. **An adaptive prediction-error expansion oriented reversible information hiding scheme.** *Pattern Recognition Letters*, **33**(16):2166 – 2172, December 2012.
- [69] V. SACHNEV, H. J. KIM, J. NAM, S. SURESH, AND Y. Q. SHI. **Reversible watermarking algorithm using sorting and prediction.** *IEEE Transactions on Circuits and Systems for Video Technology*, **19**(7):989–999, July 2009.
- [70] C.-F. LEE AND H.-L. CHEN. **Adjustable prediction-based reversible data hiding.** *Digital Signal Processing*, **22**(6):941 – 953, December 2012.
- [71] J. WANG, J. NI, AND Y. HU. **An efficient reversible data hiding scheme using prediction and optimal side information selection.** *Journal of Visual Communication and Image Representation*, **25**(6):1425 – 1431, August 2014.
- [72] G. COATRIEUX, W. PAN, N. CUPPENS-BOULAHIA, F. CUPPENS, AND C. ROUX. **Reversible watermarking based on invariant image classification and dynamic histogram shifting.** *IEEE Transactions on Information Forensics and Security*, **8**(1):111–120, January 2013.
- [73] I.-C. DRAGOI AND D. COLTUC. **Adaptive pairing reversible watermarking.** *IEEE Transactions on Image Processing*, **25**(5):2420–2422, May 2016.
- [74] C. DRAGOI AND D. COLTUC. **Improved rhombus interpolation for reversible watermarking by difference expansion.** In *Proceedings of the 20th European Signal Processing Conference*, pages 1688–1692, Bucharest, Romania, August 2012.
- [75] C. SONG, Y. ZHANG, AND G. LU. **Reversible data hiding based on directional prediction and multiple histograms modification.** In *Proceedings of 9th International Conference on Wireless Communications and Signal Processing*, pages 1–6, Nanjing, China, October 2017.

- [76] H. CHEN, J. NI, W. HONG, AND T.-S. CHEN. **High-fidelity reversible data hiding using directionally enclosed prediction.** *IEEE Signal Processing Letters*, **24**(5):574–578, May 2017.
- [77] D.-S. KIM, E.-J. YOON, C. KIM, AND K.-Y. YOO. **Reversible data hiding scheme with edge-direction predictor and modulo operation.** *Journal of Real-Time Image Processing*, **14**:137–145, January 2018.
- [78] S. KIM, X. QU, V. SACHNEV, AND H. J. KIM. **Skewed histogram shifting for reversible data hiding using a pair of extreme predictions.** *IEEE Transactions on Circuits and Systems for Video Technology*, **29**(11):3236–3246, November 2019.
- [79] Y. JIA, Z. YIN, X. ZHANG, AND Y. LUO. **Reversible data hiding based on reducing invalid shifting of pixels in histogram shifting.** *Signal Processing*, **163**:238 – 246, October 2019.
- [80] S. WENG, J.-S. PAN, AND L. ZHOU. **Reversible data hiding based on the local smoothness estimator and optional embedding strategy in four prediction modes.** *Multimedia Tools and Applications*, **76**:13173–13195, June 2017.
- [81] R. M. RAD, K. WONG, AND J.-M. GUO. **Reversible data hiding by adaptive group modification on histogram of prediction errors.** *Signal Processing*, **125**:315 – 328, August 2016.
- [82] I. C. DRAGOI AND D. COLTUC. **Improved pairwise embedding for high-fidelity reversible data hiding.** In *Proceedings of the 26th European Signal Processing Conference*, pages 1412–1416, Rome, Italy, September 2018.
- [83] B. OU, Y. ZHAO, AND R. NI. **Reversible watermarking using optional prediction error histogram modification.** *Neurocomputing*, **93**:67 – 76, September 2012.
- [84] B. OU, X. LI, Y. ZHAO, AND R. NI. **Reversible data hiding based on PDE predictor.** *Journal of Systems and Software*, **86**(10):2700 – 2709, October 2013.

- [85] H. ZHENG, C. WANG, J. WANG, AND S. XIANG. **A new reversible watermarking scheme using the content-adaptive block size for prediction.** *Signal Processing*, **164**:74 – 83, November 2019.
- [86] I.-C. DRAGOI AND D. COLTUC. **Local-prediction-based difference expansion reversible watermarking.** *IEEE Transactions on Image Processing*, **23**(4):1779–1790, April 2014.
- [87] I.-C. DRAGOI AND D. COLTUC. **On local prediction based reversible watermarking.** *IEEE Transactions on Image Processing*, **24**(4):1244–1246, April 2015.
- [88] M. LIU, H. S. SEAH, C. ZHU, W. LIN, AND F. TIAN. **Reducing location map in prediction-based difference expansion for reversible image data embedding.** *Signal Processing*, **92**(3):819 – 828, March 2012.
- [89] H. J. HWANG, S. KIM, AND H. J. KIM. **Reversible data hiding using least square predictor via the LASSO.** *EURASIP Journal on Image and Video Processing*, **42**, December 2016.
- [90] J. FAN AND T. CHEN. **Reversible watermarking using enhanced local prediction.** In *Proceedings of IEEE International Conference on Image Processing*, pages 2510–2514, September 2015.
- [91] R. NASKAR AND R. S. CHAKRABORTY. **Reversible watermarking utilising weighted median-based prediction.** *IET Image Processing*, **6**(5):507–520, July 2012.
- [92] R. NASKAR AND R. S. CHAKRABORTY. **Reversible image watermarking through coordinate logic operation based prediction.** In *Proceedings of International Conference on Information Systems Security*, pages 190–203, Kolkata, India, December 2011.
- [93] K.-H. JUNG AND K.-Y. YOO. **Data hiding method using image interpolation.** *Computer Standards and Interfaces*, **31**(2):465 – 470, February 2009.

- [94] C.-F. LEE AND Y.-L. HUANG. **An efficient image interpolation increasing payload in reversible data hiding.** *Expert Systems with Applications*, **39**(8):6712 – 6719, June 2012.
- [95] M. TANG, J. HU, AND W. SONG. **A high capacity image steganography using multi-layer embedding.** *Optik*, **125**(15):3972 – 3976, August 2014.
- [96] M. TANG, J. HU, W. SONG, AND S. ZENG. **Reversible and adaptive image steganographic method.** *International Journal of Electronics and Communications*, **69**(12):1745 – 1754, December 2015.
- [97] T.-C. LU. **An interpolation-based lossless hiding scheme based on message recoding mechanism.** *Optik*, **130**:1377 – 1396, February 2017.
- [98] G. ONO, K. INOUE, K. HARA, AND K. URAHAMA. **Reversible data hiding using maximum and minimum filters for image interpolation.** In *Proceedings of IEEE 6th Global Conference on Consumer Electronics*, pages 1–2, Nagoya, Japan, October 2017.
- [99] T.-C. LU. **Interpolation-based hiding scheme using the modulus function and re-encoding strategy.** *Signal Processing*, **142**:244 – 259, January 2018.
- [100] T. NAHEED, I. USMAN, T. M. KHAN, A. H. DAR, AND M. F. SHAFIQUE. **Intelligent reversible watermarking technique in medical images using GA and PSO.** *Optik*, **125**(11):2515 – 2525, June 2014.
- [101] X. LI, J. LI, B. LI, AND B. YANG. **High-fidelity reversible data hiding scheme based on pixel-value-ordering and prediction-error expansion.** *Signal Processing*, **93**(1):198–205, January 2013.
- [102] B. OU, X. LI, Y. ZHAO, AND R. NI. **Reversible data hiding using invariant pixel-value-ordering and prediction-error expansion.** *Signal Processing: Image Communication*, **29**(7):760 – 772, August 2014.

- [103] W. HE, K. ZHOU, J. CAI, L. WANG, AND G. XIONG. **Reversible data hiding using multi-pass pixel value ordering and prediction-error expansion.** *Journal of Visual Communication and Image Representation*, **49**:351 – 360, November 2017.
- [104] X. WANG, J. DING, AND Q. PE. **A novel reversible image data hiding scheme based on pixel value ordering and dynamic pixel block partition.** *Information Sciences*, **310**:16 – 35, July 2015.
- [105] J.-J. LI, C.-F. LEE, C.-C. CHANG, J.-Y. LIN, AND Y.-H. WU. **Reversible data hiding scheme based on quad-tree and pixel value ordering.** *IEEE Access*, **7**:142947–142962, September 2019.
- [106] X. QU AND H. J. KIM. **Pixel-based pixel value ordering predictor for high-fidelity reversible data hiding.** *Signal Processing*, **111**:249 – 260, June 2015.
- [107] H. XIANG AND H. LIU. **A pixel-based reversible data hiding method based on obtuse angle prediction.** In *Proceedings of 2nd International Conference on Multimedia and Image Processing*, pages 191–195, Wuhan, China, March 2017.
- [108] P. RAHMANI AND G. DASTGHAIBYFARD. **A reversible data hiding scheme based on prediction-error expansion using pixel-based pixel value ordering predictor.** In *Proceedings of Artificial Intelligence and Signal Processing Conference*, pages 219–223, Shiraz, Iran, October 2017.
- [109] S. WENG, G. ZHANG, J.-S. PAN, AND Z. ZHOU. **Optimal PPVO-based reversible data hiding.** *Journal of Visual Communication and Image Representation*, **48**:317 – 328, October 2017.
- [110] W. HE, G. XIONG, K. ZHOU, AND J. CAI. **Reversible data hiding based on multilevel histogram modification and pixel value grouping.** *Journal of Visual Communication and Image Representation*, **40**:459 – 469, October 2016.

- [111] I. C. DRAGOI, I. CACIULA, AND D. COLTUC. **Improved pairwise pixel-value-ordering for high-fidelity reversible data hiding.** In *Proceedings of 25th IEEE International Conference on Image Processing*, pages 1668–1672, Athens, Greece, October 2018.
- [112] W. HE, J. CAI, G. XIONG, AND K. ZHOU. **Improved reversible data hiding using pixel-based pixel value grouping.** *Optik*, **157**:68–78, March 2018.
- [113] F. PENG, X. LI, AND B. YANG. **Improved PVO-based reversible data hiding.** *Digital Signal Processing*, **25**:255 – 265, February 2014.
- [114] B. OU, X. LI, AND J. WANG. **High-fidelity reversible data hiding based on pixel-value-ordering and pairwise prediction-error expansion.** *Journal of Visual Communication and Image Representation*, **39**:12 – 23, August 2016.
- [115] B. OU, X. LI, AND J. WANG. **Improved PVO-based reversible data hiding: a new implementation based on multiple histograms modification.** *Journal of Visual Communication and Image Representation*, **38**:328–339, July 2016.
- [116] S. WENG, J.-S. PAN, AND L. LI. **Reversible data hiding based on an adaptive pixel-embedding strategy and two-layer embedding.** *Information Sciences*, **369**:144 – 159, November 2016.
- [117] W. HE, J. CAI, K. ZHOU, AND G. XIONG. **Efficient PVO-based reversible data hiding using multistage blocking and prediction accuracy matrix.** *Journal of Visual Communication and Image Representation*, **46**:58 – 69, July 2017.
- [118] W. HONG AND T. S. CHEN. **Reversible data embedding for high quality images using interpolation and reference pixel distribution mechanism.** *Journal of Visual Communication and Image Representation*, **22**(2):131 – 140, February 2011.

- [119] W. HONG, T.-S. CHEN, AND J. CHEN. **Reversible data hiding using delaunay triangulation and selective embedment.** *Information Sciences*, **308**:140 – 154, July 2015.
- [120] L. LUO, Z. CHEN, M. CHEN, X. ZENG, AND Z. XIONG. **Reversible image watermarking using interpolation technique.** *IEEE Transactions on Information Forensics and Security*, **5**(1):187–193, March 2010.
- [121] C. YUZZKOLLAR AND U. KOCABICAK. **Region based interpolation error expansion algorithm for reversible image watermarking.** *Applied Soft Computing*, **33**:127 – 135, August 2015.
- [122] G. FENG, Z. QIAN, AND N. DAI. **Reversible watermarking via extreme learning machine prediction.** *Neurocomputing*, **82**:62 – 68, April 2012.
- [123] T. LUO, G. JIANG, M. YU, C. ZHONG, H. XU, AND Z. PAN. **Convolutional neural networks-based stereo image reversible data hiding method.** *Journal of Visual Communication and Image Representation*, **61**:61 – 73, May 2019.
- [124] G. GAO, S. YAO, AND Z. CHONG. **Improved method of prediction error-based reversible watermarking for copyright protection.** In *Proceedings of International Conference on Internet Multimedia Computing and Service*, pages 395–399, Xiamen, China, July 2014.
- [125] Q. CHANG, G. CHEUNG, Y. ZHAO, X. LI, AND R. NI. **Non-local graph-based prediction for reversible data hiding in images.** In *Proceedings of 25th IEEE International Conference on Image Processing*, pages 1693–1697, Athens, Greece, October 2018.
- [126] X. HU, W. ZHANG, X. LI, AND N. YU. **Minimum rate prediction and optimized histograms modification for reversible data hiding.** *IEEE Transactions on Information Forensics and Security*, **10**(3):653–664, March 2015.

- [127] X. MA, Z. PAN, S. HU, AND L. WANG. **High-fidelity reversible data hiding scheme based on multi-predictor sorting and selecting mechanism.** *Journal of Visual Communication and Image Representation*, **28**:71 – 82, April 2015.
- [128] T.-C. LU, C.-M. CHEN, M.-C. LIN, AND Y.-H. HUANG. **Multiple predictors hiding scheme using asymmetric histograms.** *Multimedia Tools and Applications*, **76**:3361–3382, February 2017.
- [129] H.-Z. WU, H.-X. WANG, AND Y.-Q. SHI. **PPE-based reversible data hiding.** In *Proceedings of 4th ACM Workshop on Information Hiding and Multimedia Security*, pages 187–188, Vigo Galicia, Spain, June 2016.
- [130] D. COLTUC. **Low distortion transform for reversible watermarking.** *IEEE Transactions on Image Processing*, **21**(1):412–417, January 2012.
- [131] D. COLTUC AND I.-C. DRAGOI. **Context embedding for raster-scan rhombus based reversible watermarking.** In *Proceedings of the First ACM Workshop on Information Hiding and Multimedia Security*, pages 215–220, Montpellier, France, June 2013.
- [132] X. LI, B. YANG, AND T. ZENG. **Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection.** *IEEE Transactions on Image Processing*, **20**(12):3524–3533, December 2011.
- [133] H. Y. LEUNG, L. M. CHENG, F. LIU, AND Q. K. FU. **Adaptive reversible data hiding based on block median preservation and modification of prediction errors.** *Journal of Systems and Software*, **86**(8):2204 – 2219, August 2013.
- [134] M. CHEN, Z. CHEN, X. ZENG, AND Z. XIONG. **Reversible data hiding using additive prediction-error expansion.** In *Proceedings of the 11th ACM Workshop on Multimedia and Security*, pages 19–24, Princeton, USA, September 2009.

- [135] H.-T. WU AND J. HUANG. **Reversible image watermarking on prediction errors by efficient histogram modification.** *Signal Processing*, **92**(12):3000 – 3009, December 2012.
- [136] B. OU, X. LI, Y. ZHAO, R. NI, AND Y.-Q. SHI. **Pairwise prediction-error expansion for efficient reversible data hiding.** *IEEE Transactions on Image Processing*, **22**(12):5010–5021, December 2013.
- [137] B. OU, X. LI, J. WANG, AND F. PENG. **High-fidelity reversible data hiding based on geodesic path and pairwise prediction-error expansion.** *Neurocomputing*, **226**:23 – 34, February 2017.
- [138] M. XIAO, X. LI, Y. WANG, Y. ZHAO, AND R. NI. **Reversible data hiding based on pairwise embedding and optimal expansion path.** *Signal Processing*, **158**:210 – 218, May 2019.
- [139] S.-Y. WANG, C.-Y. LI, AND W.-C. KUO. **Reversible data hiding based on two-dimensional prediction errors.** *IET Image Processing*, **7**(9):805–816, December 2013.
- [140] 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*, **10**(9):2016–2027, September 2015.
- [141] T. ZHANG, X. LI, W. QI, W. LI, AND Z. GUO. **Real-time reversible data hiding based on multiple histogram modification.** *Journal of Real-Time Image Processing*, **16**(3):661–671, June 2019.
- [142] J. WANG, N. MAO, X. CHEN, J. NI, C. WANG, AND Y. SHI. **Multiple histograms based reversible data hiding by using FCM clustering.** *Signal Processing*, **159**:193 – 203, June 2019.
- [143] P. TSAI, Y.-C. HU, AND H.-L. YEH. **Reversible image hiding scheme using predictive coding and histogram shifting.** *Signal Processing*, **89**(6):1129 – 1143, June 2009.

- [144] W. HONG AND T.-S. CHEN. **A local variance-controlled reversible data hiding method using prediction and histogram-shifting.** *Journal of Systems and Software*, **83**(12):2653 – 2663, December 2010.
- [145] W. HONG. **Adaptive reversible data hiding method based on error energy control and histogram shifting.** *Optics Communications*, **285**(2):101 – 108, January 2012.
- [146] X. CHEN, X. SUN, H. SUN, Z. ZHOU, AND J. ZHANG. **Reversible watermarking method based on asymmetric-histogram shifting of prediction errors.** *Journal of Systems and Software*, **86**(10):2620 – 2626, October 2013.
- [147] X. CHEN, X. SUN, H. SUN, L. XIANG, AND B. YANG. **Histogram shifting based reversible data hiding method using directed-prediction scheme.** *Multimedia Tools and Applications*, **74**:5747–5765, August 2015.
- [148] T.-C. LU, C.-Y. TSENG, AND J.-H. WU. **Asymmetric-histogram based reversible information hiding scheme using edge sensitivity detection.** *Journal of Systems and Software*, **116**:2 – 21, June 2016.
- [149] G. XUAN, X. TONG, J. TENG, X. ZHANG, AND Y. Q. SHI. **Optimal histogram-pair and prediction-error based image reversible data hiding.** In *Proceedings of International Workshop on Digital Watermarking*, pages 363–383, Sanghai, China, October-November 2012.
- [150] I. CACIULA AND D. COLTUC. **Improved control for low bit-rate reversible watermarking.** In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 7425–7429, May 2014.
- [151] G. XUAN, X. LI, AND Y.-Q. SHI. **Histogram-pair based reversible data hiding via searching for optimal four thresholds.** *Journal of Information Security and Applications*, **39**:58 – 67, April 2018.
- [152] T.-S. NGUYEN AND C.-C. CHANG. **A reversible data hiding scheme based on the sudoku technique.** *Displays*, **39**:109 – 116, October 2015.

- [153] B. MA AND Y. Q. SHI. **A reversible data hiding scheme based on code division multiplexing.** *IEEE Transactions on Information Forensics and Security*, **11**(9):1914–1927, September 2016.
- [154] D. R. D. BRABIN, J. R. P. PERINBAM, AND D. MEGANATHAN. **A block based reversible data hiding scheme for digital images using optimal value computation.** *Wireless Personal Communications*, **94**(4):2583–2596, June 2017.
- [155] C.-F. LEE, K.-H. WANG, C.-C. CHANG, AND Y.-L. HUANG. **A reversible data hiding scheme based on dual steganographic images.** In *Proceedings of the 3rd International Conference on Ubiquitous Information Management and Communication*, pages 228–237, Suwon, Korea, February 2009.
- [156] C.-F LEE AND Y.-L. HUANG. **Reversible data hiding scheme based on dual stegano-images using orientation combinations.** *Telecommunication Systems*, **52**:2237–2247, April 2013.
- [157] T.-C. LU, C.-Y. TSENG, AND J.-H. WU. **Dual imaging-based reversible hiding technique using LSB matching.** *Signal Processing*, **108**:77 – 89, March 2015.
- [158] H. YAO, C. QIN, Z. TANG, AND Y. TIAN. **Improved dual-image reversible data hiding method using the selection strategy of shiftable pixels’ coordinates with minimum distortion.** *Signal Processing*, **135**:26 – 35, June 2017.
- [159] N. R. SWAMY, S. ISLAM, AND P. GUPTA. **Two level reversible data hiding in 2D images.** In *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, pages 58–59, Gyeongju, Korea, March 2014.
- [160] W. HE. **Improved block redundancy mining based reversible data hiding using multi-sub-blocking.** *Signal Processing: Image Communication*, **60**:199 – 210, February 2018.

- [161] R. T. MOHAMMED AND B. E. KHOO. **Robust reversible watermarking scheme based on wavelet-like transform.** In *Proceedings of IEEE International Conference on Signal and Image Processing Applications*, pages 354–359, Melaka, Malaysia, October 2013.
- [162] R. THABIT AND B. E. KHOO. **Robust reversible watermarking scheme using Slantlet transform matrix.** *Journal of Systems and Software*, **88**:74 – 86, February 2014.
- [163] I. C. DRAGOI, D. COLTUC, AND H. G. COANDA. **Adaptive pairwise reversible watermarking with horizontal grouping.** In *Proceedings of International Symposium on Signals, Circuits and Systems*, pages 1–4, Iasi, Romania, July 2017.
- [164] RESERVE BANK OF INDIA. **Frequently asked questions (FAQ) on cheque truncation project in the national capital region.** <https://rbidocs.rbi.org.in/rdocs/content/pdfs/74751.pdf>.
- [165] R. PAL AND V. NAVEENCHANDRA. **A framework of digital cheque: a prospect in payment system in India.** *The Indian Banker*, **I**(5):28–32, December 2013.
- [166] P. STROBACH. **Quadtree-structured recursive plane decomposition coding of images.** *IEEE Transactions on Signal Processing*, **39**(6):1380–1397, June 1991.
- [167] X. YUAN AND Z. CAI. **An adaptive triangular partition algorithm for digital images.** *IEEE Transactions on Multimedia*, **21**(6):1372–1383, June 2019.
- [168] K.-S. KIM, M.-J. LEE, Y.-H. SUH, AND H.-K. LEE. **Robust lossless data hiding based on block gravity center for selective authentication.** In *Proceedings of IEEE International Conference on Multimedia and Expo*, pages 1022–1025, New York, USA, June 2009.

- [169] J. LI, X. LI, AND B. YANG. **Reversible data hiding scheme for color image based on prediction-error expansion and cross-channel correlation.** *Signal Processing*, **93**(9):2748 – 2758, September 2013.
- [170] B. OU, X. LI, Y. ZHAO, AND R. NI. **Efficient color image reversible data hiding based on channel-dependent payload partition and adaptive embedding.** *Signal Processing*, **108**:642 – 657, March 2015.
- [171] D. HOU, W. ZHANG, K. CHEN, S.-J. LIN, AND N. YU. **Reversible data hiding in color image with grayscale invariance.** *IEEE Transactions on Circuits and Systems for Video Technology*, **29**(2):363–374, February 2019.
- [172] S. YI AND Y. ZHOU. **Binary-block embedding for reversible data hiding in encrypted images.** *Signal Processing*, **133**:40 – 51, April 2017.
- [173] H.-Z. WU, Y.-Q. SHI, H.-X. WANG, AND L.-N. ZHOU. **Separable reversible data hiding for encrypted palette images with color partitioning and flipping verification.** *IEEE Transactions on Circuits and Systems for Video Technology*, **27**(8):1620–1631, August 2017.
- [174] 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*, **13**(7):1670–1681, July 2018.
- [175] B. GUAN AND D. XU. **An efficient high-capacity reversible data hiding scheme for encrypted images.** *Journal of Visual Communication and Image Representation*, **66**, January 2020.
- [176] D. HUANG AND J. WANG. **High-capacity reversible data hiding in encrypted image based on specific encryption process.** *Signal Processing: Image Communication*, **80**, February 2020.
- [177] K.-L. CHUNG, C.-Y. CHIU, T.-Y. YU, AND P.-L. HUANG. **Temporal and spatial correlation-based reversible data hiding for RGB CFA videos.** *Information Sciences*, **420**:386 – 402, December 2017.

Reversible Data Hiding in Grayscale Images Using Prediction Error Expansion Based Techniques

by Ravi Uyyala

Submission date: 24-Feb-2020 02:44PM (UTC+0530)

Submission ID: 1263018683

File name: ale_Images_Using_Prediction_Error_Expansion_Based_Techniques.pdf (12.88M)

Word count: 51387

Character count: 249252

Reversible Data Hiding in Grayscale Images Using Prediction Error Expansion Based Techniques

ORIGINALITY REPORT

28%

SIMILARITY INDEX

3%

INTERNET SOURCES

26%

PUBLICATIONS

10%

STUDENT PAPERS

PRIMARY SOURCES

1

Ravi Uyyala, Rajarshi Pal, Munaga V.N.K. Prasad. "Reversible data hiding using B-tree triangular decomposition based prediction", IET Image Processing, 2019

Publication

15%

2

Ravi Uyyala, Rajarshi Pal. "Reversible Data Hiding Based on the Random Distribution of Reference Pixels", 2018 IEEE Region Ten Symposium (Tensymp), 2018

Publication

3%

3

Submitted to University of Hyderabad, Hyderabad

Student Paper

2%

4

"Proceedings of International Conference on Computer Vision and Image Processing", Springer Science and Business Media LLC, 2017

Publication

1%

5

Ravi Uyyala, Rajarshi Pal, Munaga V.N.K

<1%

Prasad. "Gradient dependent reversible watermarking with low embedding impact", 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN), 2016

Publication

6

link.springer.com

Internet Source

<1 %

7

Lecture Notes in Computer Science, 2015.

Publication

<1 %

8

Submitted to Higher Education Commission Pakistan

Student Paper

<1 %

9

Kota, Saranya, and Rajarshi Pal. "Detecting tampered cheque images in cheque truncation system using difference expansion based watermarking", 2014 IEEE International Advance Computing Conference (IACC), 2014.

Publication

<1 %

10

Submitted to SASTRA University

Student Paper

<1 %

11

Wenfa Qi, Xiaolong Li, Tong Zhang, Zongming Guo. "Optimal reversible data hiding scheme based on multiple histograms modification", IEEE Transactions on Circuits and Systems for Video Technology, 2019

Publication

<1 %

12	Lecture Notes in Computer Science, 2016. Publication	<1 %
13	www.ile.rochester.edu Internet Source	<1 %
14	eisc.univalle.edu.co Internet Source	<1 %
15	"Digital Forensics and Watermarking", Springer Science and Business Media LLC, 2017 Publication	<1 %
16	"Cloud Computing and Security", Springer Science and Business Media LLC, 2018 Publication	<1 %
17	epdf.tips Internet Source	<1 %
18	Huang, Hsiang-Cheh, and Feng-Cheng Chang. "Hierarchy-based reversible data hiding", Expert Systems with Applications, 2013. Publication	<1 %
19	qtc.jp Internet Source	<1 %
20	Submitted to University of Houston, Downtown Student Paper	<1 %
21	ijns.femto.com.tw Internet Source	<1 %

22	Submitted to Feng Chia University Student Paper	<1 %
23	"Digital-Forensics and Watermarking", Springer Science and Business Media LLC, 2014 Publication	<1 %
24	ses.library.usyd.edu.au Internet Source	<1 %
25	"Digital Watermarking", Springer Science and Business Media LLC, 2005 Publication	<1 %
26	"Advances in Intelligent Information Hiding and Multimedia Signal Processing", Springer Nature, 2017 Publication	<1 %
27	Submitted to Jordan University of Science & Technology Student Paper	<1 %
28	Submitted to Indian Institute of Technology, Bombay Student Paper	<1 %
29	"Digital Forensics and Watermaking", Springer Science and Business Media LLC, 2013 Publication	<1 %
30	doi.org Internet Source	<1 %

31	Manoj Kumar, Smita Agrawal. "Reversible data hiding based on prediction error expansion using adjacent pixels", Security and Communication Networks, 2016 Publication	<1 %
32	H.Y. Leung, L.M. Cheng, F. Liu, Q.K. Fu. "Adaptive reversible data hiding based on block median preservation and modification of prediction errors", Journal of Systems and Software, 2013 Publication	<1 %
33	addi.ehu.es Internet Source	<1 %
34	repository.lib.cuhk.edu.hk Internet Source	<1 %
35	D. R. Denslin Brabin, J. Raja Paul Perinbam, D. Meganathan. "A Block Based Reversible Data Hiding Scheme for Digital Images Using Optimal Value Computation", Wireless Personal Communications, 2016 Publication	<1 %
36	Submitted to Indian Institute of Management Calcutta Student Paper	<1 %
37	www.irjet.net Internet Source	<1 %

38

Rajeev Kumar, Satish Chand. "A reversible high capacity data hiding scheme using pixel value adjusting feature", Multimedia Tools and Applications, 2014

Publication

<1 %

39

Hong, W.. "A local variance-controlled reversible data hiding method using prediction and histogram-shifting", The Journal of Systems & Software, 201012

Publication

<1 %

40

Submitted to Chulalongkorn University

Student Paper

<1 %

41

Ya-Ting Chang, Cheng-Ta Huang, Chin-Feng Lee, Shih-Jeng Wang. "Image interpolating based data hiding in conjunction with pixel-shifting of histogram", The Journal of Supercomputing, 2013

Publication

<1 %

42

Muhammad Ishtiaq, Waqar Ali, Waseem Shahzad, Muhammad Arfan Jaffar, Yunyoung Nam. "Hybrid Predictor Based Four-Phase Adaptive Reversible Watermarking", IEEE Access, 2018

Publication

<1 %

43

Fu-Hau Hsu, Min-Hao Wu, Shih-Jeng Wang, Chia-Ling Huang. "Reversibility of image with balanced fidelity and capacity upon pixels

<1 %

differentencing expansion", The Journal of Supercomputing, 2013

Publication

44

Submitted to Jawaharlal Nehru Technological University Anantapur

Student Paper

<1 %

45

Aruna Malik, Samayveer Singh, Rajeev Kumar. "Recovery based high capacity reversible data hiding scheme using even-odd embedding", Multimedia Tools and Applications, 2017

Publication

<1 %

46

Submitted to Universiti Sains Malaysia

Student Paper

<1 %

47

Diqun Yan, Rangding Wang. "Reversible Data Hiding for Audio Based on Prediction Error Expansion", 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2008

Publication

<1 %

48

Smart Innovation Systems and Technologies, 2016.

Publication

<1 %

49

Submitted to Anna University, Chennai

Student Paper

<1 %

50

www.eurasip.org

Internet Source

<1 %

51	Yu Jing. "Study on reversible data hiding scheme for digital images", 2010 2nd International Asia Conference on Informatics in Control Automation and Robotics (CAR 2010), 03/2010 Publication	<1 %
52	Ou, Bo, Xiaolong Li, Yao Zhao, and Rongrong Ni. "Reversible data hiding based on PDE predictor", Journal of Systems and Software, 2013. Publication	<1 %
53	Hong, W.. "A high capacity reversible data hiding scheme using orthogonal projection and prediction error modification", Signal Processing, 201011 Publication	<1 %
54	www.sce.carleton.ca Internet Source	<1 %
55	Submitted to University of Bedfordshire Student Paper	<1 %
56	Submitted to Sardar Vallabhbhai National Inst. of Tech.Surat Student Paper	<1 %
57	Li, X., B. Li, B. YANG, and T. ZENG. "A general framework to histogram-shifting-based reversible data hiding", IEEE Transactions on	<1 %

58

Submitted to GLA University

Student Paper

<1 %

59

Wei Fan, Zhenyong Chen, Ming Chen, Lixin Luo, Zhang Xiong. "Reversible data hiding with context modeling, generalized expansion and boundary map", Multimedia Tools and Applications, 2010

Publication

<1 %

60

Asifullah Khan, Ayesha Siddiqa, Summuyya Munib, Sana Ambreen Malik. "A recent survey of reversible watermarking techniques", Information Sciences, 2014

Publication

<1 %

61

"Advances in Multimedia Information Processing - PCM 2009", Springer Science and Business Media LLC, 2009

Publication

<1 %

62

amp.ece.cmu.edu

Internet Source

<1 %

63

"Proceedings of the International Conference on Signal, Networks, Computing, and Systems", Springer Science and Business Media LLC, 2017

Publication

<1 %

64

Wang, J.X.. "A path optional lossless data hiding scheme based on VQ joint neighboring coding", Information Sciences, 20090909

Publication

<1 %

65

www.cs.tut.fi

Internet Source

<1 %

66

"Recent Advances in Information Hiding and Applications", Springer Science and Business Media LLC, 2013

Publication

<1 %

67

Rajeev Kumar, Satish Chand. "A novel high capacity reversible data hiding scheme based on pixel intensity segmentation", Multimedia Tools and Applications, 2015

Publication

<1 %

68

Submitted to iGroup

Student Paper

<1 %

69

Submitted to University of Northumbria at Newcastle

Student Paper

<1 %

70

Wenguang He, Zhanchuan Cai, Yaomin Wang. "Flexible spatial location-based PVO predictor for high-fidelity reversible data hiding", Information Sciences, 2020

Publication

<1 %

Submitted to Karpagam Academy of Higher

71

Education

Student Paper

<1 %

72

Peyman Rahmani, Gholamhossein Dastghaibiyfard. "A reversible data hiding scheme based on prediction-error expansion using pixel-based pixel value ordering predictor", 2017 Artificial Intelligence and Signal Processing Conference (AISP), 2017

Publication

<1 %

73

Jagadish Nayak. "Efficient Storage and Transmission of Digital Fundus Images with Patient Information Using Reversible Watermarking Technique and Error Control Codes", Journal of Medical Systems, 06/2009

Publication

<1 %

74

Submitted to Jawaharlal Nehru Technological University

Student Paper

<1 %

75

thanglong.ece.jhu.edu

Internet Source

<1 %

76

Submitted to Indian Institute of Technology, Madras

Student Paper

<1 %

77

Mohsin Shah, Weiming Zhang, Honggang Hu, Xiaojuan Dong, Nenghai Yu. "Prediction error expansion-based reversible data hiding in

<1 %

encrypted images with public key cryptosystem", IET Image Processing, 2019

Publication

78

mindwave.biz

Internet Source

<1 %

79

www.ict-bone.eu

Internet Source

<1 %

80

esrdc.mit.edu

Internet Source

<1 %

81

Submitted to King Saud University

Student Paper

<1 %

82

Tsai, P.. "Reversible image hiding scheme using predictive coding and histogram shifting", Signal Processing, 200906

Publication

<1 %

83

Submitted to Universiti Malaysia Pahang

Student Paper

<1 %

84

Submitted to Cummins College of Engineering for Women, Pune

Student Paper

<1 %

85

"Digital Forensics and Watermarking", Springer Science and Business Media LLC, 2019

Publication

<1 %

86

"Advances in Intelligent Information Hiding and Multimedia Signal Processing", Springer

<1 %

87

Zhe-Ming Lu, Shi-Ze Guo. "Lossless Information Hiding in Images on Transform Domains", Elsevier BV, 2017

Publication

<1 %

88

www.ijert.org

Internet Source

<1 %

89

Chuan Qin, Chin-Chen Chang, Ying-Hsuan Huang, Li-Ting Liao. "An Inpainting-Assisted Reversible Steganographic Scheme Using a Histogram Shifting Mechanism", IEEE Transactions on Circuits and Systems for Video Technology, 2013

Publication

<1 %

90

qspace.library.queensu.ca

Internet Source

<1 %

91

Fei Peng, Xiaolong Li, Bin Yang. "An adaptive PEE-based reversible data hiding scheme exploiting referential prediction-errors", 2015 IEEE International Conference on Multimedia and Expo (ICME), 2015

Publication

<1 %

92

Kehao Wang, Guohua Chen, Qingsong Ai, Hui Cao, Pan Zhou, Dapeng Wu. "Reversible Data Hiding Based on Structural Similarity Block Selection", IEEE Access, 2020

<1 %

93

Submitted to Queensland University of Technology

Student Paper

<1 %

94

"Intelligent Information and Database Systems", Springer Science and Business Media LLC, 2012

Publication

<1 %

95

www.igi-global.com

Internet Source

<1 %

96

lib.ugent.be

Internet Source

<1 %

97

Bin Ma, Jian Xu, Yun Q. Shi. "A Novel CDMA Based High Performance Reversible Data Hiding Scheme", Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security - IH&MMSec '16, 2016

Publication

<1 %

98

Submitted to The Hong Kong Polytechnic University

Student Paper

<1 %

99

Kumar, Rajeev, and Satish Chand. "A reversible data hiding scheme using bit flipping strategy", Journal of Discrete Mathematical Sciences and Cryptography, 2016.

Publication

<1 %

100	Brian K. Lien, Yen-ming Lin. "High-capacity reversible data hiding by maximum-span pairing", Multimedia Tools and Applications, 2010 Publication	<1 %
101	Submitted to Mizoram University Student Paper	<1 %
102	www.google.im Internet Source	<1 %
103	Vasiliy Sachnev, Hyoung Joong Kim, Sundaram Suresh, Yun Qing Shi. "Reversible Watermarking Algorithm with Distortion Compensation", EURASIP Journal on Advances in Signal Processing, 2011 Publication	<1 %
104	Zhe-Ming Lu, Shi-Ze Guo. "Lossless Information Hiding in Block Truncation Coding–Compressed Images", Elsevier BV, 2017 Publication	<1 %
105	Al-Qershi, Osamah M., and Bee Ee Khoo. "Two-dimensional difference expansion (2D-DE) scheme with a characteristics-based threshold", Signal Processing, 2013. Publication	<1 %
106	Rad, Reza Moradi, KokSheik Wong, and Jing-Ming Guo. "Reversible data hiding by adaptive	<1 %

group modification on histogram of prediction errors", Signal Processing, 2016.

Publication

-
- 107 Santosh Arjun, Narasimha Rao. "An approach to reversible information hiding for images", TENCON 2008 - 2008 IEEE Region 10 Conference, 2008 <1 %

Publication

-
- 108 graphics.tu-bs.de <1 %

Internet Source

-
- 109 Lin, Y.C.. "Reversible data-hiding for progressive image transmission", Signal Processing: Image Communication, 201111 <1 %

Publication

-
- 110 www.dtic.mil <1 %

Internet Source

-
- 111 Yun-Qing Shi, Xiaolong Li, Xinpeng Zhang, Hao-Tian Wu, Bin Ma. "Reversible data hiding: Advances in the past two decades", IEEE Access, 2016 <1 %

Publication

Exclude quotes

On

Exclude matches

< 14 words

Exclude bibliography

Off

Reversible Data Hiding in Grayscale Images Using Prediction Error Expansion Based Techniques

ORIGINALITY REPORT

40%

SIMILARITY INDEX

3%

INTERNET SOURCES

39%

PUBLICATIONS

10%

STUDENT PAPERS

PRIMARY SOURCES

1

Ravi Uyyala, Rajarshi Pal, Munaga V.N.K. Prasad. "Reversible data hiding using B-tree triangular decomposition based prediction", IET Image Processing, 2019

Publication

27%

2

Ravi Uyyala, Rajarshi Pal. "Reversible Data Hiding Based on the Random Distribution of Reference Pixels", 2018 IEEE Region Ten Symposium (Tensymp), 2018

Publication

5%

3

Submitted to University of Hyderabad, Hyderabad

Student Paper

2%

4

Ravi Uyyala, Rajarshi Pal, Munaga V.N.K Prasad. "Gradient dependent reversible watermarking with low embedding impact", 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN), 2016

Publication

<1%

5	link.springer.com Internet Source	<1 %
6	"Proceedings of International Conference on Computer Vision and Image Processing", Springer Science and Business Media LLC, 2017 Publication	<1 %
7	Lecture Notes in Computer Science, 2015. Publication	<1 %
8	"Advances in Intelligent Information Hiding and Multimedia Signal Processing", Springer Nature, 2017 Publication	<1 %
9	Lecture Notes in Computer Science, 2014. Publication	<1 %
10	Submitted to Feng Chia University Student Paper	<1 %
11	D. R. Denslin Brabin, J. Raja Paul Perinbam, D. Meganathan. "A Block Based Reversible Data Hiding Scheme for Digital Images Using Optimal Value Computation", Wireless Personal Communications, 2016 Publication	<1 %
12	Submitted to Jordan University of Science & Technology Student Paper	<1 %

13

Manoj Kumar, Smita Agrawal. "Reversible data hiding based on prediction error expansion using adjacent pixels", Security and Communication Networks, 2016

Publication

<1 %

14

"Digital Forensics and Watermarking", Springer Nature, 2017

Publication

<1 %

15

Muhammad Ishtiaq, Waqar Ali, Waseem Shahzad, Muhammad Arfan Jaffar, Yunyoung Nam. "Hybrid Predictor Based Four-Phase Adaptive Reversible Watermarking", IEEE Access, 2018

Publication

<1 %

16

Fu-Hau Hsu, Min-Hao Wu, Shih-Jeng Wang, Chia-Ling Huang. "Reversibility of image with balanced fidelity and capacity upon pixels differencing expansion", The Journal of Supercomputing, 2013

Publication

<1 %

17

Submitted to Jawaharlal Nehru Technological University Anantapur

Student Paper

<1 %

18

Submitted to Universiti Sains Malaysia

Student Paper

<1 %

19

Diqun Yan, Rangding Wang. "Reversible Data

<1 %

Hiding for Audio Based on Prediction Error Expansion", 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2008

Publication

20

repository.ubn.ru.nl

Internet Source

<1 %

21

amp.ece.cmu.edu

Internet Source

<1 %

22

Hong, W.. "A local variance-controlled reversible data hiding method using prediction and histogram-shifting", The Journal of Systems & Software, 201012

Publication

<1 %

23

Wei Fan, Zhenyong Chen, Ming Chen, Lixin Luo, Zhang Xiong. "Reversible data hiding with context modeling, generalized expansion and boundary map", Multimedia Tools and Applications, 2010

Publication

<1 %

24

"Digital-Forensics and Watermarking", Springer Science and Business Media LLC, 2016

Publication

<1 %

25

Asifullah Khan, Ayesha Siddiqa, Summuyya Munib, Sana Ambreen Malik. "A recent survey of reversible watermarking techniques", Information Sciences, 2014

<1 %

- 26 Aruna Malik, Samayveer Singh, Rajeev Kumar. "Recovery based high capacity reversible data hiding scheme using even-odd embedding", Multimedia Tools and Applications, 2017 $<1\%$

Publication

- 27 "Cloud Computing and Security", Springer Science and Business Media LLC, 2018 $<1\%$

Publication

- 28 Hong, W.. "A high capacity reversible data hiding scheme using orthogonal projection and prediction error modification", Signal Processing, 201011 $<1\%$

Publication

- 29 Rajeev Kumar, Satish Chand. "A novel high capacity reversible data hiding scheme based on pixel intensity segmentation", Multimedia Tools and Applications, 2015 $<1\%$

Publication

- 30 Submitted to iGroup $<1\%$

Student Paper

- 31 Wenguang He, Zhanchuan Cai, Yaomin Wang. "Flexible spatial location-based PVO predictor for high-fidelity reversible data hiding", Information Sciences, 2020 $<1\%$

Publication

32

www.eurasip.org

Internet Source

<1 %

33

Submitted to Karpagam Academy of Higher Education

Student Paper

<1 %

34

Submitted to SASTRA University

Student Paper

<1 %

35

Ou, Bo, Xiaolong Li, Yao Zhao, and Rongrong Ni. "Reversible data hiding based on PDE predictor", Journal of Systems and Software, 2013.

Publication

<1 %

36

Mohsin Shah, Weiming Zhang, Honggang Hu, Xiaojuan Dong, Nenghai Yu. "Prediction error expansion-based reversible data hiding in encrypted images with public key cryptosystem", IET Image Processing, 2019

Publication

<1 %

37

"Digital Forensics and Watermarking", Springer Science and Business Media LLC, 2019

Publication

<1 %

38

"Advances in Intelligent Information Hiding and Multimedia Signal Processing", Springer Science and Business Media LLC, 2018

Publication

<1 %

Zhe-Ming Lu, Shi-Ze Guo. "Lossless Information

- | | | |
|----|--|------|
| 39 | Hiding in Images on Transform Domains", Elsevier BV, 2017
Publication | <1 % |
| 40 | qspace.library.queensu.ca
Internet Source | <1 % |
| 41 | Fei Peng, Xiaolong Li, Bin Yang. "An adaptive PEE-based reversible data hiding scheme exploiting referential prediction-errors", 2015 IEEE International Conference on Multimedia and Expo (ICME), 2015
Publication | <1 % |
| 42 | www.irjet.net
Internet Source | <1 % |
| 43 | Kehao Wang, Guohua Chen, Qingsong Ai, Hui Cao, Pan Zhou, Dapeng Wu. "Reversible Data Hiding Based on Structural Similarity Block Selection", IEEE Access, 2020
Publication | <1 % |
| 44 | "Intelligent Information and Database Systems", Springer Science and Business Media LLC, 2012
Publication | <1 % |
| 45 | www.igi-global.com
Internet Source | <1 % |
| 46 | "Advances in Multimedia Information Processing - PCM 2009", Springer Science and Business | <1 % |

47

Bin Ma, Jian Xu, Yun Q. Shi. "A Novel CDMA Based High Performance Reversible Data Hiding Scheme", Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security - IH&MMSec '16, 2016

Publication

<1 %

48

Submitted to GLA University

Student Paper

<1 %

49

Brian K. Lien, Yen-ming Lin. "High-capacity reversible data hiding by maximum-span pairing", Multimedia Tools and Applications, 2010

Publication

<1 %

50

Submitted to Anna University, Chennai

Student Paper

<1 %

51

Al-Qershi, Osamah M., and Bee Ee Khoo. "Two-dimensional difference expansion (2D-DE) scheme with a characteristics-based threshold", Signal Processing, 2013.

Publication

<1 %

52

Rad, Reza Moradi, KokSheik Wong, and Jing-Ming Guo. "Reversible data hiding by adaptive group modification on histogram of prediction errors", Signal Processing, 2016.

Publication

<1 %

53

Santosh Arjun, Narasimha Rao. "An approach to reversible information hiding for images",
TENCON 2008 - 2008 IEEE Region 10
Conference, 2008

Publication

<1%

54

Submitted to Higher Education Commission
Pakistan

Student Paper

<1%

Exclude quotes On

Exclude matches < 14 words

Exclude bibliography On