REVERSIBLE DATA HIDING METHODS: A COMPARITIVE STUDY ON TECHNIQUES AND PERFORMANCE

Saurav Sharma

*Dept. of Computer Engineering Delhi Technological University* New Delhi, India sauravsharmaco20b6[66@dtu.ac.in](mailto:66@dtu.ac.in)

Ritik Jain

*Dept. of Computer Engineering Delhi Technological University* New Delhi, India ritikjainco20b6[15@dtu.ac.in](mailto:15@dtu.ac.in)

Sachin Mishra

*Dept. of Computer Engineering Delhi Technological University* New Delhi, India sachinmishraco20b6[32@dtu.ac.in](mailto:32@dtu.ac.in)

***Abstract*—Reversible data hiding(RDH) has made significant progress in terms of theory and algorithmic studies since the beginning of 1990s. However, these developments occasionally need complicated and specialist information, which might provide challenges for researchers, particularly those who are inexperi- enced, in comprehending the fundamental concepts. In order to give readers a fundamental understanding of RDH, we aim to provide a concise summary of the primary RDH algorithms em- ployed in this work for image processing, along with an analysis of their distinct characteristics. We will examine widely used RDH frameworks and their typical alterations, beginning with conventional techniques such as lossless compression, histogram shifting, difference expansion, prediction-error expansion (PEE), integer transform (IT), and so on. Next, we will examine three widely used contemporary methods: pairwise PEE, pixel-value- ordering, and multiple histograms modification, and we will offer detailed evaluations of each approach. We shall evaluate these widely used techniques from four perspectives: the advance- ment of embedding frameworks, specific characteristics of the technology, expansions, and the current state of the art at the time of writing. In addition, we will discuss potential topics for future study that are derived from the initial motivations of the discipline.**

***Index Terms*—Reversible data hiding, prediction-error expan- sion, histogram shifting, pairwise prediction-error expansion, pixel-value-ordering, multiple histograms modification**

1. Introduction

RDH, also known as Reversible Data Hiding, is a well respected method employed to protect multimedia data [91]. It provides multiple benefits, as depicted in Figure 1. RDH facilitates covert communication across a shared channel. Embedding has a negligible effect on the quality, but it presents difficulties for adversaries to identify the marked carrier within large data sets. The receiver end of RDH performs two functions: it retrieves the data embedded and then precisely restores the original content or information of the carrier. RDH demonstrates exceptional proficiency in retrieving the carrier without any errors. RDH is a highly promising method or technique that can be employed in situations where main- taining data integrity is of utmost importance, such as in engineering images, business documents, and legal evidence. RDH is particularly advantageous as it can prevent irreversible

deformation. Presently, RDH processes commonly employ digital images as a means of research because they are simple to use and can be easily transmitted. The methods related to storage formats can be classified into three groups: RDH in encrypted images [6, 19, 37, 59, 76, 79, 81, 83, 87, 111, 117,

125, 127–129, 136, 137], RDH in uncompressed images [1,

2, 9, 14, 16, 30, 38, 42, 45, 47, 48, 51, 57, 62, 72, 86, 90, 94,

95, 101, 135, 138], and RDH in compressed images [13, 20,

25, 31, 32, 80, 85, 107, 113].

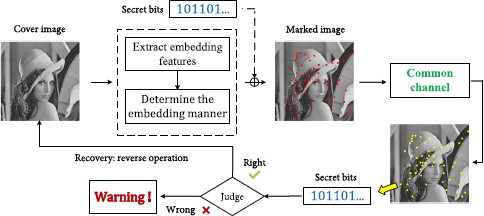


Fig. 1. The function and characteristics of RDH.

Among the three categories, uncompressed photographs have received the most significant focus from the academic community regarding RDH techniques. This is mostly due to the presence of excessive redundancy in the information representation of uncompressed pictures, which facilitates the creation of efficient reversible embedding systems. Many approaches undergo initial testing on uncompressed images before being extended to various carrier formats. Regarding RDH in uncompressed images, the earlier algorithms primarily emphasized the development of efficient methods for chang- ing pixels in a way that can be undone. The subsequent advancements in the field have been profoundly influenced by these fundamental algorithms. There are essentially five fundamental embedding frameworks. RDH is performed using lossless compression with the following parameters: [2, 16, 135, 138]. Difference expansion (DE) is applied with the parameters [30, 38, 95]. Integer transform (IT) is utilized with

the parameters [1, 9, 72, 86, 101]. Histogram shifting (HS)

is performed with the parameters [42, 45, 51, 62]. Prediction- error expansion (PEE) is applied with the parameters [14, 47, 48, 90, 94]. These systems share a fundamental notion, albeit having distinct technological characteristics: they utilize image redundancy to achieve reversible data embedding. Scholars have conducted additional research and development on em- bedding theories that are built upon standard RDH algorithms, with the aim of enhancing performance.

The primary criteria for evaluating the efficacy of a re- versible data hiding (RDH) system are typically the embed- ding capacity(EC) and the peak signal to noise ratio(PSNR). The carrying capacity of the method, often measured in bits per pixel(bpp), is indicated by the EC, which is often evaluated using the embedding rate (ER). A higher error rate (ER) implies that the cover picture has the capacity to accommodate a greater number of hidden message bits, potentially alleviating limitations on database storage. On the other hand, PSNR quantifies the level of distortion caused by embedding by assessing the similarity of each pixel between the annotated images and the original content. A greater Peak Signal-to-Noise Ratio (PSNR) indicates superior quality preservation and implies a less perceptible hidden message. In order to achieve a more optimal balance between embedding capacity and PSNR, RDH algorithms must adapt their data embedding strategies according to the content of the images. In response to this need, several adaptive RDH methods have been created, including pairwise prediction-error expan- sion(PEE) [68], pixel-value-ordering(PVO) [46], and multiple histograms modification (MHM) [49]. These methods have been developed with distinct objectives: enhancing embed- ding theory and addressing adaptive embedding challenges. However, in practical implementation, these systems often require customization to suit specific requirements and uses. This customization enhances the algorithm’s practical utility and ability to be applied to many scenarios. Several efficient algorithms have been published in various fields, including RDH in encrypted data, RDH for JPEG images, RDH for 2D- vector graphics, RDH for 3D-mesh models, RDH with the contrast-enhancement, RDH for color-images, robust RDH, visible reversible watermarking, RDH for videos, and RDH for human-visual system (HVS).

*A. Applications*

* Digital Rights Management (Buyer-vendor Protocol): To prevent unauthorized access, a seller of the content en- crypts content before transmitting it. The seller guarantees the uniqueness of each piece of content they sell by watermarking it and integrating the buyer’s information (fingerprinting) to deter unauthorized distribution and enable monitoring and tracing. Nevertheless, by incorpo- rating their thumbprint into the content and distributing it without charge, there is a possibility of falsely incrim- inating a genuine consumer.
* Cloud Storage: In order to safeguard their privacy, cus- tomers employ encryption to secure their private pho-

tographs, videos, and audio files before transferring them to the cloud. Cloud administrators incorporate encrypted data into various materials for administrative needs, in- cluding the data which is required for collection of statistics and indexing.

* Patient’s confidentiality: In hospitals, images and doc- uments of diagnosis that contain an information of the patient are encoded for identification and encrypted to ensure privacy. Nurses can extract this embedded infor- mation, which facilitates content linking. Physicians and other medical professionals have the ability to access original unaltered diagnostic images and documents, as well as patient identity information.
* Confidential Data: In military environments, subordinate personnel such as clerks and lower-ranking officers are able to retrieve hidden tags from encrypted documents, audio files, and video files for the purpose of organizing them. Personnel with higher level clearance, such as generals, are able to read labels which are embedded, and original files.
* Encryption of audio, picture, or video content for field re- porters before delivering it to headquarters. This ensures that only authorized personnel, excluding competitors, may access it for exclusive coverage. In order to address the issue of content counterfeiting, it is possible to incorporate the GPS position and sender ID of the field reporter for the purpose of verification.
* Surveillance video is recorded and to ensure privacy, certain techniques such as region-of-interest masking are used to encrypt the film selectively. Moreover, crucial information for identification, such as camera ID, time, and date, is included, particularly when the footage is used as a proof in a court.
* Labelling data is essential for ensuring privacy and con- fidentiality in the current era of extensive data develop- ment and collection, often driven by advancements in data-intensive scientific discoveries. The joint adaptation methodology is a practical method for effectively man- aging such data.

1. CLASSIC RDH TECHNIQUES

The process of creating RDH for uncompressed pictures typically involves two parts. In the initial stages, we focus on constructing several RDH frameworks and devised five conventional techniques:

* lossless compression,
* Prediction Error Expansion,
* Integer Transform,
* Histogram Shifting,
* Difference Expansion.

The second step involves the development of the theory and the exploration of more efficient methods. It aims to enhance performance by addressing the issue of unnecessary utilization. The classical PEE framework is utilized as the main development framework for implementing efficient algorithms in this stage. These algorithms include

TABLE I

Summary of the RDH Techniques

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RDH**  **Techniques** | **Description** | **Expectable ER** | **Embedding Distortion** | **Reference** |
| Lossless  compression | After making some room in the image by  compressing it, embed the pieces. | 1.5 bpp | High | [2, 16, 135, 138] |
| DE | Use difference expansion for data embed-  ding. | 0.9 bpp | Medium | [30, 38, 95] |
| IT | Apply the integer transform for data embed-  ding. | 2.17 bpp | Medium | [1, 9, 72, 101] |
| HS | According to the histogram, embed the bits. | 1 bpp | Low | [17, 42, 62] |
| PEE | For data embedding, alter the prediction-  error histogram. | 1.8 bpp | Low | [34, 47, 90, 94] |

* Pixel Value Ordering,
* Multiple Histograms Modification,
* pairwise PEE.

In this section, we will focus on the conventional RDH techniques implemented in the early phase and briefly discuss their expansions. An elementary illustration is visible in Table I.

*A. Compression-based RDH Technique*

RDH algorithms were initially developed primarily through the use of lossless compression methods [2, 16, 135, 138]. To utilize the available positions for information encryption, the partial picture material undergoes lossless compression to provide a compact representation with reduced size. Because compression is lossless, it is possible to restore the image. The compression method the embedding performance are closely related. One objective is to enhance the balance between performance and compression efficiency. Celik et al.

[2] proposed the classic approach of least significant bit (LSB) based reversible data hiding (RDH). This method employs quantization to determine the minimum value of the cover pixels, which is then utilized for embedding. The information is recursively incorporated in [135] through the decompression and compression methods of an entropy coder. Zhang [138] proposed partitioning the cover pixels into many segments and applying the optimal value transfer rule to change them. This method effectively accomplishes reversible embedding using a range of compression algorithms. The RDH approach, which

where *m* represents the binary message bit. To put it simply, the disparity between the two pixels is increased by 2(*p*2 − *p*1) + *m* in order to include one additional bit. The least significant bit (LSB) of the magnified difference can be utilized by the recipient to compute the embedded bit, whereas the inverse transformation is employed to recover the original cover pixels.

In the context of data embedding (DE) in digital images, the embedding rate (ER) can reach a maximum of 0.5 bits per pixel (bpp) when using a single layer embedding technique. Additionally, the average values of two adjacent pixels before and after the embedding process stay unchanged. Several variations of RDH have been evolved from this fundamen- tal method. However, the DE technique in the initial stage sometimes fails to include the discriminating processing for the pixel pairings during embedding. Hence, the utilization of multi-layer embedding may lead to substantial distortion while aiming for a higher ER. Furthermore, to reduce the amount of embedding capacity used, DE-based approaches typically rely on the location map to identify pixels that are not available and then compress them without losing any data.

*C. IT-based RDH Technique*

The primary objective of IT-based RDH technique is to modify a sequence of pixels at the same time in order to embed multiple bits. In the embedding process, it is possible to modify a pixel sequence (*p*1*, p*2*, . . . , pn*) as a single entity in order to incorporate *n* − 1 bits. When *n* = 2, DE can be considered as a specific case of IT. Based on the citation [9],

relies on compression, has demonstrated its effectiveness. the specified pixels (*p , p , . . . , p*

*n*

*j*=1

2*pi* − ⌊2*p* − 1 Σ*n*−1 *mj*⌋ + ⌊*p*⌋*,* if *i* = 1

) are recognized as

However, a drawback of attempting to incorporate additional

hidden information into the image is that it can become

*p*

(

˜1 ˜2 ˜*n*

difficult to preserve its intended meaning. When there are

*i*

*n*

*j*=1

*j*

*i*−1

enormous amounts of something, it is common to see clear

˜*i* = 2*p* − ⌊2*p* − 1 Σ*n*−1 *m* ⌋ + ⌊*p*⌋ + *m ,* if *i* ≥ 2

(1)

visual distortion.

*B. DE-based RDH Technique*

The DE-based RDH algorithm applies the Haar transfor- mation to modify each of the two adjacent pixels for data embedding. The reversible embedding and extraction of data is facilitated by the commonalities between the two pixels. The designated pixels for a pair of pixels (*p , p* ) are *p* =

The message bit, represented as *mj*, belongs to the set 0, 1. The jth index range starts from 1 and ends at n-1, and the ith index ranges starts from 1 and ends at n. When the value of n is more than 2, it is evident that an information technology- based resource discovery and handling (RDH) strategy may offer a higher capacity compared to a data embedding (DE) approach. For a large value of n, the maximum error rate is around 1 bit per pixel. The succeeding works [72, 86, 101]

1 1

˜1 ˜ 2 ˜1

2*p*1 −

˜

2 (*p*1 + *p*2) and *p*2 = 2*p*2 − 2 (*p*1

+ *p*2) + *m*

significantly improve the performance in terms of ER and

PSNR. Furthermore, considering many pixels simultaneously enhances the stability of the integer average, rendering it more resistant to alterations.

*D. HS-based RDH Technique*

Ni et al. [62] proposed the RDH with HS technique. This technique utilizes gray-scale histogram to alter the pixels. In order to include the hidden information, the histogram’s topmost bins are selected for the expansion in HS. This involves grouping pixels based on their grey values. There will be no ambiguity when reversible alterations are applied to the pixels from different categories. Pixels can be classified into three broad categories: moving, expanding, and unchanging pixels. They perform multiple jobs during the process of embedding. Usually, only the shifting and enlargement of pixels are altered.

The bins of histogram corresponding to pixels having values of b will extend to incorporate an extra bit. The histogram bins are adjusted to the left for pixels with values less than or equal to a. Figure 2 illustrates an instance of HS-based customization. The expansion bins are chosen as (a,b) = (57, 59) at this location. The three expansion-pixels (57, 59, 59) are adjusted based on the bits with the secret (0, 0, 1), resulting in the pixels mark (57, 59, 60) as per the HS-modification rule. Pixels highlighted in blue color are the pixels which might be shifting. The values are incremented or decremented by 1 without encoding any bits. The HS predictable modification approach enables receiver to detect bits which are hidden and perform the recovery in a reverse way, meaning it can determine the cover pixels.

*p*˜*i* − 1*,* if *p*˜*i > b*

 ˜

˜

*p*

=

*p ,* if *a* ≤ *p*

≤ *b*

(3)

±1 attached to them or remain unchanged, depending on the message bits. Meanwhile, in order to guarantee reversibility,

The expansion pixels during embedding will either have

the moving pixels will be appended with either 1 or −1.

*i i* *i*

if *pi < a*

˜

*p*˜*i* + 1*,*

The embedded bits extraction depends on category of pixel

The specific pixels (*p , p , . . . , p* ) that are designated for indicated. Specifically, if the pixel *p* belongs to the set a -

1 2 *N* ˜*i*

˜

the cover pixel sequence (*p , p , . . . , p*

) are determined.

1, b + 1, then m = 1. Otherwise, if *p*

belongs to the set

˜1

˜2

˜*N*

*i*

a, b, then m = 0. The approach of embedding of Histogram

*pi* + 1*,* if *pi > b pi* + *m,* if *pi* = *b*







˜*i* = *pi, a < pi < b*

if

*p*

(2)

Shifting is simple, and more instinctive as compared to the techniques of DE and IT. Alteration methods do not require intricate mathematical calculations, and the performance may be directly assessed by simulating Histogram Shifting before

*pi* − *m,* if *pi* = *a*



*pi* − 1*,* if *pi < a*

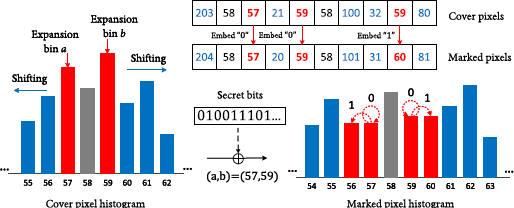


Fig. 2. Example of the modification of the pixels and shifting of histograms in HS based RDH Technique

Assuming that N is representing the number of pixels, m representing the message bit, and (a,b) represent the two expansion bins, where a and b are both greater than 0 and less than 255. In order to achieve reversible embedding, it

embedding. This approach possesses significant flexibility. In general, the design of embedding of HS is efficient for features of statistics derived from some other coefficients, such as his- togram of quantified discrete cosine-transform coefficients [32, 113] and histogram of deep convolutional-network parameters [17], among others.

*E. PEE-based RDH Technique*

Prediction Error Expansion introduces prediction in the RDH, which further enhances the performance of embedding by using correlation exploitation more effectively. The pre- diction errors(PE) are sequentially calculated based on cover pixels and their adjoining pixels. This process allows for the generation of a prediction error histogram (PEH).

The PEH is a mathematical function that defines the number of pixels gathered in a given sequence. It is denoted as *h*(*e*) = #{1 ≤ *i* ≤ *N* : *ei* = *e*}, where the function #{·} represents the counting of pixels. The expression *ei* = *pi* − *p*∗ represents the pixel’s prediction error, where *pi* is the actual

*i*

*i*

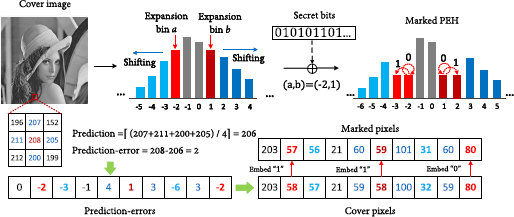
is important to store the parameters (a,b) as side information.

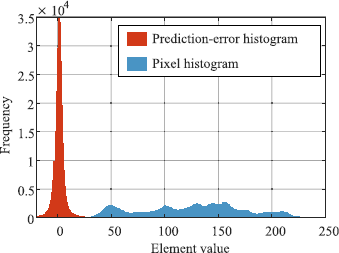
pixel value and *p*∗

is the predicted pixel value. Within the

Usually, the bins of expansions, which usually have the highest peaks in the pixel histogram, are selected to reflect the grey values that correspond to the largest categories. Expanding pixels are pixels that have values within the range of a and b, while shifting pixels are pixels that have values greater than b or less than a. This rule stipulates that while embedding, the distributive nature of the pixel-value histogram will be changing in a predictable manner. Pixels with values greater than b will have their histogram bins moved to the right.

context of PEE, the PEs will categorize the pixels based on their attributes rather than their gray values. The produced PEH closely resembles the Laplacian distribution. The dis- tribution of the PEH exhibits a higher degree of sharpness and concentration when compared to the pixel histogram. Consequently, the compression of visual data results in a more condensed representation, thereby creating additional capacity for embedding data. Figure 3 presents a comparison between the histogram derived from PEs and pixels. PEH exhibits a



Fig. 3. The comparison of pixel histogram and Prediction-error Histogram for Lena image

clear concentration at zero, with significantly larger values in its bins. The PEE-based technique utilizes a similar approach to HS for the embedding process. One distinction lies in the fact that PEE selects the expansion pixels based on the values

Fig. 4. Example of modification of pixels and the histogram-shifting in the PEE RDH technique.

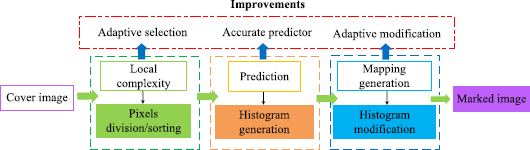


Fig. 5. Embedding modules of PEE, and the used enhancements.

and *p*∗. Additionally, the parameters (*a, b*) are acquired from

of PEs. The highlighted pixels (*p , p , . . . , p*

if

) are obtained *i*

in the following manner:

˜1 ˜2 ˜*N*

the side information. The value of the cover pixels is restored

to *pi*

˜



The user’s text is empty. The value of the cover pixels

— 1 if its designated PE (Pixel Error) is greater than *b*.

*pi* + 1*,*



if *e* = *b*



*ei > b*

is preserved as

itself, provided that its designated PE

*pi* + *m,* *i*

*pi* = *pi,* if *a < ei < b*

its designated PE is smaller than a. Similarly, the embed-

˜



(4)

falls within the range of a and b. No information provided. The value of the cover pixels is restored as *pi* + 1, when

*pi* − *m,* if *ei* = *a*

*pi* − 1*,* if *ei < a*

ded bits will be extracted as ”1” if *ei* belongs to the set

˜

*p*˜*i*

˜

The two expansion bins, a and b, are determined adaptively based on the distribution of PEH and the provided capacity. The values of a range from -255 to 0, while the values of b range from 0 to 255. It should be noted that the expansion bins in PEE are now defined as the values of PEs instead of the pixel values. Just like in HS, the histogram bins of PEH will undergo predictable modifications during the embedding process. Figure 4 displays the structure of the PEE framework. In contrast to the image depicted in Figure 2, the significant difference is in the categorization of expansion pixels, shifting pixels, and unaltered pixels based on the PEs. In the context of PEE, it is crucial to maintain the original prediction for a pixel both before and after embedding in order to guarantee reversibility. The recipient should acquire identical forecasts as those of the giver. An effective approach is to ensure that the identical context utilized for prediction may be retrieved following the embedding process. The local environment for the receiver can be made identical by scanning and processing the designated pixels in reverse order, as indicated by the popular predictors [8, 14, 90, 94]. The PEs are thus configured. The estimated data obtained from the annotated image can be utilized to accurately categorize the annotated pixels. There are three instances to consider. In each case, the pixel *pi* is noted and its PE is calculated as the difference between *pi*

˜

˜

˜

a - 1, b + 1, and extracted as ”0” if *ei* belongs to the set a, b. PEE combines the benefits of both DE and HS. The computational complexity is deemed acceptable, and the embedding efficiency is higher. The collection of embedding modules offers researchers numerous chances for exploration and enhancement. Figure 5 depicts the progression of PEE and its associated enhancements.

1. RDH TECHNIQUES FOR ADAPTIVE EMBEDDING

In the subsequent section three widely used RDH schemes and their extensions: MHM , pairwise PEE, and PVO. These schemes represent advancements over the traditional approaches outlined earlier and possess their own distinct technological features. Furthermore, the presentation of these typical techniques will be provided in the end.

1. *Multiple Histograms Modification*

MHM is a comprehensive framework that encompasses the principles and concepts of PEE. Within the context of MHM, the classification of cover pixels is determined by considering both the properties of PEs (Prediction Error) and the degrees of texture. In PEE-based approaches, the smooth pixels typically have PEs with values close to zero. This is due to the fact that smooth regions typically exhibit strong correlations. Therefore,

TABLE II

An examination of the existing methodologies for adaptive embedding.

|  |  |  |
| --- | --- | --- |
| **Method** | **No. of Citations** | **Description** |
| Pixel sorting [90] | 967 | The sorting approach mostly modifies the smooth pixels. |
| Pixel selection [47] | 711 | Additional covert information is concealed inside the fine details of  the pixels. |
| Multi-level embedding [18] | 94 | The pixels are divided according to their smoothness levels. Each level  is assigned a distinct embedding capability. |
| MHM [49] | 342 | First, we make different PEHs to set the suitable parameters for each  sub-PEHs. Different approaches are used for pixels based on the local context. |
| Pairwise PEE [68] | 494 | The PEH modification is enhanced in the two-dimensional space by  leveraging the correlations of second order between pixels. |
| PVO [46] | 485 | Choose the starting and ending points from the sequence of pixels  which are arranged from every single block for insertion. |
| AGM [89] | 40 | The PE with a high frequency is embedded with more than one bit  and the low-frequency PEs are discarded for embedding. |
| Reduce shifting [34] | 135 | The excess moving pixels are removed in an adaptive manner based  on the surrounding environment. |

it is more feasible to acquire precise forecasts for the seamless pixels. The Prediction Error Histogram (PEH) derived from those places exhibits a more focused distribution. By including the message into the seamless pixels, the embedding process will experience reduced distortion. A viable and logical ap- proach is to assign separate expansion bins to pixels with different amounts of texture. In Reference [49], the N cover pixels (*p*1*, p*2*, ..., pN* ) are classified into M categories. Each category is individually handled using distinct parameters. Initially, the pixels that possess an equivalent amount of complexity are gathered and grouped in the corresponding subset. Now to determine the complexity level, we analyse the specific circumstances and conditions of the given situation. Furthermore, each subset will generate a Prediction Error Histogram (PEH) consisting of two expansion bins. To clarify, the original PEH will be partitioned into M sub-PEHs that fulfill the condition h(e) = hc(e) for c ranging from 1 to

Σ

M. In this context, ”hc” refers to the sub-PEH that counts

the PEs with complexity of ”*ci*”, where ”c” belongs to the set 1, 2, ..., M. Consequently, the cover pixels are categorized based on the pair (*ei, ci*). Furthermore, by the implementation of exhaustive search, the most optimal expansion bin set (*ac, bc*)M c=1 is determined adaptively, taking into account the specific characteristics of each subset. Thus, each subset will have its own distinct mappings for the pixels. This constitutes the modification model with several tiers. Figure 7 displays an illustration of the embedding process based on MHM, as described in reference [49]. The cover pixels are

less than *ei* and *bc*.

* + The cover pixel is enlarged to *pi* − *m* in order to encode a single secret bit, only when *ei* is equal to *ac*.
  + When the value of *ei* is less than *ac*, the cover pixel is adjusted by subtracting 1 to create a space.

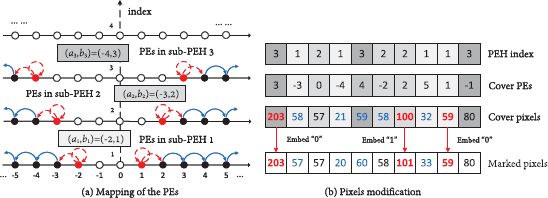


Fig. 6. The data embedding example of MHM.

Alteration of the pixels is done at various levels based on their corresponding embedding parameters. Similarly, it is imperative to assign minimal storage capacity for storing the chosen parameters. In the course of the extraction procedure, the cover pixels are restored in a manner opposite to the alteration. Generally, MHM exhibits flexibility by modifying the set of expansion bins. Precise alteration can be achieved by choosing appropriate expansion bins for the pixels from various texture regions. Subsequently, Ou et al. made further enhancements by extending its framework to a more com- prehensive structure. The *cth* histogram will find *Kc* pairs of expansion bins {(*ac,k, bc,k*)}*Kc* that fulfill the conditions

assigned varied gray levels based on their histogram indexes.

*ac,k*

*<* · · · *< a*

*c,*1

*k*=1

≤ 0 ≤ *bc,*1 *<* · · · *< bc,k*

. During the

The alterations for the cover pixels corresponding to the sub- PEH hc can be summarized into five scenarios, based on the expansion bins (*ac, bc*)M c=1 for each subhistogram.

* + When the value of ei is greater than bc, the cover pixel is incremented by 1 to shift the space.
  + The cover pixel is enlarged to *pi* + *m* in order to encode one hidden bit, when *ei* is equal to *bc*.
  + The cover pixel remains intact when the value of ac is

process of embedding, the alterations made to the cover pixels in each sub-PEH are categorized into five distinct instances. The distinction lies in the adjustment of the shifting step size for each pixel. Specifically, the identified pixels are determined in the following manner:

* + Case 1: If the value of *ei* falls between *bc,k* and *bc,k*+1, the cover pixel should be relocated to the right by *k* steps, meaning that *pi* = *pi* + *k*.
  + Case 2: If the value of *ei* is equal to *bc,k*, then the pixel is enlarged to obtain the designated one by adding (*k* − 1) + *m* to *pi*.
  + Case 3: If the value of *ei* falls between *ac,*1 and *bc,*1, the cover pixel remains unaltered.
  + Case 4: If the value of *ei* is equal to *ac,k*, then the marked pixel is calculated as *pi* − (*k* − 1) − *m*. The user’s text is empty.
  + Case 5: If the value of *ei* is between *ac,k*+1 and *ac,k*, then the pixel in the cover should be moved to the left by *k* steps, so that *pi* = *pi* − *k*.

1. *Pairwise PEE*

In the traditional PEE [90, 94], pixel modification is de- termined using low-dimensional statistical analysis of the picture data. The mapping derived from the one-dimensional Prediction Error Histogram (PEH) lacks the ability to fully exploit the intricate interaction between neighboring PEs. Pairwise PEE [68] provides a viable solution by taking into account the alteration in a space with more dimensions. This approach initially groups adjacent Prediction Error (PEs) in pairs to categorize the pixels and produce a two-dimensional PEH. Next, the expanding pixels, shifting pixels, and unaltered pixels are identified by creating modification rules for the 2D PEH bins, specifically by building a 2D mapping. Next, we will provide a concise overview of the particular structure outlined in approach [68] and examine its distinctive features.

TABLE III

The Conventional PEE’s Nine Potential PE Pair Mappings

|  |  |  |
| --- | --- | --- |
| **Case** | **Mapping directions** | **Condition** |
| (a) | (*x, y*) *→ {*(*x* + 1*, y*)*,* (*x, y*)*,* (*x* +  1*, y* + 1)*,* (*x, y* + 1)*}* | *x* = *b, y* = *b* |
| (b) | (*x, y*) *→* (*x, y* + 1) | *x < b, y > b* |
| (c) | (*x, y*) *→* (*x* + 1*, y*) | *x > b, y < b* |
| (d) | (*x, y*) *→* (*x, y*) | *x < b, y < b* |
| (e) | (*x, y*) *→* (*x* + 1*, y* + 1) | *x > b, y > b* |
| (f) | (*x, y*) *→ {*(*x, y* + 1)*,* (*x, y*)*}* | *x < b, y* = *b* |
| (g) | (*x, y*) *→ {*(*x* + 1*, y* + 1)*,* (*x* +  1*, y*)*}* | *x > b, y* = *b* |
| (h) | (*x, y*) *→ {*(*x* + 1*, y* + 1)*,* (*x, y* +  1)*}* | *x* = *b, y > b* |
| (i) | (*x, y*) *→ {*(*x* + 1*, y*)*,* (*x, y*)*}* | *x* = *b, y < b* |

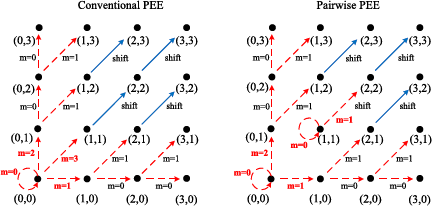


Fig. 7. Two-dimensional mapping comparison, with the standard PEE expansion bin is denoted by the symbols b = 0.

In the context of paired PEE, the sequence

{(*e*1*, e*2)*,* (*e*3*, e*4)*, . . . ,* (*eN*−1*, eN* )} is created by merging the two PEs that are next to each other. In a similar manner, the PEH is generated, and the mapping is made in accordance

with its distribution. The distinction lies in the fact that the PEH is produced inside a 2D coordinate system, specifically expressed as *h*(*x, y*) = #1 *< i* ≤ *N/*2 : *e*2*i*−1 = *x, e*2*i* = *y*, where *h* represents the frequency count for the PE pairings of (*x, y*). Next, the expansion bins are dynamically chosen

to create a two-dimensional mapping. While the embedding process is being carried out, the pixels that are adjacent to one another will undergo collective changes that will either conceal or shift a single bit. This method enables greater flexibility in the directional shift of each PEH bin. The modification approach of pairwise PEE is compatible with 1D mapping-based methods, serving as an extension of PEE. The alteration of the traditional PEE, as specified in Equation (4), can also be elucidated in the two-dimensional space. For the standard PEE, there are nine possible mappings for PE pairings (*x, y*). The items are enumerated in Table III. It should be noted that only the mapping for the PEs with positive values is shown for the sake of being concise. The authors in Reference [68] devised an alternative modification model that relied on an examination of embedding efficiency. The clear enhancement is case (a) specified in Table III. For the pair (*b, b*), the pairwise PEE converts the mapping outputs to {(*b, b*)*,* (*b* + 1*, b*)*,* (*b, b* + 1)}. Meanwhile, the pair (*b* + 1*, b* + 1) is chosen as the expansion bin, and the self-mapping is included in its outputs. Figure 8 depicts two distinct 2D mappings. The pairwise PEE [68] utilizes a 2D mapping that presents a novel modification framework.

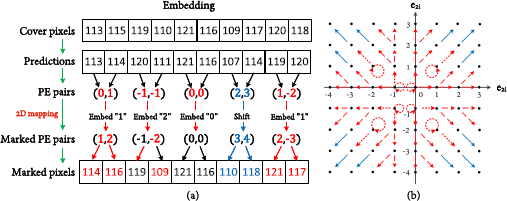


Fig. 8. The alteration of pixels using pairwise pixel error estimation (PEE).

(a) The process of embedding in the paired PEE. (b) The entire process of mapping the pairwise PEE.

The performance of the recently developed 2D mapping has been exhibited [68]. According to the established 2D mapping, it is conceivable that the covert message is no more encoded in binary format. Figure 9 displays the data embedding of paired PEE. The paired PEE offers a novel approach to representing data in the context of RDH (Re- versible Data Hiding) design. The high-dimensional-based modification has gained significant interest, and numerous 2D RDH approaches have been presented. Subsequent studies have demonstrated that adaptively modifying the expansion bins based on the provided histogram distribution is beneficial

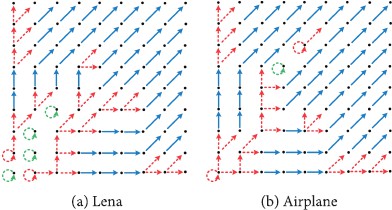


Fig. 9. The 2D mapping generated using the adaptive approach for the Lena and Airplane test image with an EC value of 10,000 bits.

for enhancing performance. The proposed approach involves producing adaptive 2D mapping based on the given strategy [5, 15, 65, 84, 114, 131]. Ou et al. [65] introduced two techniques for creating a 2D map of a given image. The initial approach entails exploring a substantial number of potential solutions. The distortion-capacity ratio of each candidate can be compared in order to determine which mapping technique is the most effective. The utilization of the optimal probability matrix is an alternate method that can be utilized. On the basis of the evaluation of the mapping probability, the final modification model is validated. The modification method is depending on the image, allowing for improved performance. Chang et al. [5] and Zhang et al. [131] sequentially developed an adaptive 2D mapping technique. In the study cited as Reference [5], the ideal two-dimensional mapping is initially set as a random solution and then determined using an iterative updating method.

Two examples are presented here in Figure 10. The two- dimensional mapping can be modified according to the con- tent of the image. An alternative approach was proposed by Zhang and Ou [131] as a means of improving the mapping optimization. The method entails first sorting the acquired PE pairings according to their frequencies and placements, and then modifying the mapping inputs of each of these pairs on an individual basis. It was suggested by Fan et al. [15] that the pairwise PEE could be combined with a skewed histogram in order to improve performance. In addition, they made use of the distorted PEH-based two-dimensional mapping that was included in the MHM framework, which allowed them to carry out modifications that were more highly effective. The authors of References [3, 4] have recently created a method to expand the pairwise modification methodology to color images and create a related optimization strategy for 3D mapping. The mapping generation model in Reference [3] is trained using the reinforcement learning approach known as double deep Q-network.

1. *Pixel value ordering*

PVO is a method that makes predictions by taking into account the similarity of pixels inside a block. In contrast

to the approach of producing predictions for each individual pixel, Li et al. suggested utilizing only the maximum and minimum pixel values within a block for data embedding. The maximum and minimum pixels are determined by the penultimate greatest and smallest pixels, respectively. The prediction takes into account both the coordinates of the pixels and their grayscale values, resulting in a higher level of accuracy. In addition, just two pixels are altered inside a block, resulting in a lower level of distortion compared to traditional PEE approaches. Given the similarity of the embedding processes, we will focus solely on explaining the modification of the maximum for the sake of simplicity. The pixels in the block (*p*1*, p*2*, . . . , pn*) will be arranged in ascending order depending on their gray values, resulting in the sequence *pσ*(1) ≤ *pσ*(2) ≤ *. . .* ≤ *pσ*(*n*). Here, the mapping *σ* provides a bijective relationship between the new index and the original one. The new order of two pixels with identical values is determined by their original indexes, so that *σ*(*i*) *< σ*(*j*) when *i < j*. Within the ordered sequence, only the pixel with the highest value is anticipated and altered.

Firstly, it employs the closest element as the forecast, and the prediction error (PE) is computed as *e*max = *pσ*(*n*) − *pσ*(*n*−1). It is evident that the PEs (Prediction Error) in PVO are always larger than or equal to 0. Furthermore, a PE with a value of 1 is utilized for expansion purposes. The secret bits are embedded

by modifying the biggest pixels in a reversible manner. More precisely, the pixels with pixel elements labeled as ”0” will remain unaltered and will not be included in the embedding process. The pixels with PEs ”1” are enlarged as *pσ*(*n*) + *m* in order to hide the secret bit. If the associated pixel error exceeds 1, the pixel is displaced and increased by 1. Given that the maximum pixel can only be increased by 1 or remain the same, the maximum value will always be the largest one within the designated block. The order of pixel values in the marked block remains unaltered, ensuring reversibility. The restoration and extraction of the indicated pixel are performed in a reverse manner. Figure 11 illustrates the process of data embedding and extraction using the PVO approach.

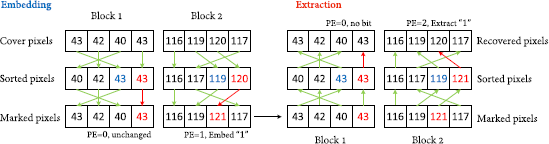


Fig. 10. Diagram depicting the process of embedding and extracting data in a PVO-based PEE system.

Within the PVO framework, the process of identifying the most comparable pixels in a block involves arranging the pixel values in a sorted manner. It has the ability to acquire precise forecasts and produce the PEH with reduced entropy, hence enhancing the efficiency of the embedding process. When it comes to embedding with a limited storage capacity, PVO typically demonstrates greater performance in terms of Peak Signal-to-Noise Ratio (PSNR). Several enhancements to

PVO have been suggested, including improved PVO (IPVO), PVO-k, k-pass PVO, pixel-based PVO (PPVO), pairwise PVO, MHM-based PVO, adaptive block complexity computation, and dynamic block size determination.

1. *IPVO:* In IPVO, the calculation of PEs is redefined by considering the original positions of the target pixel and its forecast. The PEs are calculated for the sorted block pixels (*pσ*(1)*, pσ*(2)*, . . . , pσ*(*n*)) in the following manner:

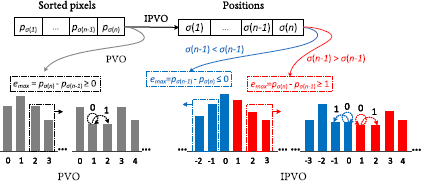


Fig. 11. Comparing the alteration of the PEH for PVO and IPVO during the embedding process.

* + The prediction can be determined by calculating the difference between the current value and the previous value, which is represented by the PVO equation: *e*max = *pσ*(*n*) − *pσ*(*n*−1).
  + If the initial index of the maximum is greater than the

index of the prediction, then the value of PE will be negative, specifically *e*max = *pσ*(*n*−1) − *pσ*(*n*).

By using this particular definition, the resulting PEH closely resembles the distribution found in traditional PEE analysis, specifically a Laplace-like distribution with its center at zero.In Figure 12, a comparison is shown between the histogram modification strategies used for PVO [46] and IPVO [73]. There is a possibility that the shifting of PEs could take place in two different directions in IPVO because of the symmetrical distribution of the histogram. It has been decided that the expandable PEs will be placed in the two highest places, espe- cially PE ”0” and PE ”1”. During the process of embedding, only the pixels that have PEs with values of ”0” or ”1” are chosen to carry the secret bits. These pixels are used to encode a binary value of ”0” or ”1” by either preserving the original value or increasing it by 1, respectively. For pixel errors more than 1 or less than 0, the relevant pixels are moved to the right, meaning they are increased by 1, in order to guarantee reversibility. It is important to observe that in IPVO, the direction in which the PEs change is distinct from the direction in which the pixels move. Regarding the cover pixels, they undergo either an increment of 1 or remain unaltered during the embedding process. In IPVO, several PEs are employed to transport the secret bits, as opposed to standard PVO. The classification process is more sophisticated, enabling a greater payload capacity and improved visual quality.

1. *PVO-k:* PVO-k is an expanded version of PVO that encompasses a wider range of possibilities and consid- ers a greater number of situations. In traditional PVO

(Pixel Value Ordering), just the final pixel in the sequence (*pσ*(1)*, pσ*(2)*, . . . , pσ*(*n*)) is utilized for embedding. However, there is one exceptional scenario when a single block has many pixels that are the largest, specifically when *pσ*(*n*−2) =

*pσ*(*n*−1) = *pσ*(*n*). The standard PVO technique leads to the

omission of several blocks throughout the embedding process.

In PVO-k, the ability to modify the top *k* biggest pixels simultaneously in order to embed one bit is implemented to resolve this problem. For the block of pixels that meet the condition *pσ*(1) ≤ *. . .* ≤ *pσ*(*n*−*k*) *< pσ*(*n*−*k*+1) = *. . .* = *pσ*(*n*), there are *k* pixels that are the largest. The prediction is made

by selecting the second largest pixel. These *k* pixels can be assigned to the same PE, where *e*max is equal to *pσ*(*n*) minus *pσ*(*n*−*k*). By choosing PE ”1” as the expansion bin, the embedding and extraction procedures are virtually the same as those in PVO. It is important to observe that the *k* biggest pixels need to be updated as a complete unit in order to

maintain reversibility. This means that each block will still conceal only one bit. The authors in [67] have demonstrated that the utilization of PVO-2 and PVO-1 in a flexible manner can lead to improved embedding performance.

1. *k-pass PVO:* The goal of k-pass PVO is similar to that of PVO-k, which uses more than one pixel in each block to insert information. The distinction lies in the fact that the chosen multiple pixels in k-pass PVO may not be identical. In the k-pass PVO algorithm, with a sequence of pixels that satisfies

the condition *pσ*(1) ≤ *. . .* ≤ *pσ*(*n*−*k*) ≤ *pσ*(*n*−*k*+1) ≤ *. . .* ≤

*pσ*(*n*), each of the *k* biggest pixels is assigned to a PE. The

forecast commences with the highest value and persists for the remaining duration. To begin, the pixel with the index *n* − *k* in the ordered sequence is chosen.

The greatest prediction and the PE can be calculated using the formula *en* = *pσ*(*n*) − *pσ*(*n*−*k*). Subsequently, the PEs of the remaining *k* − 1 biggest pixels are ascertained using *en* as a basis. For the pixel *pσ*(*i*), where *i* belongs to the set of

integers from *n* − *k* + 1 to *n* − 1, there are two possible values for its PE. If the highest PE (Pixel Energy) is greater than 1, then the PE is calculated by subtracting the value of the *n* − *k* indexed pixel from the value of the maximum pixel,

i.e., *ei* = *pσ*(*i*) − *pσ*(*n*−*k*). Alternatively, the highest value becomes the forecast, namely, the PE is computed as *ei* =

*pσ*(*i*) − (*pσ*(*n*) − 2).

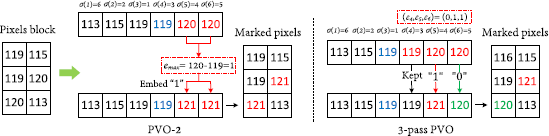


Fig. 12. Embedding comparison of PVO-k and k-pass PVO.

Through the use of this prediction technique, every pixel that is to be transformed will possess a Prediction Error (PE), and the original sequence can be reinstated following the embedding process. During the process of embedding, pixels that have a value of ”1” in the PEs (Pixel Expansion) are

identified as expansion pixels. Pixels with PEs more than 1 are categorized as shifting pixels, while the remaining pixels are considered unaffected. The k-pass PVO approach involves embedding one bit into each individual pixel, rather than processing them collectively. The maximum capacity of each block is *k* bits. Figure 13 illustrates the contrast between PVO-k [67] and k-pass PVO [23]. In the PVO-k algorithm, the two largest pixels, located at coordinates (120*,* 120), are concurrently enlarged to (121*,* 121) in order to incorporate the hidden bit ”1”. Within the k-pass PVO, the pixels located at coordinates (120*,* 120) contain the binary values ”1” and ”0” respectively, and they are the two largest pixels in the image. In the k-pass PVO, it is evident that the arrangement of the marked pixels is not the same as that of the cover pixels. Specifically, the pixel with a value of 120*σ*(5) is increased to 121, resulting in it becoming the largest one in the marked sequence. Nevertheless, the extraction result will remain unaffected as the retrieved bits can be further fine-tuned according to their original places before being arranged.

1. *PPVO:* As part of the PPVO methodology, the prediction model of PVO is combined with the usual predictors that are utilized in PEE calculations. In particular, each and every pixel that is contained within the block will be assigned to a processing element for processing. At some point throughout the process of prediction, the target pixel and the pixels that are immediately adjacent to it are brought together and combined to form a pixel block. Within the block, if the pixel that is being targeted is the largest element, then it will be expected based on the pixel that is immediately adjacent to it that is the largest. In the event that the element is the smallest, the outcome of its prediction will be the lowest value among the elements that are adjacent to it. Alternatively, if this pixel is not altered, it will be designated as an unmodified pixel during the embedding process. An exceptional scenario occurs when all neighboring pixels have identical values, causing the target pixel to become the smallest among them. In this scenario, the anticipated target pixel is determined by subtracting one from the greatest value. In the process of embedding, the type of alteration that is selected can be determined by studying the relationship that exists between the target pixel and the pixels that are adjacent to it. When the current pixel is equivalent to either the greatest value or the lowest value among its nearby pixels, which is expressed by the symbol PE = 0, the pixel is expanded in order to convey the information.If the value of the target pixel is higher than the maximum value of the pixels that are next to it, then the value of the target pixel will be increased by one in order to shift. Likewise, if the current pixel is smaller than the minimum element, it will be decreased by 1. By employing PPVO, the capacity to conceal supplementary bits is enhanced due to the increased number of embeddable pixels.
2. *Pairwise PVO:* From the aforementioned schemes, it can be observed that the expansion bins are consistently chosen as either PE ”1” or ”0.” It does not possess adaptive variations. 2D mapping is an excellent adaptive approach that offers more choices for modifying PVO, hence significantly enhancing

performance. Ou et al. [63] expanded the pairwise partial expected error (PEE) to partial variance optimization (PVO) by taking into account the two largest elements in each block as a pair.

The third largest pixel in the ordered pixel sequence *pσ*(1) ≤ *pσ*(2) ≤ *. . .* ≤ *pσ*(*n*) is used to estimate the values of the pixels *pσ*(*n*) and *pσ*(*n*−1). In order to ensure reversibility, it is necessary to impose constraints on the pairing order. The PE pair (*e*1max*, e*2max) is formed by taking into account both the positions and the gray values. The first PE of the pair is calculated as the difference between the lower indexed one, represented by *e*1max = *pu* − *pσ*(*n*−2), where *u* is the minimum

value between *σ*(*n*) and *σ*(*n*−1). The second PE in the pair is

determined by subtracting the higher indexed one, specifically *e*2max = *pv* − *pσ*(*n*−2), where *v* = max(*σ*(*n*)*, σ*(*n* − 1)). By utilizing a generated 2D mapping, it is possible to alter the block pixels in this manner. The four approaches stated

above, namely PVO [46], IPVO [73], PVO-k [67], and k-pass PVO [23], can also be explained in the 2D space. Figure 14 displays the comparison of their two-dimensional mappings. In addition to the enhancements discussed in Section 3.2, several studies have been carried out to improve the pairwise PVO by optimizing 2D mapping [21, 130] and employing adaptive pairing [112, 133].

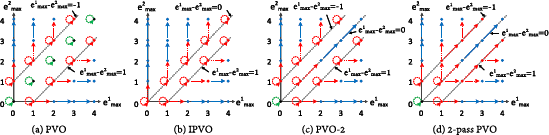


Fig. 13. Two-dimensional mapping comparison for (a) PVO, (b) IPVO, (c) PVO-k when k = 2, and (d) k-pass PVO when k = 2.

1. *MHM-based PVO:* It has been established through research that MHM is an adaptable embedding approach that proves to be quite successful. In light of this, it is possible to reach a better level of performance satisfaction by extending the MHM framework to incorporate PVO. It is possible to generate a one-dimensional histogram for *N* cover pixel blocks {*B*1*, B*2*, . . . , BN* } by collecting the high- est value from each block and storing it in a set. Here, *Bi* = (*pσ*(1)*, pσ*(2)*, . . . , pσ*(*n*)). The usual technique typically classifies the pixel *pσ*(*n*) based on its PE, namely *e*max. After looking at the histogram distribution, the best embedding values are found in an adaptive way. With the MHM-based PVO method, the 1D histogram that is made is then split into many smaller histograms based on measures of complexity. In Reference [64], the block complexity *ci* is defined as the second highest discrepancy in the *N* th block. In other words,

*ci* is equal to *pσ*(*n*−1) minus *pσ*(1). Based on the pair (*e*max*, ci*),

the highest numbers are put into different sub-histograms.

Next, the right expansion bin set is picked in an adaptable way by looking at the distributions of the sub-histograms that were collected.

1. *Discussion:* This section will provide a detailed analysis of three commonly used RDH techniques: MHM, pairwise PEE, and PVO. The purpose is to demonstrate their advantages over conventional methods.

TABLE IV

Performance metrics for different predictors on test images

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Image** | **Predictor** | **VAR** | **MSE** | **PEH Entropy** | **Mean** |
| Airplane | DP [95]  MED [94]  GAP [14]  RP [90]  PVO [46] | 93.95  105.71  31.87  41.84  30.46 | 120.67  134.52  43.81  56.24  8.50 | 4.59  4.67  4.12  4.19  1.27 | 5.31  5.51  3.53  3.81  2.84 |
| Lena | DP [95]  MED [94]  GAP [14]  RP [90]  PVO [46] | 87.05  72.71  25.38  32.86  23.78 | 127.57  106.16  40.58  50.86  7.22 | 5.05  4.91  4.38  4.49  1.39 | 6.48  5.89  3.96  4.31  2.99 |

With regard to the adaptive embedding strategy, the MHM- based techniques might be considered the logical progression of the strategy. The label of the unaffected class is applied to particular shifting pixels in earlier methods, such as pixel sort- ing and adaptive expansion-bins selection. This helps to reduce the amount of shifting that is unnecessary. These procedures can be viewed as sophisticated alterations. Nevertheless, these procedures are commonly carried out using global statistical characteristics, contradicting the more subtle alteration. An issue arises when a significant quantity of pixels that can be embedded are also classified as the unaltered category. In order to tackle this problem in MHM, the approach involves using clustered local statistical features to partition the pixels. This leads to the transformation of the initial histogram into a large number of sub-histograms at the same time. As a result, the embedding goes through a transformation on multiple levels. Finding a method that is more suitable for altering the local feature is something that is within the realm of possibility.

PEE is an alternative approach for making modifications at several levels. In order to maintain reversibility, the adjust- ments for the 1D mapping must follow specific restrictions, setting certain limitations on the more detailed alterations to a certain degree. In paired PEE, the use of 2D mapping allows for greater freedom in modifying the embedding in different directions. This enables the 2D mapping to finely alter the pixels, allowing for numerous layers of customization.

The reason why the pairwise PEE could attain greater per- formance is as follows. PVO utilizes the intrinsic attributes of the local visual information to make more precise predictions, rather than depending on contrived formulas. The two largest pixels in a given pixel block are naturally in close proximity. The classic predictors [14, 90, 94, 95] often select surrounding pixels from the local block for prediction. Within the same block, the adjoining pixels with the highest value are all lower than or equal to the second highest value. It enhances the accuracy of PVO prediction by aligning it more closely with the specific local context, resulting in a more plausible estimation.

The block size in PVO is specified as 3 × 3 here. The

statement suggests that PVO has the ability to produce PEs with reduced magnitudes, thereby leading to more precise pre- dictions. In a similar vein, the mean squared error, commonly known as MSE, between the pixels and the predictions that were produced using PVO is likewise the lowest.

When it comes to the histogram distribution, the PEH that was produced using PVO has a significantly lower entropy than conventional PEH. Based on this, it appears that its histogram is more distinct. When it comes to predicting the maximum of each block, The PVO algorithm is able to make more accurate predictions for the eight test photographs in a significant proportion (more than 83 percent).

1. THE STATES OF THE ART

Comparative analysis and analysis of the demonstrations of the sample works of PEE, MHM, pairwise PEE, and PVO are carried out in this part. It is common practice to evaluate the imperceptibility of RDH based on the patient’s visual state. PSNR is the measurement that is chosen the most frequently. Regarding the reversibility, the quality of the image that was recovered is taken into consideration. Generally speaking, RDH necessitates that the transmission channel be lossless in order to facilitate flawless recovery. Therefore, standard RDH algorithms don’t really handle resilience very well.

Next, the PSNR data for a given capacity will be used to compare performance. Ten thousand bits and twenty thousand bits, respectively, are encoded in the test picture of Lena. There is a comparison of the PSNRs and technological characteristics of the two different techniques, which can be found in Table

1. The data from the experiments were all taken from these previously published studies.

According to the findings shown in Table V, it is possible to achieve greater performance using MHM-based methods [22, 27, 49, 78, 105] and 2D mapping-based approaches [5, 15, 29,

63, 65, 112, 131, 133]. One of the most recent developments in the RDH community is the achievement of a PSNR of 61.70 dB by Reference [29], which is the highest of these. The higher concentration of PEH is provided by its CNN-based prediction. In addition, adaptive two-dimensional mapping is utilized in order to render the change process more efficient. One of the more recent studies for MHM is the scheme that was developed by Weng et al. [105], which undeniably enhances the performance of traditional MHM. Several his- tograms are constructed using this method, which makes use of the k-means clustering algorithm. When compared to the normal MHM [49], its PSNR gain = 0.49 decibels below the standard. For adaptive embedding, Fang et al. [15] created twenty possible two-dimensional mappings by applying the asymmetric predictor to MHM. The methodologies described in References [133] and [112] are examples of research that are considered to be excellent in the field of PVO. When compared to the typical PVO, the PSNR gains of the two techniques are determined to be 1.19 and 1.23 dB, respectively [46]. For the purpose of achieving a more even distribution, Zhang et al. [133] suggested that the pixel pairings be produced in a different way. An extensive search was carried out in order

TABLE V

Comparison of embedding capacities (in bits) achieved by various methods and techniques

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Technique** | **10000 bits** | **20000 bits** |
| Sachnev et al.  [90] | PEE with sorting | 58.21 | 55.04 |
| Xiao et al. [115] | Improved predictor having mapping model | 61.40 | 57.77 |
| Hu et al. [29] | CNN predictor with two-dimensional map-  ping | 61.70 | 57.96 |
| Abolfazl et al.  [41] | Asymmetric PEE | 60.86 | 57.26 |
| Kim et al. [40] | Asymmetric PEE | 59.92 | 56.67 |
| Li et al. [49] | MHM | 61.02 | 57.55 |
| Hou et al. [27] | MHM with DNN | 61.46 | 57.65 |
| Qi et al. [78] | MHM with Multi-parameter | 61.04 | 57.64 |
| Weng et al. [105] | MHM with Clustering | 61.51 | 57.65 |
| Ou et al. [68] | Pairwise PEE (two-dimensional mapping) | 59.75 | 56.21 |
| Zhang et al. [131] | Adaptive two-dimensional mapping | 60.93 | 57.38 |
| Chang et al. [5] | Adaptive two-dimensional mapping | 61.35 | 57.83 |
| Ou et al. [65] | Two-dimensional mapping with Adaptive  pairing | 60.77 | 57.25 |
| Fang et al. [15] | Two-dimensional mapping with Skewed  MHM | 61.58 | 58.02 |
| Li et al. [46] | PVO | 60.34 | 56.21 |
| Ou et al. [67] | PVO-k | 60.59 | 56.58 |
| He et al. [23] | k-pass PVO | 60.64 | 56.77 |
| Peng et al. [73] | IPVO | 60.49 | 56.57 |
| He et al. [22] | k-pass PVO with MHM | 61.39 | 57.75 |
| Ou et al. [63] | PVO with two-dimensional mapping | 60.91 | 56.69 |
| Xiang et al. [112] | PVO with two-dimensional mapping | 61.57 | 57.69 |
| Zhang et al. [133] | PVO with two-dimensional mapping | 61.53 | 57.74 |

to identify a suitable solution from the possibilities that were provided based on the statistical data.

The approach that was developed by Xiao et al. [115] have also shown good result (with a PSNR ¿ 61.30 dB for 10,000 bits). These techniques are in addition to the five strategies that were covered above. Researchers Chang et al. [5] conducted an exhaustive investigation into the potential of paired PEE. There are now additional options available for the two-dimensional mapping. Reference [22] investigates the possibility of im- provement by combining the MHM framework with the k-pass PVO technique. He and his colleagues improved the accuracy of the forecast by taking into account the absolute positional link that exists between two pixels. The numerous PEHs are generated in a dynamic manner, with the adaptive collection strategy serving as the basis. According to Xiao et al. [115], a new prediction method was proposed in order to mitigate the loss of precision that is brought about by round-off errors. After being confronted with a bigger payload (20,000 bits), it is very clear that most of the 8 ways that were previously discussed continue to beat others in performance. The six methods, which include the CNN-based RDH method [29], the pixel-residual PEH modification [115], the adaptive 2D mapping [5], the skewed MHM-based pairwise PEE method [15], the MHM-based k-pass PVO [22], and the location-based PVO [133], have the potential to get a value of PSNR that is more than 57.73 dB.

V. Future Work

The RDH in photographs has been the subject of thor- ough investigation at present. Using redundancy of photo is

something that current framework has a significant amount of expertise with. According to our point of view, the future RDH efforts might concentrate mostly on two different directions: the improvement of the schematic design and the investigation of empirical theory. There are still questions that need to be answered about the layout of the embedding phases and the optimization of parameters for the frameworks that are currently in use. It is possible to conduct additional research on the approaches that are currently in use, such as adaptive mapping, the construction of histograms, and complexity cal- culation, in order to investigate their potential. On the basis of the combination of a number of strategies, there are also an infinite number of effective variants that need to be researched. Moreover, there is a potential for substantial enhancement in RDH and the possibility of significant progress in schematic design through the application of novel techniques, such as deep learning algorithms, to address this issue. Following this, we will present four different potential avenues of inquiry for further research, described in Figures 16-19, respectively.

First, there is high-dimensional mapping. This course of action for further research organizes the modification mapping in higher dimensions, which makes it possible to incorporate the data in a more comprehensive manner. As the number of dimensions in the space increases, the mapping will display a greater number of variations, and the modification model can be altered to accommodate a particular carrier. The complete realization of the possibilities of adaptive embedding is a plausible outcome. In the same vein, there are a number of research that need to be investigated in order to develop a suit-

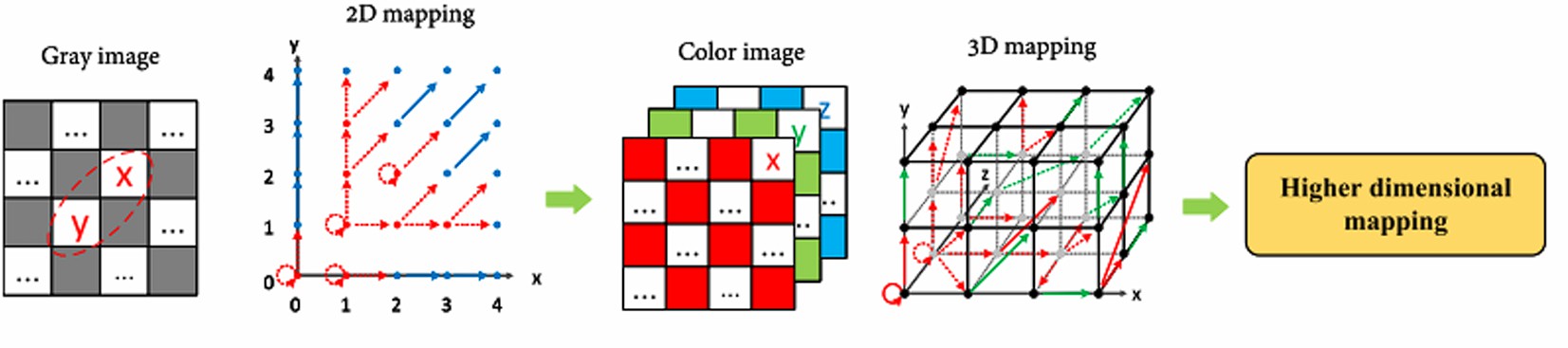


Fig. 14. High Dimensional Mapping

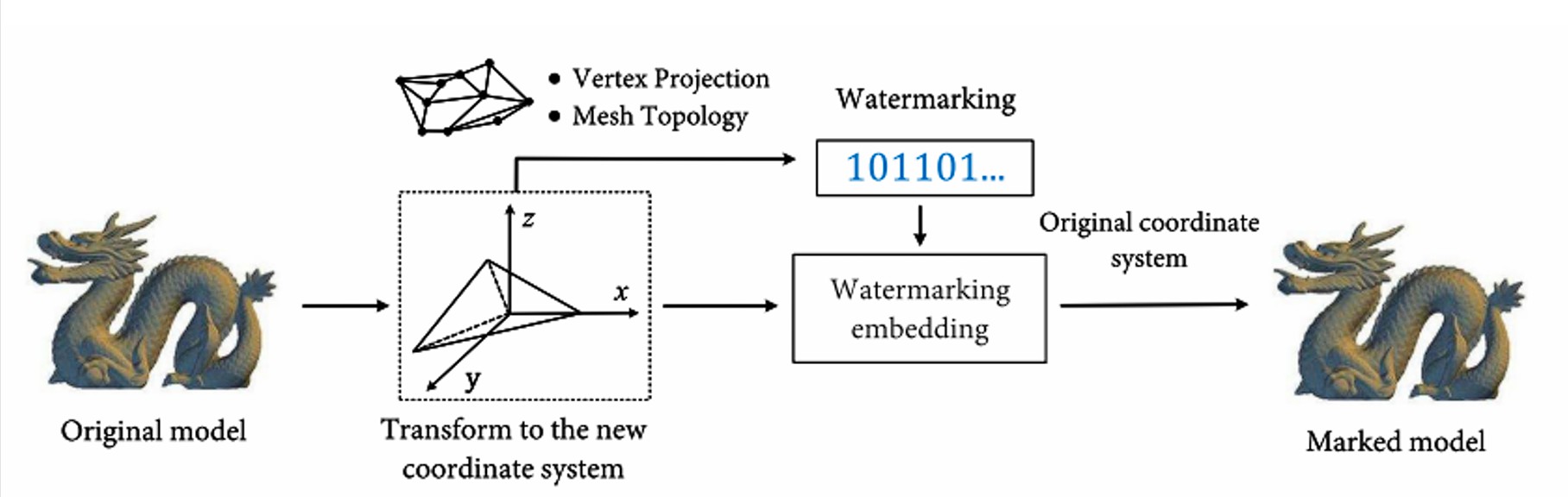
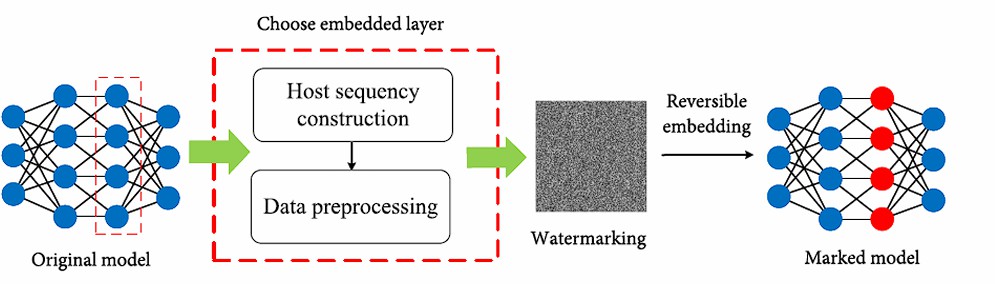


Fig. 15. RDH for the 3D Mesh Model

able optimization algorithm. The utilization of sophisticated computing approaches such as convolutional neural networks (CNN) and reinforcement learning may be beneficial in order to achieve a more favorable equilibrium between capacity, distortion, and time cost.

RDH is used for both two-dimensional vector graphics and three-dimensional mesh modeling. The proposed course of action for future development pertains to the utilization of multimedia data, which is frequently employed and necessi- tates novel embedding strategies and enhanced performance. In contemporary times, the utilization of 2D vector graphics and 3D mesh models has become prevalent in various applications such as engineering design and game modeling, among others. Every one of them has stringent standards for precision. Due to the fact that the RDH, 3D mesh models and in 2D vector graphics are different from the structures of normal raster images, new versions of these models should be implemented. For the content of the image, the RDH ought to take into consideration the unique positioning of the vertices and the mesh topology. Because floating-point values are more difficult to work with than integers, the data format presents a greater level of difficulty. A further thing to consider is that the 2D vector pictures and 3D mesh models that are deployed in practical applications are often subject to encryption. In order to achieve these objectives, it is essential to examine the RDH within the realm of encryption. A proposal for an embedding system for two-dimensional engineering graphics was made by Lin et al. [52]. The watermarking is constructed using their method, which involves applying the area nesting algorithm to the construction of numerous sub-regions. Making a corre- lation between the initial vertices and the subspaces that are generated is the first step in the process of doing the reversible embedding. After that, Peng et al. [75] developed a semi- fragile RDH method by further adapting this approach to 3D mesh models than they had originally done. The watermarking that was retrieved can be used to learn about the history of manipulation and the sort of manipulation that occurred.

Fig. 16. RDH for DNN

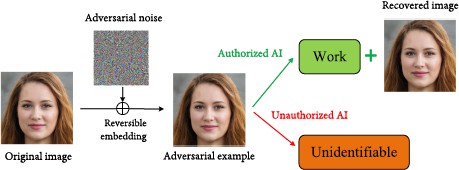


Fig. 17. Example for reversible adversarial

Third, RDH for the DNN. The purpose of this course of action for future development is to address the security flaws that have arisen in the era of artificial intelligence. The amount of mature models that are available on the internet nowadays makes it impossible to authenticate the owner of an artificial intelligence model. Additionally, the penalty for unlawful modification is quite minimal. If the owners wish to achieve considerable results, they might want to make sure that their copyright is protected while at the same time sharing their networks with other academics so that more people can access and develop upon their work. When using RDH, the watermarking could be included into the DNN in a reversible manner for the purpose of integrity authentication. It was still possible for the selected DNN to achieve its goal while maintaining its previous level of performance. For the purpose of preserving the network’s integrity, it is possible to recreate the network at the appropriate time without incurring any loss. It has been demonstrated by Guan et al. [17] that the idea of model watermarking is quite interesting.

The fourth case is a reversible adversarial example. Re- versible adversarial examples are a concept that was created not too long ago by Liu and colleagues [53]. Through the pro- cess of reversibly embedding the matching perturbation into the aimed photo, this technique for future research endeavors to construct adversarial examples. First, the watermarking is removed for the approved AI model in order to finish the authentication process. After that, the original image can be remade using the hostile example that was received. Finally, the AI model is able to proceed with its tasks by making use of the image that has been corrected. While other artificial intelligences might not function properly because of the hostile impact. The user’s privacy can be protected via the reversible adversarial example, which has the ability to fool unautho- rized artificial intelligence. When this scenario is taken into consideration, the aimed photo is seen as the data that needs to be protected, and the optimization aim shifts toward the

adversarial effect.

1. Summary

[14] M. Fallahpour, “Reversible image data hiding based on gradient adjusted prediction,” IEICE Electron. Express, vol. 5, no. 20, pp. 870–876, 2008.

Due to its better image recovery capabilities, RDH, a spe- cialized technology that ensures transmission security, stands out for its capacity to obtain extraordinary results in sensitive uses. The ability allowed RDH to accomplish remarkable results. In this paper, classic techniques and popular adaptive solutions for Reversibl Data Hiding in photos are investi- gated. The compression-based method, DE, IT, HS, PEE, MHM, pairwise PEE, and PVO are all included in these techniques with their respective acronyms. A presentation, analysis, and evaluation of the typical methodologies and approaches utilized by these eight fields of study are included in this article. Additionally, it discusses the benefits of the three most frequent techniques and evaluates the performance of the most advanced RDH algorithms no available. There is no way around the fact that RDH theory and embedding frameworks will continue to develop in the future, given the growing number of application scenarios and the increasing needs. Within the context of early motives, the paper proposes four prospective areas for further research, and it provides insights and comments on each of these potential avenues.

References

1. A. M. Alattar, “Reversible watermark using the difference expansion of a generalized integer transform,” IEEE Trans. Image Process., vol. 13, no. 8, pp. 1147–1156, 2004.
2. M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, “Lossless generalized-LSB data embedding,” IEEE Trans. Image Process., vol. 14, no. 2, pp. 253–266, 2005.
3. J. Chang, G. Zhu, H. Zhang, Y. Zhou, X. Luo, and L. Wu, “Reversible data hiding for color images based on adaptive 3D prediction-error expansion and double deep Q-network,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 8, pp. 5055–5067, 2022.
4. Q. Chang, X. Li, and Y. Zhao, “Reversible data hiding for color images based on adaptive three-dimensional histogram modification,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 9, pp. 5725–5735, 2022.
5. Q. Chang, X. Li, Y. Zhao, and R. Ni, “Adaptive pairwise prediction- error expansion and multiple histograms modification for reversible data hiding,” IEEE Trans. Circuits Syst. Video Technol., vol. 31, no. 12, pp. 4850–4863, 2021.
6. K. Chen and C.-C. Chang, “High-capacity reversible data hiding in encrypted images based on extended run-length coding and block-based MSB plane rearrangement,” J. Vis. Commun. Image Represent., vol. 58, pp. 334–344, 2019.
7. X. Chen, X. Sun, H. Sun, Z. Zhou, and J. Zhang, “Reversible water- marking method based on asymmetric-histogram shifting of prediction errors,” J. Syst. Softw., vol. 86, no. 10, pp. 2620–2626, 2013.
8. D. Coltuc, “Improved embedding for prediction-based reversible water- marking,” IEEE Trans. Inf. Forensics Secur., vol. 6, no. 3, pp. 873–882, 2011.
9. D. Coltuc and J.-M. Chassery, “Very fast watermarking by reversible contrast mapping,” IEEE Signal Process. Lett., vol. 14, no. 4, pp. 255–258, 2007.
10. I.-C. Dragoi and D. Coltuc, “Local-prediction-based difference expan- sion reversible watermarking,” IEEE Trans. Image Process., vol. 23, no. 4, pp. 1779–1790, 2014.
11. I.-C. Dragoi and D. Coltuc, “On local prediction based reversible water- marking,” IEEE Trans. Image Process., vol. 24, no. 4, pp. 1244–1246, 2015.
12. I.-C. Dragoi and D. Coltuc, “Adaptive Pairing Reversible Watermark- ing,” IEEE Trans. Image Process., vol. 25, no. 5, pp. 2420–2422, 2016.
13. Y. Du, Z. Yin, and X. Zhang, “High capacity lossless data hiding in JPEG bitstream based on general VLC mapping,” IEEE Trans. Dependable Secure Comput., pp. 1–1, 2020.
14. G. Fan, Z. Pan, Q. Zhou, X. Gao, and X. Zhang, “Multiple histogram based adaptive pairwise prediction-error modification for efficient re- versible image watermarking,” Inf. Sci. (Ny), vol. 581, pp. 515–535, 2021.
15. J. Fridrich, M. Goljan, and R. Du, “Invertible authentication,” in SPIE Proceedings, 2001.
16. X. Guan, H. Feng, W. Zhang, H. Zhou, J. Zhang, and N. Yu, “Reversible watermarking in deep convolutional neural networks for integrity authentication,” in Proceedings of the 28th ACM International Conference on Multimedia, 2020.
17. 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, vol. 98, pp. 370–380, 2014.
18. J. He, J. Chen, W. Luo, S. Tang, and J. Huang, “A novel high-capacity reversible data hiding scheme for encrypted JPEG bitstreams,” IEEE Trans. Circuits Syst. Video Technol., vol. 29, no. 12, pp. 3501–3515, 2019.
19. J. He, J. Chen, and S. Tang, “Reversible data hiding in JPEG images based on negative influence models,” IEEE Trans. Inf. Forensics Secur., vol. 15, pp. 2121–2133, 2020.
20. W. He and Z. Cai, “Reversible data hiding based on dual pairwise prediction-error expansion,” IEEE Trans. Image Process., vol. 30, pp. 5045–5055, 2021.
21. W. He, G. Xiong, and Y. Wang, “Reversible data hiding based on adaptive multiple histograms modification,” IEEE Trans. Inf. Forensics Secur., vol. 16, pp. 3000–3012, 2021.
22. W. He, K. Zhou, J. Cai, L. Wang, and G. Xiong, “Reversible data hiding using multi-pass pixel value ordering and prediction-error expansion,”

J. Vis. Commun. Image Represent., vol. 49, pp. 351–360, 2017.

1. W. Hong, G. Horng, C.-W. Shiu, T.-S. Chen, and Y.-C. Chen, “Re- versible steganographic method using complexity control and human visual system,” Comput. J., vol. 58, no. 10, pp. 2583–2594, 2015.
2. D. Hou, H. Wang, W. Zhang, and N. Yu, “Reversible data hiding in JPEG image based on DCT frequency and block selection,” Signal Processing, vol. 148, pp. 41–47, 2018.
3. D. Hou, W. Zhang, K. Chen, S.-J. Lin, and N. Yu, “Reversible data hiding in color image with grayscale invariance,” IEEE Trans. Circuits Syst. Video Technol., vol. 29, no. 2, pp. 363–374, 2019.
4. J. Hou, B. Ou, H. Tian, and Z. Qin, “Reversible data hiding based on multiple histograms modification and deep neural networks,” Signal Process. Image Commun., vol. 92, no. 116118, p. 116118, 2021.
5. R. Hu and S. Xiang, “CNN prediction based reversible data hiding,” IEEE Signal Process. Lett., vol. 28, pp. 464–468, 2021.
6. R. Hu and S. Xiang, “Reversible data hiding by using CNN prediction and adaptive embedding,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 44, no. 12, pp. 10196–10208, 2022.
7. Y. Hu, H.-K. Lee, and J. Li, “DE-based reversible data hiding with improved overflow location map,” IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 2, pp. 250–260, 2009.
8. Y. Hu, K. Wang, and Z.-M. Lu, “An improved VLC-based lossless data hiding scheme for JPEG images,” J. Syst. Softw., vol. 86, no. 8, pp. 2166–2173, 2013.
9. F. Huang, X. Qu, H. J. Kim, and J. Huang, “Reversible Data Hiding in JPEG Images,” IEEE Trans. Circuits Syst. Video Technol., vol. 26, no. 9, pp. 1610–1621, 2016.
10. C.-L. Jhong and H.-L. Wu, “Grayscale-invariant reversible data hiding based on multiple histograms modification,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 9, pp. 5888–5901, 2022.
11. Y. Jia, Z. Yin, X. Zhang, and Y. Luo, “Reversible data hiding based on reducing invalid shifting of pixels in histogram shifting,” Signal Processing, vol. 163, pp. 238–246, 2019.
12. R. Jiang, H. Zhou, W. Zhang, and N. Yu, “Reversible data hiding in encrypted three-dimensional mesh models,” IEEE Trans. Multimedia, vol. 20, no. 1, pp. 55–67, 2018.
13. S.-W. Jung, L. T. Ha, and S.-J. Ko, “A new histogram modification based reversible data hiding algorithm considering the human visual system,” IEEE Signal Process. Lett., vol. 18, no. 2, pp. 95–98, 2011.
14. Y. Ke, M. Zhang, X. Zhang, J. Liu, T. Su, and X. Yang, “A reversible data hiding scheme in encrypted domain for secret image sharing based on Chinese remainder theorem,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 4, pp. 2469–2481, 2022.
15. H. J. Kim, V. Sachnev, Y. Q. Shi, J. Nam, and H.-G. Choo, “A novel difference expansion transform for reversible data embedding,” IEEE Trans. Inf. Forensics Secur., vol. 3, no. 3, pp. 456–465, 2008.
16. S. Kim, R. Lussi, X. Qu, F. Huang, and H. J. Kim, “Reversible data hid- ing with automatic brightness preserving contrast enhancement,” IEEE Trans. Circuits Syst. Video Technol., vol. 29, no. 8, pp. 2271–2284, 2019.
17. S. Kim, X. Qu, V. Sachnev, and H. J. Kim, “Skewed histogram shifting for reversible data hiding using a pair of extreme predictions,” IEEE Trans. Circuits Syst. Video Technol., vol. 29, no. 11, pp. 3236–3246, 2019.
18. A. Kouhi and M. H. Sedaaghi, “Prediction error distribution with dynamic asymmetry for reversible data hiding,” Expert Syst. Appl., vol. 184, no. 115475, p. 115475, 2021.
19. S.-K. Lee, Y.-H. Suh, and Y.-S. Ho, “Reversible image authentication based on watermarking,” in 2006 IEEE International Conference on Multimedia and Expo, 2006.
20. J. Li and S. Xiang, “Audio-lossless robust watermarking against desynchronization attacks,” Signal Processing, vol. 198, no. 108561,

p. 108561, 2022.

1. W. Li, X. Li, R. Ni, and Y. Zhao, “PVO-based reversible data hiding using adaptive multiple histogram generation and modification,” Signal Process. Image Commun., vol. 99, no. 116405, p. 116405, 2021.
2. X. Li, B. Li, B. Yang, and T. Zeng, “General framework to histogram- shifting-based reversible data hiding,” IEEE Trans. Image Process., vol. 22, no. 6, pp. 2181–2191, 2013.
3. 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, vol. 93, no. 1, pp. 198–205, 2013.
4. X. Li, B. Yang, and T. Zeng, “Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection,” IEEE Trans. Image Process., vol. 20, no. 12, pp. 3524–3533, 2011.
5. X. Li, W. Zhang, X. Gui, and B. Yang, “A novel reversible data hiding scheme based on two-dimensional difference-histogram modification,” IEEE Trans. Inf. Forensics Secur., vol. 8, no. 7, pp. 1091–1100, 2013.
6. X. Li, W. Zhang, X. Gui, and B. Yang, “Efficient reversible data hiding based on multiple histograms modification,” IEEE Trans. Inf. Forensics Secur., vol. 10, no. 9, pp. 2016–2027, 2015.
7. X. Liang and S. Xiang, “Robust reversible audio watermarking based on high-order difference statistics,” Signal Processing, vol. 173, no. 107584, p. 107584, 2020.
8. C.-C. Lin, W.-L. Tai, and C.-C. Chang, “Multilevel reversible data hiding based on histogram modification of difference images,” Pattern Recognit., vol. 41, no. 12, pp. 3582–3591, 2008.
9. Z.-X. Lin, F. Peng, and M. Long, “A low-distortion reversible water- marking for 2D engineering graphics based on region nesting,” IEEE Trans. Inf. Forensics Secur., vol. 13, no. 9, pp. 2372–2382, 2018.
10. J. Liu, W. Zhang, K. Fukuchi, Y. Akimoto, and J. Sakuma, “Unautho- rized AI cannot recognize me: Reversible adversarial example,” Pattern Recognit., vol. 134, no. 109048, p. 109048, 2023.
11. M. Long, F. Peng, and H.-Y. Li, “Separable reversible data hiding and encryption for HEVC video,” J. Real Time Image Process., vol. 14, no. 1, pp. 171–182, 2018.
12. T. Luo, G. Jiang, M. Yu, C. Zhong, H. Xu, and Z. Pan, “Convolutional neural networks-based stereo image reversible data hiding method,” J. Vis. Commun. Image Represent., vol. 61, pp. 61–73, 2019.
13. W.-L. Lyu, L. Cheng, and Z. Yin, “High-capacity reversible data hiding in encrypted 3D mesh models based on multi-MSB prediction,” Signal Processing, vol. 201, no. 108686, p. 108686, 2022.
14. B. Ma and Y. Q. Shi, “A reversible data hiding scheme based on code division multiplexing,” IEEE Trans. Inf. Forensics Secur., vol. 11, no. 9, pp. 1914–1927, 2016.
15. B. Ma et al., “Adaptive error prediction method based on multiple linear regression for reversible data hiding,” J. Real Time Image Process., vol. 16, no. 4, pp. 821–834, 2019.
16. K. Ma, W. Zhang, X. Zhao, N. Yu, and F. Li, “Reversible data hiding in encrypted images by reserving room before encryption,” IEEE Trans. Inf. Forensics Secur., vol. 8, no. 3, pp. 553–562, 2013.
17. S. Ma, X. Li, M. Xiao, B. Ma, and Y. Zhao, “Fast expansion-bins- determination for multiple histograms modification based reversible data hiding,” IEEE Signal Process. Lett., vol. 29, pp. 662–666, 2022.
18. N. Mao, F. Chen, H. He, and Y. Yang, “Reversible data hiding based on adaptive IPVO and two-segment pairwise PEE,” Signal Processing, vol. 198, no. 108577, p. 108577, 2022.
19. 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, 2006.
20. B. Ou, X. Li, and J. Wang, “High-fidelity reversible data hiding based on pixel-value-ordering and pairwise prediction-error expansion,” J. Vis. Commun. Image Represent., vol. 39, pp. 12–23, 2016.
21. B. Ou, X. Li, and J. Wang, “Improved PVO-based reversible data hiding: A new implementation based on multiple histograms modi- fication,” J. Vis. Commun. Image Represent., vol. 38, pp. 328–339, 2016.
22. B. Ou, X. Li, W. Zhang, and Y. Zhao, “Improving pairwise PEE via hybrid-dimensional histogram generation and adaptive mapping selection,” IEEE Trans. Circuits Syst. Video Technol., vol. 29, no. 7,

pp. 2176–2190, 2019.

1. B. Ou, X. Li, Y. Zhao, and R. Ni, “Reversible data hiding based on PDE predictor,” J. Syst. Softw., vol. 86, no. 10, pp. 2700–2709, 2013.
2. B. Ou, X. Li, Y. Zhao, and R. Ni, “Reversible data hiding using invariant pixel-value-ordering and prediction-error expansion,” Signal Process. Image Commun., vol. 29, no. 7, pp. 760–772, 2014.
3. B. Ou, X. Li, Y. Zhao, R. Ni, and Y.-Q. Shi, “Pairwise prediction- error expansion for efficient reversible data hiding,” IEEE Trans. Image Process., vol. 22, no. 12, pp. 5010–5021, 2013.
4. B. Ou and Y. Zhao, “High capacity reversible data hiding based on multiple histograms modification,” IEEE Trans. Circuits Syst. Video Technol., vol. 30, no. 8, pp. 2329–2342, 2020.
5. Z. Pan, X. Gao, E. Gao, and G. Fan, “Adaptive complexity for pixel- value-ordering based reversible data hiding,” IEEE Signal Process. Lett., vol. 27, pp. 915–919, 2020.
6. F. Peng, W.-Y. Jiang, Y. Qi, Z.-X. Lin, and M. Long, “Separable robust reversible watermarking in encrypted 2D vector graphics,” IEEE Trans. Circuits Syst. Video Technol., vol. 30, no. 8, pp. 2391–2405, 2020.
7. F. Peng, X. Li, and B. Yang, “Adaptive reversible data hiding scheme based on integer transform,” Signal Processing, vol. 92, no. 1, pp. 54–62, 2012.
8. F. Peng, X. Li, and B. Yang, “Improved PVO-based reversible data hiding,” Digit. Signal Process., vol. 25, pp. 255–265, 2014.
9. F. Peng, T. Liao, and M. Long, “A semi-fragile reversible watermarking for authenticating 3D models in dual domains based on variable direc- tion double modulation,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 12, pp. 8394–8408, 2022.
10. F. Peng, B. Long, and M. Long, “A general region nesting-based semi- fragile reversible watermarking for authenticating 3D mesh models,” IEEE Trans. Circuits Syst. Video Technol., vol. 31, no. 11, pp. 4538–4553, 2021.
11. P. Puteaux and W. Puech, “A recursive reversible data hiding in encrypted images method with a very high payload,” IEEE Trans. Multimedia, vol. 23, pp. 636–650, 2021.
12. W. Qi, S. Guo, and W. Hu, “Generic reversible visible watermarking via regularized graph Fourier transform coding,” IEEE Trans. Image Process., vol. 31, pp. 691–705, 2022.
13. W. Qi, X. Li, T. Zhang, and Z. Guo, “Optimal reversible data hiding scheme based on multiple histograms modification,” IEEE Trans. Circuits Syst. Video Technol., vol. 30, no. 8, pp. 2300–2312, 2020.
14. Z. Qian, H. Xu, X. Luo, and X. Zhang, “New framework of reversible data hiding in encrypted JPEG bitstreams,” IEEE Trans. Circuits Syst. Video Technol., vol. 29, no. 2, pp. 351–362, 2019.
15. Z. Qian and X. Zhang, “Lossless data hiding in JPEG bitstream,” J. Syst. Softw., vol. 85, no. 2, pp. 309–313, 2012.
16. Z. Qian, X. Zhang, and S. Wang, “Reversible data hiding in en- crypted JPEG bitstream,” IEEE Trans. Multimedia, vol. 16, no. 5, pp. 1486–1491, 2014.
17. C. Qin, Z. He, H. Yao, F. Cao, and L. Gao, “Visible watermark removal scheme based on reversible data hiding and image inpainting,” Signal Process. Image Commun., vol. 60, pp. 160–172, 2018.
18. C. Qin, X. Qian, W. Hong, and X. Zhang, “An efficient coding scheme for reversible data hiding in encrypted image with redundancy transfer,” Inf. Sci. (Ny), vol. 487, pp. 176–192, 2019.
19. J. Qin and F. Huang, “Reversible data hiding based on multiple two- dimensional histograms modification,” IEEE Signal Process. Lett., vol. 26, no. 6, pp. 843–847, 2019.
20. Y. Qiu, Z. Qian, H. He, H. Tian, and X. Zhang, “Optimized lossless data hiding in JPEG bitstream and relay transfer-based extension,” IEEE Trans. Circuits Syst. Video Technol., vol. 31, no. 4, pp. 1380–1394, 2021.
21. Y. Qiu, Z. Qian, and L. Yu, “Adaptive reversible data hiding by extending the generalized integer transformation,” IEEE Signal Process. Lett., vol. 23, no. 1, pp. 130–134, 2016.
22. Y. Qiu, Q. Ying, Y. Yang, H. Zeng, S. Li, and Z. Qian, “High-capacity framework for reversible data hiding in encrypted image using pixel prediction and entropy encoding,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 9, pp. 5874–5887, 2022.
23. X. Qu and H. J. Kim, “Pixel-based pixel value ordering predictor for high-fidelity reversible data hiding,” Signal Processing, vol. 111, pp. 249–260, 2015.
24. R. M. Rad, K. Wong, and J.-M. Guo, “Reversible data hiding by adaptive group modification on histogram of prediction errors,” Signal Processing, vol. 125, pp. 315–328, 2016.
25. V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y. Q. Shi, “Reversible watermarking algorithm using sorting and prediction,” IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 7, pp. 989–999, 2009.
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, vol. 4, pp. 3210–3237, 2016.
27. Z. Su, Y. Ye, Q. Zhang, W. Li, and Y. Dai, “Robust 2D engineering CAD graphics hashing for joint topology and geometry authentication via covariance-based descriptors,” IEEE Trans. Inf. Forensics Secur., vol. 13, no. 4, pp. 1018–1030, 2018.
28. X. Tang, H. Wang, and Y. Chen, “Reversible data hiding based on a modified difference expansion for H.264/AVC video streams,” Multimed. Tools Appl., vol. 79, no. 39–40, pp. 28661–28674, 2020.
29. D. M. Thodi and J. J. Rodriguez, “Expansion embedding techniques for reversible watermarking,” IEEE Trans. Image Process., vol. 16, no. 3, pp. 721–730, 2007.
30. J. Tian, “Reversible data embedding using a difference expansion,” IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 8, pp. 890–896, 2003.
31. J. Wang, X. Chen, J. Ni, N. Mao, and Y. Shi, “Multiple histograms- based reversible data hiding: Framework and realization,” IEEE Trans. Circuits Syst. Video Technol., vol. 30, no. 8, pp. 2313–2328, 2020.
32. 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, vol. 159, pp. 193–203, 2019.
33. J. Wang, J. Ni, X. Zhang, and Y.-Q. Shi, “Rate and distortion opti- mization for reversible data hiding using multiple histogram shifting,” IEEE Trans. Cybern., pp. 1–12, 2016.
34. X. Wang, J. Ding, and Q. Pei, “A novel reversible image data hiding scheme based on pixel value ordering and dynamic pixel block partition,” Inf. Sci. (Ny), vol. 310, pp. 16–35, 2015.
35. X. Wang, X. Li, and Q. Pei, “Independent embedding domain based two-stage robust reversible watermarking,” IEEE Trans. Circuits Syst. Video Technol., vol. 30, no. 8, pp. 2406–2417, 2020.
36. X. Wang, X. Li, B. Yang, and Z. Guo, “Efficient generalized integer transform for reversible watermarking,” IEEE Signal Process. Lett., vol. 17, no. 6, pp. 567–570, 2010.
37. X. Wang, X. Wang, B. Ma, Q. Li, and Y.-Q. Shi, “High precision error prediction algorithm based on ridge regression predictor for reversible data hiding,” IEEE Signal Process. Lett., vol. 28, pp. 1125–1129, 2021.
38. S. Weng, T. Hou, T. Zhang, and J.-S. Pan, “Adaptive smoothness eval- uation and multiple asymmetric histogram modification for reversible data hiding,” J. Vis. Commun. Image Represent., vol. 90, no. 103732,

p. 103732, 2023.

1. S. Weng, Y. Shi, W. Hong, and Y. Yao, “Dynamic improved pixel value ordering reversible data hiding,” Inf. Sci. (Ny), vol. 489, pp. 136–154, 2019.
2. S. Weng, W. Tan, B. Ou, and J.-S. Pan, “Reversible data hiding method for multi-histogram point selection based on improved crisscross opti- mization algorithm,” Inf. Sci. (Ny), vol. 549, pp. 13–33, 2021.
3. S. Weng, G. Zhang, J.-S. Pan, and Z. Zhou, “Optimal PPVO-based reversible data hiding,” J. Vis. Commun. Image Represent., vol. 48,

pp. 317–328, 2017.

1. S. Weng, Y. Zhou, and T. Zhang, “Adaptive reversible data hiding for JPEG images with multiple two-dimensional histograms,” J. Vis. Commun. Image Represent., vol. 85, no. 103487, p. 103487, 2022.
2. H. Wu, X. Li, X. Luo, X. Zhang, and Y. Zhao, “General expansion- shifting model for reversible data hiding: Theoretical investigation and practical algorithm design,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 9, pp. 5989–6001, 2022.
3. H.-T. Wu, X. Cao, R. Jia, and Y.-M. Cheung, “Reversible data hiding with brightness preserving contrast enhancement by two-dimensional histogram modification,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 11, pp. 7605–7617, 2022.
4. H.-T. Wu, J.-L. Dugelay, and Y.-Q. Shi, “Reversible image data hiding with contrast enhancement,” IEEE Signal Process. Lett., vol. 22, no. 1, pp. 81–85, 2015.
5. Y. Wu, Y. Xiang, Y. Guo, J. Tang, and Z. Yin, “An improved reversible data hiding in encrypted images using parametric binary tree labeling,” IEEE Trans. Multimedia, vol. 22, no. 8, pp. 1929–1938, 2020.
6. S. Xiang and G. Ruan, “Efficient PVO-based reversible data hiding by selecting blocks with full-enclosing context,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 5, pp. 2868–2880, 2022.
7. M. Xiao, X. Li, B. Ma, X. Zhang, and Y. Zhao, “Efficient reversible data hiding for JPEG images with multiple histograms modifica- tion,” IEEE Trans. Circuits Syst. Video Technol., vol. 31, no. 7, pp. 2535–2546, 2021.
8. M. Xiao, X. Li, Y. Wang, Y. Zhao, and R. Ni, “Reversible data hiding based on pairwise embedding and optimal expansion path,” Signal Processing, vol. 158, pp. 210–218, 2019.
9. M. Xiao, X. Li, Y. Zhao, B. Ma, and G. Guo, “A novel reversible data hiding scheme based on pixel-residual histogram,” ACM Trans. Multimed. Comput. Commun. Appl., vol. 19, no. 1s, pp. 1–19, 2023.
10. L. Xiong, X. Han, C.-N. Yang, and Y.-Q. Shi, “Robust reversible watermarking in encrypted image with secure multi-party based on lightweight cryptography,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 1, pp. 75–91, 2022.
11. L. Xiong, X. Han, C.-N. Yang, and Z. Xia, “RDH-DES: Reversible Data Hiding over Distributed Encrypted-Image Servers based on secret sharing,” ACM Trans. Multimed. Comput. Commun. Appl., vol. 19, no. 1, pp. 1–19, 2023.
12. D. Xu and Y. Liu, “Reversible data hiding in H.264/AVC videos based on hybrid-dimensional histogram modification,” Multimed. Tools Appl., vol. 81, no. 20, pp. 29305–29319, 2022.
13. D. Xu and R. Wang, “Two-dimensional reversible data hiding-based approach for intra-frame error concealment in H.264/AVC,” Signal Process. Image Commun., vol. 47, pp. 369–379, 2016.
14. X. Yang and F. Huang, “New CNN-based predictor for reversible data hiding,” IEEE Signal Process. Lett., vol. 29, pp. 2627–2631, 2022.
15. Y. Yang, W. Zhang, D. Liang, and N. Yu, “Reversible data hiding in medical images with enhanced contrast in texture area,” Digit. Signal Process., vol. 52, pp. 13–24, 2016.
16. Y. Yang, T. Zou, G. Huang, and W. Zhang, “A high visual quality color image reversible data hiding scheme based on B-R-G embedding principle and CIEDE2000 assessment metric,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 4, pp. 1860–1874, 2022.
17. Y. Yao, W. Zhang, H. Wang, H. Zhou, and N. Yu, “Content-adaptive re- versible visible watermarking in encrypted images,” Signal Processing, vol. 164, pp. 386–401, 2019.
18. Y. Yao, W. Zhang, and N. Yu, “Inter-frame distortion drift analysis for reversible data hiding in encrypted H.264/AVC video bitstreams,” Signal Processing, vol. 128, pp. 531–545, 2016.
19. S. Yi and Y. Zhou, “Separable and reversible data hiding in encrypted images using parametric binary tree labeling,” IEEE Trans. Multimedia, vol. 21, no. 1, pp. 51–64, 2019.
20. Z. Yin, Y. Ji, and B. Luo, “Reversible data hiding in JPEG images with multi-objective optimization,” IEEE Trans. Circuits Syst. Video Technol., vol. 30, no. 8, pp. 2343–2352, 2020.
21. Z. Yin, Y. Peng, and Y. Xiang, “Reversible data hiding in encrypted images based on pixel prediction and bit-plane compression,” IEEE Trans. Dependable Secure Comput., pp. 1–1, 2020.
22. Z. Yin, Y. Xiang, and X. Zhang, “Reversible data hiding in encrypted images based on multi-MSB prediction and Huffman coding,” IEEE Trans. Multimedia, vol. 22, no. 4, pp. 874–884, 2020.
23. C. Yu, X. Zhang, G. Li, S. Zhan, and Z. Tang, “Reversible data hiding with adaptive difference recovery for encrypted images,” Inf. Sci. (Ny), vol. 584, pp. 89–110, 2022.
24. C. Yu, X. Zhang, D. Wang, and Z. Tang, “Reversible data hiding with pairwise PEE and 2D-PEH decomposition,” Signal Processing, vol. 196, no. 108527, p. 108527, 2022.
25. C. Zhang and B. Ou, “Reversible data hiding based on multiple adap- tive two-dimensional prediction-error histograms modification,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 7, pp. 4174–4187, 2022.
26. C. Zhang, B. Ou, X. Li, and J. Xiong, “Human visual system guided reversible data hiding based on multiple histograms modification,” Comput. J., vol. 66, no. 4, pp. 888–906, 2023.
27. T. Zhang, X. Li, W. Qi, and Z. Guo, “Location-based PVO and adaptive pairwise modification for efficient reversible data hiding,” IEEE Trans. Inf. Forensics Secur., vol. 15, pp. 2306–2319, 2020.
28. T. Zhang, T. Hou, S. Weng, F. Zou, H. Zhang, and C.-C. Chang, “Adaptive reversible data hiding with contrast enhancement based on multi-histogram modification,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 8, pp. 5041–5054, 2022.
29. W. Zhang, X. Hu, X. Li, and N. Yu, “Recursive histogram modification: Establishing equivalency between reversible data hiding and lossless data compression,” IEEE Trans. Image Process., vol. 22, no. 7, pp. 2775–2785, 2013.
30. X. Zhang, “Reversible data hiding in encrypted image,” IEEE Signal Process. Lett., vol. 18, no. 4, pp. 255–258, 2011.
31. X. Zhang, “Separable reversible data hiding in encrypted image,” IEEE Trans. Inf. Forensics Secur., vol. 7, no. 2, pp. 826–832, 2012.
32. X. Zhang, “Reversible data hiding with optimal value transfer,” IEEE Trans. Multimedia, vol. 15, no. 2, pp. 316–325, 2013.
33. X. Zhang, Y. Yao and N. Yu, ”Convolutional Neural Network-driven Optimal Prediction for Image Reversible Data Hiding,” 2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP), Tampere, Finland, 2021, pp. 1-6.
34. H. Zheng, C. Wang, J. Wang, and S. Xiang, “A new reversible water- marking scheme using the content-adaptive block size for prediction,” Signal Processing, vol. 164, pp. 74–83, 2019.