Analysis of Reversible Data Hiding Techniques

Ms. Himani Patel, Ms. Himani Parekh

Abstract— Data hiding is the type of steganography in which secret data is embedded into the digital media such as texts, audio, images and videos. The categories of data hiding techniques are reversible and non-reversible. Reversible data hiding can be defined as an approach where the data is hidden in the host media and host image can be recovered without loss after secret data is extracted. This report describes different techniques of reversible data hiding such as difference expansion, interpolation technique, prediction and sorting, histogram modification.

*Index Terms— **Reversible Data Hiding (RDH), Difference Expansion (DE), Prediction Error Expansion (PEE), histogram shifting.***

I. INTRODUCTION

Data hiding is the art and science of communicating secret data in an appropriate multimedia carrier, e.g., image, audio, and video files. Digital Steganography and watermarking are two kind of data hiding. Data hiding can be done by two approaches reversible data hiding and irreversible data hiding. Irreversible data hiding can be defined as an approach in which original image cannot be recovered after extracting data. A reversible data hiding is an approach in which the original image can be recovered losslessly after the data have been extracted. Reversible data embedding, which is also called lossless data embedding, embeds invisible data into a digital image in a reversible fashion. As a basic requirement, the quality degradation on the image after data embedding should be low. An intriguing feature of reversible data embedding is the reversibility, that is, one can remove the embedded data to restore the original image. Data hiding can be used for copyright protection, media notation, integrity authentication, covert communication, etc. Most data hiding methods embed messages into the cover media like image or video to generate the marked media by only modifying the least significant part of the cover and, thus, ensure perceptual transparency. The embedding process will usually introduce permanent distortion to the cover, that is, the original cover can never be reconstructed from the marked cover. However, in some applications, such as medical imagery, military, and law forensics, no degradation of the original cover is allowed. We need a special kind of data hiding method, for such cases which is referred to as reversible data hiding (RDH) or lossless data hiding, by which the original cover can be lossless restored after the embedded message is extracted. The block diagram of reversible data hiding is shown in Figure 1. Reversible Steganography or watermarking can restore the original carrier without any distortion or with ignorable distortion after

the extraction of hidden data. So reversible data hiding is now getting popular. An information-hiding system is characterized using four different aspects: capacity, security, perceptibility and robustness [1] as shown in Fig. 1.

- **Capacity** refers to the amount of information that can be hidden in the cover medium.
- **Security** refers the inability of the hacker to extract hidden information.
- **Perceptibility** means the inability to detect the hidden information.
- **Robustness** is the amount of modification the stego-medium can withstand before an adversary can destroy the hidden information.

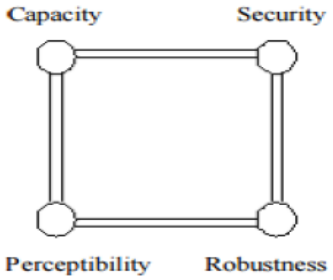

Figure 1: Characteristic of data hiding

The rest of these paper are organized as follows. Section III introduces the detailed RDH. In section IV, introduces the various techniques of RDH. In section V comparative study of various RDH techniques. And finally we conclude in section VI.

II. REVERSIBLE DATA HIDING

Reversible data hiding [11] is a technique which enables images to be authenticated and then restored to their original form by removing the digital watermark and replacing the image data that had been overwritten. This would make the images acceptable for legal purposes. Reversible data hiding (RDH) has the capability to erase the distortion introduced by embedding step after cover restoration. For this reason, RDH becomes a hot research topic and is extensively studied over the years. Reversible data hiding (RDH) in images is a technique, by which the

original cover can be losslessly recovered after the embedded message is extracted. The block diagram of RDH is shown in Figure.2.

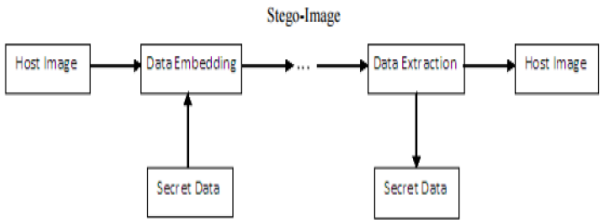

Figure 2: Reversible data hiding

Classification of data hiding technique

Data hiding techniques are classified into mainly two broad categories. Classification of each category is shown in figure 3.

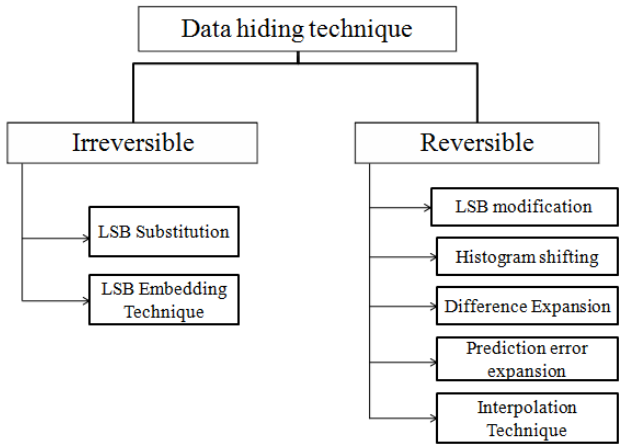

Figure 3:classification of data hiding

III. REVERSIBLE DATA HIDING TECHNIQUES

Various reversible data hiding techniques are described as follows:

A. LSB Modification Based Technique.

In this method, [4] a novel reversible data hiding technique enables the exact discovery of the original image after extracting the embedded information. LSB (Least Significant Bit) modification is proposed as the data embedding method, which introduces additional operating points on the capacity distortion(C-D) curve. It modifies the lowest levels instead of bit planes of the host image to accommodate the payload information. A prediction based conditional entropy coder utilizes static portions of the host as side-information. Information bits are embedded by modifying the selected features of the host image. It improves the compression efficiency and lossless data embedding capacity.

B. Histogram Shifting

Histogram based data hiding technique embeds the data in the cover media by shifting the histogram of the image. Histogram technique finds peak or zero points in the histogram and data embedding is done by shifting these peak and zero points. This technique yields higher data hiding capacity with low distortion. Histogram based reversible data hiding method was introduced by Ni et al. in [10], where message is embedded within the histogram. This technique prevents overflow and underflow problem. Overflow is the condition that the gray value exceeds above 255. Underflow is the condition that the
gray value falls below 0.

Embedding Process:

Consider an N pixel 8 bit gray scale image with pixel value yi (0-255).

Embedding process is done as follows:

1. Divide the image into two blocks.
2. Generate the histogram of each block.
3. Find the tree level, L of the binary tree.
4. For the first block, do the following steps
   a) Narrow the histogram in the range 2^L to $255-2^L$ by shifting the histogram from both sides.
   b) Scan the image block in the inverse S order and find difference between adjacent pixel values. Let di be the difference value.
   c) Then scan the image block in the same order and if difference value d_i is greater than 2^L , then shifting is done by 2^L units.
   Z_i={ y_i, if i=0 or $d_i < 2^L$,
   y_i+2^L, if $d_i > 2^L$ and $y_i \geq y_{i-1}$,
   $y_i - 2^L$, if and $y_i < y_{i-1}$}
   zi represents the pixels of the watermarked image.
   d) If $d < 2^L$, then message bits are embedded
   z_i = {$y_i + (d_i+b)$ if $y_i \geq y_{i-1}$
   $y_i (d_i + b)$ if $y_i < y_{i-1}$}
5. The above steps a)-d) is repeated for the second block.

Extraction process:

Consider a N pixel 8 bit watermarked image with pixel value z. Message bits can be extracted from the watermarked image blocks using the following steps:
For the first image block, do the following steps:

a) Scan the watermarked image block in the inverse S order
b) If $|z_i\ y_{i-1}| < 2^{(L+1)}$, extract message bit b by
   b={ 0, if $|z_i-y_{i-1}|$ is even
   1, if $|z_i-y_{i-1}|$ is odd}
   Where y_{i-1} denotes the restored value of z_{i-1}.

c) Original pixel value of host image block is restored by
Y_i= {zi + floor ($|z_i-y_{i-1}|$) /2 if $|z_i-y_{i-1}| < 2^{(L+1)}$ and $z_i < y_{i-1}$
z_i - floor ($|z_i-x_{i-1}|$) /2 if $|z_i-y_{i-1}| < 2^{(L+1)}$ and $z_i < y_{i-1}$
$z_i + 2^L$ if $|z_i-y_{i-1}| \geq 2^{(L+1)}$ and $z_i < y_{i-1}$
$z_i - 2^L$ if $|z_i-y_{i-1}| \geq 2^{(L-1)}$ and $z_i < y_{i-1}$
z_i otherwise}

d) This process is repeated until all the message bits are extracted from the two image blocks.

C. Difference Expansion

DE [5] technique discovers extra storage space by exploring the redundancy in the image content. In digital image, one can select the expandable difference values of pixels and embed one bit into each of them to extract the embedded data and to restore the original values, the decoder needs to know which difference values have been selected for the DE. Location map contains the location information of all selected expandable difference values. It also embedded into the marked image. This method suits all audio and videos with limited payload capacity.

The embedding and extracting algorithms are as follows:
Embedding phase in Difference Expansion (DE)

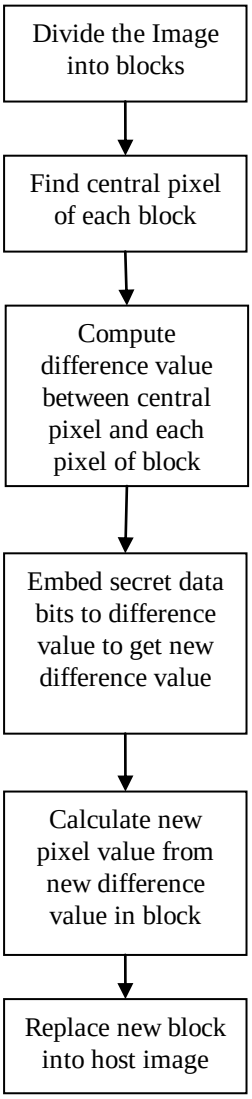

Figure 4: Embedding phase

1. Find the central-pixel b_c of each block by,
$$b_c = B(\lceil \frac{n}{2} \rceil, \lceil \frac{n}{2} \rceil)$$

2. Compute the difference value $di\ i \in \{1.., ((n \times n) - 1)\}$ between b_c and all pixels in the block by $d_i = |b_i - b_c|$

3. Embed the secret data bit s into each difference value and obtain new difference value $d_i' = (d_i \times 2) + s$

4. Calculate the new pixel values b_i for all of pixels in the block by $b_i' = \begin{cases} b_c - d_i', if b_i < b_c \\ b_c + d_i', if b_i \geq b_c \end{cases}$

5. Replace the new block into host image.

Extracting and recovery phase in show in DE

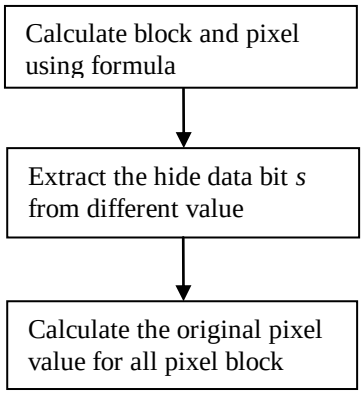

Figure 4:Extracting phase

1. Compute the difference value d_i $d_i', i \in \{1,.., ((n \times n) - 1)\}$ between b_c and all pixels in the block by $d_i' = |b_i' - b_c|$

2. Extract the secret data bit *s* from each difference value as $s = d_i' \bmod 2,$

3. Calculate the original pixel value for all pixels in the block by
$$b_i = \begin{cases} b_c - \left\lfloor \frac{d_i'}{2} \right\rfloor, if b_i' < b_c \\ b_c + \left\lfloor \frac{d_i'}{2} \right\rfloor, if b_i' \geq b_c \end{cases}$$

After this process, the original host image will be recovered completely and without any distortion.

C. Predication Error Expansion

In this method Combinations of histogram shifting and difference expansion two algorithms are introduced which helps to increase the capacity control and undesirable distortion at low embedding capacities. [7] The first one uses a highly compressible overflow map and the second one uses flag bits. Prediction Error based technique exploits the inherent correlation in the neighborhood of a pixel better than the Difference Expansion scheme. The maximal embedding capacity of a PE-base embedding technique in a single pass is 1 bpp, which is double the maximal capacity of 0.5 bpp for a DE based embedding technique. It offers a

significant improvement in the quality of the watermarked image, especially at moderate embedding capacities.

The embedding and extracting algorithms are as follows:

Embedding algorithm diagram show:

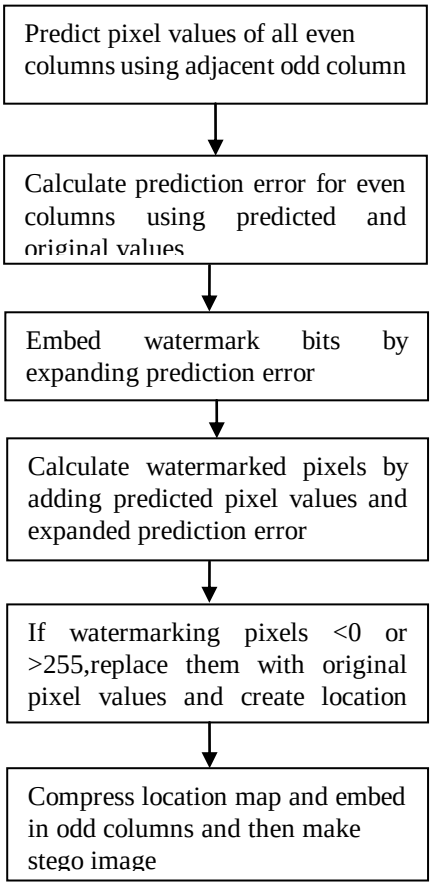

Figure 5: Embedding phase

1. Consider the input image *I* and predict the pixel values of even columns using adjacent odd columns pixel values.

$$\hat{x}_{2i,2j} = \frac{x_{(2i-1,2j-1)} + x_{(2i+1,2j+1)}}{2}$$

2. Obtain prediction error *p* as $p_{2i,2j} = x_{2i,2j} - \hat{x}_{2i,2j}$

where, $x_{2i,2j}$ is the original pixel value at the $(2i, 2j)^{th}$ position

3. Embed watermark bits by expanding even columns prediction errors. Let *wi* be the watermark bit at the *i*th position and $p'_{2i,2j}$ be the modified prediction error such as $p'_{2i,2j} = 2 \times p_{2i,2j} + w_i \quad where, w_i = 0 or 1$

4. Calculate watermarked pixel values by adding predicted pixel values and modified prediction error values as.

$$\hat{x}_{2i,2j} = \hat{x}_{2i,2j} + p'_{2i,2j}$$

5. If watermark pixel < 0 or >255 replace them with original pixel values and create location map for these pixels.

6. Compress location map *L* using arithmetic coding and embed it into odd columns and obtain final watermarked image I'.

Extraction algorithm show:

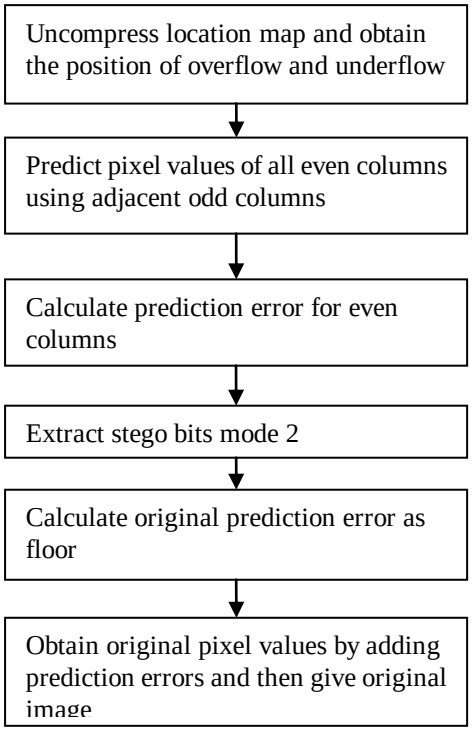

Figure 6: Extraction phase

1. Consider the watermarked image *I*0. Uncompress location map *L* from odd columns.

2. Obtain locations of overflow and underflow pixels.
3. Apply the following steps on even columns except pixel positions obtained from step2.

a. Calculate prediction error p'_{2i2j} for even columns.
b. Extract watermark bits $p'_{2i2j} \bmod 2$.
c. Calculate original prediction errors as floor $\left\lfloor \frac{p'_{2i,2j}}{2} \right\rfloor$
d. Obtain original pixel values by adding prediction errors.

D. Interpolation Technique

An interpolation technique [8] can embed a large amount of covert data into images with imperceptible modification. It utilizes the interpolation error, the difference between interpolation value and corresponding pixel value, to embed bit by expanding it additively or leaving it unchanged. The

data embedding approach of this proposed scheme, namely preservative interpolation error expansion, is a kind of DE. But it is different from most DE approaches in two aspects. One is it uses interpolation error instead of inter pixel difference or prediction error to embed data. Secondly, it expands difference by addition instead of bit shifting. The advantages of this method are that the distortion of preservative expansion is smaller since each pixel is altered at most by 1.

The embedding and extracting algorithms are as follows:

Embedding Algorithm:

I_{mxn}: original cover image
C_{mxn}: I_{mxn} with the interpolation; after embedding it becomes stego-image
S_{kxl}: secret image
B_{ij}: 5 by 5 non-overlapping sub-blocks of I_{mxn}
D_{ij}: 5 by 5 non-overlapping sub-blocks of C_{mxn}
M_{mxn}: edge map
E_bit: embedded bit
T: threshold

1. Convert S into a bit stream and use PRNG to shuffle it
2. Divide I into 5 by 5 non-overlapping sub-blocks, denoted Bij
3. Shrink each 5 by 5 non-overlapping sub-blocks of I into 3 by 3, non-overlapping sub-blocks, those kept left points are the control points, then use interpolation enlarge back to the size of I to get an image, denoted C and the sub blocks denoted Dij
4. Use edge detection method to find the edge map of C, called M.
5. for i=1 to m step 5 for j=1 to n step 5 if running out of secret data then stop if the corresponding sub-block of M and Dij has an edge point for p=1 to 5 step 2 for q=1 to 5 step 2 if Dij(p,q) + 2T > 255 or Dij(p,q) - 2T < 0 Do nothing;

else

Diff = | Dij(p,q) - Bij(p,q)|;

if Diff ≥ T if Dij(p,q) > Bij(p,q) Dij(p,q)= Bij(p,q)-T;

else if Dij(p,q) < Bij(p,q) Dij(p,q)= Bij(p,q)+T;

else

if Difference=0 Dij(p,q)=Dij(p,q)-E_ bit;

else if Dij(p,q) > Bij(p,q) Dij(p,q)=Dij(p,q)-2*Diff-E_bit;

else Dij(p,q)=Dij(p,q)+2*Diff+ E_bit;

6. Output Stego-image C

Extraction Algorithm

C_{mxn}: Stego-image
E_{mxn}: C_{mxn} with the interpolation
S_{kxl}: secret image
B_{ij}: 5 by 5 non-overlapping sub-blocks of C
D_{ij}: 5 by 5 non-overlapping sub-blocks of E
M_{mxn}: edge map
E_bit: embedded bit
T: threshold

1. Divide C into 5 by 5 non-overlapping sub-blocks, denoted B_{ij}
2. Shrink each 5 by 5 non-overlapping sub-blocks of C into 3 by 3, non-overlapping sub-blocks, those kept left points are the control points, then use interpolation enlarge back to the size of C to get an image, denoted E, and the sub-blocks denoted D_{ij}
3. Use edge detection method to find the edge map of E, called M.
4. for i=1 to m step 5

for j=1 to n step 5

if all secret data extracted then stop

if the corresponding sub-block of M and D_{ij} has an edge point

for p=1 to 5 step 2

for q=1 to 5 step 2

if $D_{ij}(p,q) + 2T > 255$ or $D_{ij}(p,q) - 2T < 0$

Do nothing;

else Diff=| Dij(p,q) - Bij(p,q)|;

if Diff : 2T if $D_{ij}(p,q)$ > Bij(p,q) $B_{ij}(p,q)$= Bij(p,q)+T;

else if Dij(p,q) < Bij(p,q) $B_{ij}(p,q)$= Bij(p,q)-T;

else

E_bit = Diff mod 2;

Collect it into a bit stream;

else

Bij(p,q)=Dij(p,q)+(Diff-E_bit)/2;

5. Use PRNG shuffle back the bit stream then reconstruct the secret image S.
6. Output image C (= the original image I).

IV. COMPARATIVE STUDY OF RDH TECHNIQUES

The table describes comparative study on various reversible data hiding techniques.

Techniques	**Computational complexity**	**Visual quality**	**Embedding capacity**	**Embedded data**
LSB modification	Low	Low	High	Grayscale Image
Histogram shifting	Low	Low	Moderate	Grayscale Image
Difference expansion	High	Moderate	Moderate	Audio and video
Interpolation technique	Moderate	High	Low	Uncompressed image used
Prediction error expansion	Low	High	High	Uncompressed image used

It includes various parameter like computational complexity, visual quality, embedding capacity, embedded data. Analysis of

comparative study of reversible data hiding in which the prediction error expansion techniques is provide better data security.

V. CONCLUSION

Various reversible data hiding techniques such as LSB modification, histogram shifting, Difference expansion, interpolation technique, and prediction error expansion are studied. Based on the study, prediction error expansion provides better security, better visual quality and it has high embedded capacity.

ACKNOWLEDGMENT

A special thank I give to my guide, Ms. Himani Parekh, Assistant Professor in the Department of Computer Engineering and Information Technology, Chhotubhai Gopalbhai Patel Institute of Technology(C.G.P.I.T). It was her efforts and constant guidance who encouraged me to come up with my work. Her knowledge, innovative ideas, dedication to work has always assumed me. I will be always thankful to her.

REFERENCES

[1] Jun Tian, "Reversible Data Embedding Using a Difference Expansion," *IEEE Trans. on circuits and systems for video technology*, vol. 13, august 2003.

[2] Bouslimi, Gouenou Coatrieux, Michel Cozic and Christian Roux "A Joint Encryption/Watermarking System for Verifying the Reliability of Medical Images," in *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, no. 5, pp. 891-899, Sept. 2012.

[3] Rathika R and S. Kumaresan, "Survey on reversible data hiding techniques," *3rd International Conference on Advanced Computing and Communication Systems*, Coimbatore, NOV. 2016, pp. 1-4.

[4] Mehmet U. Celik, Gaurav Sharma, A. Murat Tekalp, and Eli Saber, 2002, "*Reversible Data Hiding*," in IEEE ICIP.pp.157–160.

[5] Jun Tian, 2003, *"*Reversible *Data* Embedding Using a Difference Expansion,*"* in IEEE Transactions on Circuits and Systems for Video Technology, VOL 13, No 18. pp. 890-896.

[6] X. L. Li, B. Yang, and T. Y. Zeng,2011, "Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection," *IEEE Trans. Image Process.*, vol. 20, no. 12, pp. 3524–3533.

[7] Lixin Luo, Zhenyong Chen, Ming Chen, Xiao Zeng and Zhang Xiong, 2010, "Reversible Image watermarking using Interpolation technique,*"* IEEE Transactions on Information Forensics and security, vol. 5, no. 1, pp. 187–193.

[8] Manoj Kumar, Smita Agrawal, "Reversible data hiding based on prediction error expansion using adjacent pixels," in Security and Communication Network, 2016.

[9] M.Khodaei and K.Faez, "Reversible data hiding by using modified difference expansion," 2010 2nd International Conference on Signal Processing Systems, 2010, pp. 31-34.

[10] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 3, pp. 354–362, Mar.2006.

[11] X. Zhang, "Reversible data hiding in encrypted images," IEEE Signal Process. Lett., vol. 18, no. 4, pp. 255–258, Apr. 2011.