

## Research Article

# Obtuse Angle Prediction and Factor Evaluation for Image Reversible Data Hiding

Hongyin Xiang, Jinsha Yuan, and Sizu Hou

*Institute of Electrical and Electronic Engineering, North China Electric Power University, Baoding 071000, China*

Correspondence should be addressed to Hongyin Xiang; [leo\\_xianghy@126.com](mailto:leo_xianghy@126.com)

Received 1 September 2016; Revised 6 February 2017; Accepted 12 March 2017; Published 11 April 2017

Academic Editor: Huaguang Zhang

Copyright © 2017 Hongyin Xiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A pixel-based pixel-value-ordering (PPVO) has been used for reversible data hiding to generate large embedding capacity and high-fidelity marked images. The original PPVO invented an effective prediction strategy in pixel-by-pixel manner. This paper extends PPVO and proposes an obtuse angle prediction (OAP) scheme, in which each pixel is predicted by context pixels with better distribution. Moreover, for evaluating prediction power, a mathematical model is constructed and three factors, including the context vector dimension, the maximum prediction angle, and the current pixel location, are analyzed in detail. Experimental results declare that the proposed OAP approach can achieve higher PSNR values than PPVO and some other state-of-the-art methods, especially in the moderate and large payload sizes.

## 1. Introduction

Reversible data hiding (RDH) originated from a US patent by Barton in 1997 [1]. This technique can not only hide secret information into a cover medium, but also restore the host medium perfectly in the meantime decoding the embedded data. The superexcellent feature has aroused much attention in the whole research community in no time. For a long time, plenty of applications have been developed in all walks of life, such as secret message transmission, authentication, court evidence, military affairs, and intellectual property protection. What is more, the cover medium contains image, text, audio, and even video. This paper focuses on the image RDH technique.

In the past for more than two decades, RDH has been developed at full speed. So far, several application examples have been reported in numerous literatures, such as image authentication [1, 2], integrity check [3], trusted cloud data coloring [4], medical image processing [5, 6], and stereo image coding [7, 8]. All the aforementioned reality applications benefit from a number of outstanding methods, such as lossless compression, least significant bit (LSB) replacement, difference expansion (DE), integer wavelet transform (IWT), histogram shifting (HS), prediction error expansion (PEE),

and sorting. For clarity, four categories shall be designated and elaborated for a brief review in terms of major technical routes.

Firstly, lossless-compression-based methods [9–13] hide data bits into some space of the cover image by losslessly compressing a subset. Fridrich et al. [9] released space by compressing a bit-plane of cover image. This plane was determined as the lowest one to hide a 128-bit hash value. Goljan et al. [10] proposed the so-called R-S scheme. In this way, the cover image was classified into three different block categories, including regular, singular, and unusable, on the basis of the smoothness captured by a discrimination function. R-blocks and S-blocks prepared for data embedding using a flipping function. U-blocks were skipped. Xuan et al. [12] raised a high capacity RDH project on account of IWT, which utilized the coefficients of high frequency subbands to carry data bits. Furthermore, they compressed some selected middle bit-planes of IWT coefficients to make space. Celik et al. [13] put forward a compression method named generalized least significant bit (G-LSB). This technique took advantage of unaltered cover data portions as side information to improve the compression efficiency.

Secondly, DE-based schemes expanded some pixel parameter differences to make space for data hiding. Tian [14, 15]

presented the first DE method using pair-wise pixel differences in view of integer Haar wavelet transform [16]. Afterwards, Alattar [17] generalized DE from two pair-wise pixels to arbitrary sized pixel blocks, which improved the embedding rate nearly to  $n - 1$  bits per pixel (BPP). In view-point of computational complexity, Coltuc and Chassery [18] proposed a reversible contrast mapping scheme. Considering the magnitude of difference value, Weng et al. [19] adjusted the sum of pixel pairs and pair-wise differences for the invariability characteristic. Wang et al. [20] reformulated DE as a general integer transformation and extended this transformation. Coltuc [21] reduced the impact of location map for low distortion embedding.

Thirdly, HS-based approaches are the most successful for image RDH. Ni et al. [22] realized the fundamental HS program and van Leest et al. [23] expressed the similar thing. Later on, Ni et al.'s HS scheme was extensively investigated and many improved works came into being [24]. Fallahpour and Sedaaghi [25] applied HS for specified blocks instead of the whole cover image. This processing trick increased EC and reduced distortion. Lee et al. [26] depicted a novel difference histogram rather than conventional pixel intensity histogram. Experiments demonstrated that this Laplacian-like distribution, having a much higher peak point, performed better than aforementioned plane. Hence, pixel spatial correlation was well exploited. Xuan et al. [27] modified the histogram issued from IWT high frequency coefficients. This made distortion minimizing workable by selecting appropriate histogram pairs. Li et al. [28] fine investigated many previous HS schemes [29], treated them as special cases, and constructed a general framework for HS-based RDH. Via this way, one can derive a new RDH operation only needing an element setting.

Finally, PEE-based algorithms, deriving from DE technique, has attracted considerable attention and has developed to a large class SDH branch. Not the practical pixel differences, but the computed prediction errors utilized for expansion are the key issue. Thodi and Rodríguez [30] initiated PEE program and embedded large EC payload with low distortion. Incorporating with HS, PEE exploited much pixel correlation inherent and played well in distortion control. Hu et al. [31] designed an optimized payload-dependent overflow location map for looser pixel shifting constraint than conventional location maps. Therefore, it was constructed with two parts including embedding and shifting. This feature made great auxiliary bit stream compressibility for watermarking payload space. Combining prediction error modification with the above predictor, Hong et al. [32] revised the histogram of prediction errors for vacant position, obtaining EC enhancement efficiently, even several times higher than that of Ni et al. [22]. Sachnev et al. [33] invented a double-layer embedding method uniting HS and PEE. This promoted much accuracy for pixel prediction.

Recently, Li et al. [34] kneaded together sorting, PEE, and HS and raised a high-fidelity RDH method called pixel-value-ordering (PVO). PVO displayed impressive performance for moderate payload size. Soon, Peng et al. [35] and Ou et al. [36] extended this predictor and produced improved pixel-value-ordering (IPVO) and PVO-K techniques. All these predictors

obtained prediction values in a block-by-block manner. There are two pixels that might be predicted in each block and this constrains embedding capacity (EC) to some extent. Qu and Kim [37] created a novel pixel-based predictor in raster-scanning order, which predicted pixels in a pixel-by-pixel manner. In this method, the current pixel is previously determined and then predicted by its context pixels. The strategy has been verified as an effective prediction for high EC and peak signal-to-noise ratio (PSNR). However, PPVO only used parts of available referenced pixels within a right angle zone. In order to remedy this problem, a novel obtuse angle prediction (OAP) method will be proposed in this paper. Furthermore, prediction factors and its power weight shall be analyzed in detail.

In the rest sections, this paper shall be organized as follows. Section 2 presents four type classic predictors, containing dynamic prediction (DP), half-enclosing prediction (HEP), and full-enclosing prediction (FEP). PPVO is analyzed as a typical example of right angle prediction (RAP). Section 3 specifies the new proposed OAP type method and influencing factors in detail. Section 4 applies abundant experiments demonstrating the advantage of OAP art. Section 5 gives the conclusion.

## 2. Related Works

In this section, four prediction schemes based on PEE including dynamic prediction [34–36], half-enclosing prediction [30–32], full-enclosing prediction [33, 40], and PPVO prediction [37] are analyzed as follows.

*2.1. Dynamic Prediction.* Dynamic prediction obtains prediction values on the basis of context pixels with uncertain locations. Li et al. [34] provided these type prediction frames on behalf of PVO. Afterwards, Peng et al. [35] proposed IPVO and Ou et al. [36] put forward PVO-K as extensive methods. In general, PVO series methods predicted pixels in a block-by-block manner.

Firstly, the host image shall be cut into a temp processing image for some size appropriating for being divided into a great deal of nonoverlapped blocks with equal-size. As for Figure 1, a general  $n = r \times c$  sized block  $Bx$  is described, where  $i \in [2, r]$  and  $j \in [2, c]$ . For simplicity,  $Bx$  can be written as a vector  $(x_1, \dots, x_n)$  in the raster-scan order.

Secondly, the scanning pixel vector is sorted into a new one  $(x_{o(1)}, \dots, x_{o(n)})$  in an ascending order. The new ordered block is denoted by  $Bx_o$ . Thus a vector mapping is derived from  $(x_1, \dots, x_n)$  to  $(x_{o(1)}, \dots, x_{o(n)})$  with  $o(1) \leq o(2) \leq \dots \leq o(n-1) \leq o(n)$ . If  $1 \leq k_1, k_2 \leq n$ ,  $o(k_1) < o(k_2)$ , and  $x_{o(k_1)} = x_{o(k_2)}$ , then  $k_1 < k_2$ .

Then, prediction benchmark shall be established for the location of the current pixel or pixels. That is, there two pixels in a block at most can be determined as the current pixels. Note that, PVOs define different prediction error and its extension methods. Here, how to generate prediction error is mainly explained.

In conventional PVO method, the second largest pixel  $x_{o(n-1)}$  and the second smallest pixel  $x_{o(2)}$  are determined as

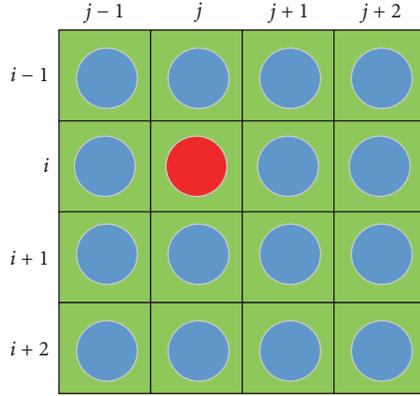


FIGURE 1: A general pixel block in PVOs. Red circle indicates the current pixel and blue circles are the referenced pixels.

the two current pixels, which are predicted by the largest pixel  $x_{o(n)}$  and the smallest pixel  $x_{o(1)}$ . Two prediction errors are obtained as

$$\begin{aligned} PE_{\max} &= x_{o(n)} - x_{o(n-1)}, \\ PE_{\min} &= x_{o(1)} - x_{o(2)}, \end{aligned} \quad (1)$$

where  $PE_{\max} \geq 0$  and  $PE_{\min} \leq 0$ .

In IPVO method, two current pixels are decided in the same way with PVO. But the two prediction errors are generated by

$$\begin{aligned} PE_{\max} &= x_u - x_v, \\ PE_{\min} &= x_s - x_t, \end{aligned} \quad (2)$$

where

$$\begin{aligned} u &= \min(o(n), o(n-1)), \\ v &= \max(o(n), o(n-1)), \\ s &= \min(o(1), o(2)), \\ t &= \max(o(1), o(2)). \end{aligned} \quad (3)$$

Thus, the corresponding couple prediction errors may take some values in  $(-\infty, +\infty)$ .

In PVO-K method, two prediction errors are determined to be much more complicated than PVO and IPVO. Three terms shall be assumed for the current pixels confirmation as follows.

- (i) The smallest pixel value is  $x_{\min} = x_{o(1)} = \dots = x_{o(k_1-1)}$  and the largest pixel value is  $x_{\max} = x_{o(k_2+1)} = \dots = x_{o(n)}$ ,  $2 \leq k_1 < k_2 \leq (n-1)$ .
- (ii) The second smallest value  $x_{o(k_1)}$  is no less than the second largest value  $x_{o(k_2)}$  and these two second minimum and maximum values are from different pixels.

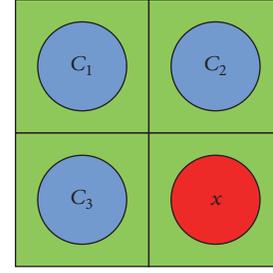


FIGURE 2: Context pixels of a  $2 \times 2$  block.

Thereout, the second smallest pixel  $x_{o(k_1)}$  and the second largest pixel  $x_{o(k_2)}$  are considered to be the current pixels. The two prediction errors are decided by

$$\begin{aligned} PE_{\max} &= x_{\max} - x_{o(k_2)}, \\ PE_{\min} &= x_{\min} - x_{o(k_1)}, \end{aligned} \quad (4)$$

where  $PE_{\max} \geq 0$ ,  $PE_{\min} \leq 0$ .

PVO, IPVO, and PVO-K have the common feature for prediction confirmation. The current pixels must be determined only after pixel sorting. Referenced pixels cannot be identified independently before sorting and context pixels are also incapable. The current pixels are dynamic, the context pixels are dynamic, and so, this type of prediction is called dynamic prediction.

**2.2. Half-Enclosing Prediction.** Thodi and Rodríguez [30] proposed the first PEE-based method, in which histogram shifting was combined to embed secret data. It invented a simple constructor well exploiting the correlation inherent of the context pixels. Figure 2 described a  $2 \times 2$  pixel block. This predictor appointed the red bottom-right pixel to be the current one  $x$  and the other three blue ones ( $C_1, C_2, C_3$ ) the context pixels. Thus, only pixels before the current pixel were probable to be used for prediction. Here, this is categorized as half-enclosing prediction.

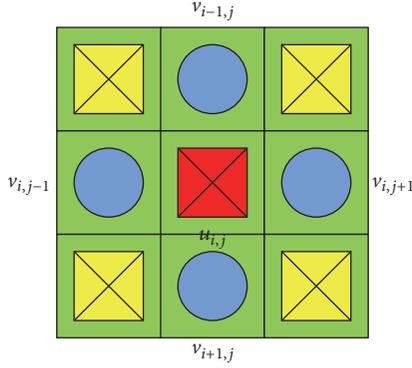
Some feature elements were believed in the three context pixels nearby the current pixel. The predictor output was defined by

$$x_T = \begin{cases} \max(C_2, C_3), & \text{if } C_1 \leq \min(C_2, C_3) \\ \min(C_2, C_3), & \text{if } C_1 \geq \max(C_2, C_3) \\ C_2 + C_3 - C_1, & \text{otherwise.} \end{cases} \quad (5)$$

The prediction value of  $x$  was marked by  $x_p$

$$x_p = 2 \cdot \left\lfloor \frac{x_T}{2} \right\rfloor. \quad (6)$$

**2.3. Full-Enclosing Prediction.** A rhombus prediction scheme was presented by Sachnev et al. [33]. This predictor predicted pixels with high accuracy. The core idea was to launch all the four closely surrounding pixels to compute the prediction value. Figure 3 depicted such a case. Two pixel sets are defined

FIGURE 3: Context pixels of  $u_{i,j}$ .

as the cross set and the dot set, respectively. In a single-track procedure for embedding or restoring, if the cross set was assigned for current pixels, then the dot set should be the context pixels.

As shown in Figure 3, the current cross pixel is  $u_{i,j}$ , and the four neighboring pixels are  $v_{i,j-1}$ ,  $v_{i+1,j}$ ,  $v_{i,j+1}$ , and  $v_{i-1,j}$ . The predicted center pixel value is computed as follows:

$$u_{i,j}^p = \left\lfloor \frac{v_{i,j-1} + v_{i+1,j} + v_{i,j+1} + v_{i-1,j}}{4} \right\rfloor. \quad (7)$$

The prediction error is computed as

$$d_{i,j} = u_{i,j} - u_{i,j}^p. \quad (8)$$

The expanded prediction error can be applied for a data bit hiding like the difference expansion scheme by Tian [14],

$$D_{i,j} = 2d_{i,j} + b, \quad (9)$$

where  $b$  is the to-be-hidden data bit.

Besides this predictor, histogram shift technique was also utilized in this paper to improve embedding performance in terms of low distortion. Two thresholds, including the negative one  $T_n$  and the positive one  $T_p$ , were introduced to separate the expandable set and the shifting set. The predicted errors belonging to  $[T_n, T_p]$  should be expanded and the outer region ones should be shifted towards right or left. Thus, the complete error expansion algorithm is expressed by

$$D_{i,j} = \begin{cases} 2d_{i,j} + b, & \text{if } d_{i,j} \in [T_n, T_p] \\ d_{i,j} + T_p + 1, & \text{if } d_{i,j} > T_p, T_p \geq 0 \\ d_{i,j} + T_n, & \text{if } d_{i,j} < T_n, T_n < 0. \end{cases} \quad (10)$$

**2.4. PPVO Prediction.** Qu and Kim [37] presented PPVO recently. This method predicted pixel in a pixel-by-pixel manner. Each pixel except a very tiny one had a chance to be predicted, and thus, the prediction probability was larger than those block-mannered prediction schemes.

All the pixels shall be scanned in the sequential raster-scanning order from left to right and then from top to bottom.

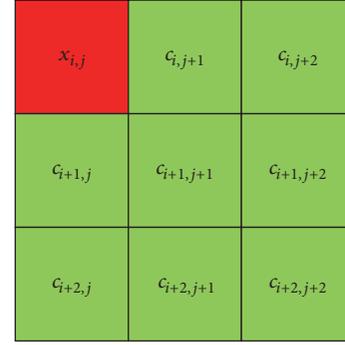


FIGURE 4: Context pixels for PPVO.

The top-left pixel is prerequisite to be the current pixel and the others in some sized block are responsible for computing its prediction value. In addition, block size is fixed as  $3 \times 3$ , for example. Figure 4 draws the pixel distribution of PPVO.

### 3. Proposed Method

**3.1. Obtuse Angle Prediction (OAP).** Pixel-based scheme predicts pixels in following some principles. In general, context pixels are employed for predicting the current pixel. Considering reversibility, in raster-scanning order, the to-be-predicted pixel must locate before the context pixels. Figure 5 schedules the pixel distributions in an angle zone. Red circle indicates the current pixel and blue circles denote context pixels. For clarity, pixels shall be processed from left to right and then from top to bottom. The minimum straight-line pixel distance is identified with  $u > 0$ . From the viewpoint of location relations, three prediction structures are defined as follows.

**Definition 1** (prediction angle (PA)). PA refers to the angle developed by the border sides of the current pixel and context pixels. In Figure 5, PA was filled with green arcs.

**Definition 2** (right angle prediction (RAP)). RAP stands for pixel distribution with angle cased PA, in which the current pixel locates on the top-left position and context pixels locate within PA region. As shown in Figure 5(a), in the  $xOy$  coordinate system, the current pixel is  $O(0, 0)$  and two context pixels  $p_m^1(u, 0)$  and  $p_m^2(0, u)$  situate on the boundary lines severally,  $\angle p_m^1 O p_m^2 = 90^\circ$ . Totally, all the context pixels site in the same quadrant or on the  $x$  positive axis and  $y$  negative axis. Conventional PPVO belongs to RAP.

**Definition 3** (acute angle prediction (AAP)). AAP refers to the type of pixel distribution prediction while PA is acute angle, in which the current pixel lies on the top-left location and context pixels locate within PA area. As shown in Figure 5(b), in the  $xOy$  coordinate system, the current pixel is  $O(0, 0)$  and two context pixels  $p_m^1(-u, u)$  and  $p_m^2(0, u)$  situate on the boundary lines severally,  $\angle p_m^1 O p_m^2 < 90^\circ$ . Totally, all the context pixels site in the same quadrant or on the only  $y$  negative axis.

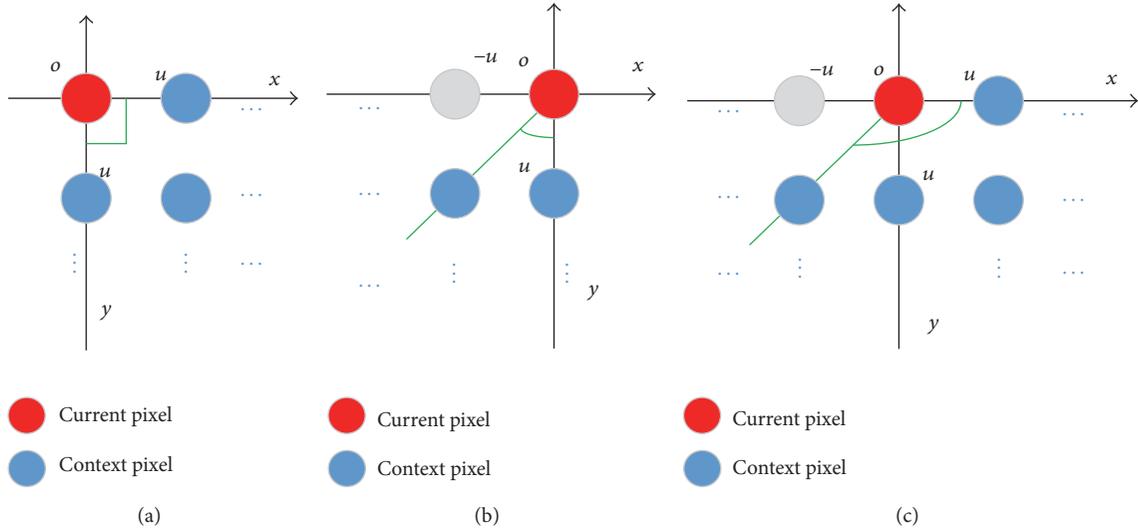


FIGURE 5: The pixel distribution of angle predictions. (a) Right angle prediction (RAP). (b) Acute angle prediction (AAP). (c) Obtuse angle prediction (OAP).

*Definition 4* (obtuse angle prediction (OAP)). OAP refers to the type of pixel distribution prediction while PA is obtuse angle, in which the current pixel lies on the top-left location and context pixels locate within PA area. As shown in Figure 5(c), in the  $xOy$  coordinate system, the current pixel is  $O(0, 0)$  and two context pixels  $p_m^1(u, 0)$  and  $p_m^2(-u, u)$  situate on the boundary lines severally,  $90^\circ < \angle p_m^1 O p_m^2 < 180^\circ$ . Totally, all the context pixels site across two different quadrants or on the  $x$  positive axis and  $y$  negative axis. For convenience, if the aimed pixel locates close behind the top-left position, it is denoted as OAP-I. Along with the next location of the current pixel away from the top-left position, OAP-II, OAP-III, or other multi-OAP are marked. If the number of the first row pixels is signed with  $n_r^1$ , the maximum digit shall be  $n_r^1 - 2$  and the according mark is OAP- $(n_r^1 - 2)$ .

It is easy to see that the difference between the three type predictions is the current pixel's location, causing different context pixel numbers and distributions. In view of quantity, pixels in the first row, except for the aimed one, are all context pixels in RAP. There is no context pixel at all in the first row in AAP. Nevertheless, there is more than one context pixel in the first row in OAP. Hence, valid context pixel quantities of the above three predictions satisfy this inequation  $n_{RAP} > n_{OAP} > n_{AAP}$ . In terms of prediction angle, consulting the aforementioned pixel distributions of RAP, AAP, and OAP, the layout advantages can be expressed by  $a_{OAP} > a_{RAP} > a_{AAP}$ . Overall, each of OAP and RAP has different strengths and AAP behaves the worst.

### 3.2. RDH Algorithm for OAP

*3.2.1. Data Embedding Procedure.* Figure 6 summarizes the embedding procedure with the following four steps.

*Step 1* (image preprocess). There are two main points. One is to modify marginal pixels probably to overflow the gray value section. For an eight-bit gray-scale image, the intensity range is  $[0, 255]$ . On account of the maximum pixel modification being one, 0 and 255 should be adapted to be 1 and 254 before prediction. The other one is to prepare some parameters and data for the follow-up procedure. Location map is used to stamp the modified pixel, which contributes to the reversible workflow. For less size, it should usually be compressed in a lossless compressing way.

*Step 2* (testing model). OAP has optional schemes when block size is large enough. With a certain model, context pixel distribution and neighboring regional noise level of the current pixel have to be computed according to EC demand. Generally, iterative computation utilizes unit step size.

*Step 3* (determining parameters). A successful embedding happens only when EC is larger than payload size. The minimum noise level threshold (NLT) and the context pixel number (CN) are the most important parameters. Auxiliary information (AI) makes for a reversible scheme. The AI sized headmost pixels extracted the least signal bits (LSB) for AI storage. Thus, LSB, compressed location map, and payload must be combined together to form the to-be-embedded data sequence (DS). Assume that the cover image has pixel size  $W \times H$ . AI contents are listed as follows:

- (i) The length of compressed location map ( $\log_2^{W \times H}$  bits).
- (ii) The noise level threshold NLT (8 bits).
- (iii) The context pixel number CN (8 bits).
- (iv) The location of the last embedded pixel ( $\log_2^{W \times H}$  bits).

*Step 4* (data embedding). Context pixels make up a vector  $C$ . The maximum value and minimum value of this vector

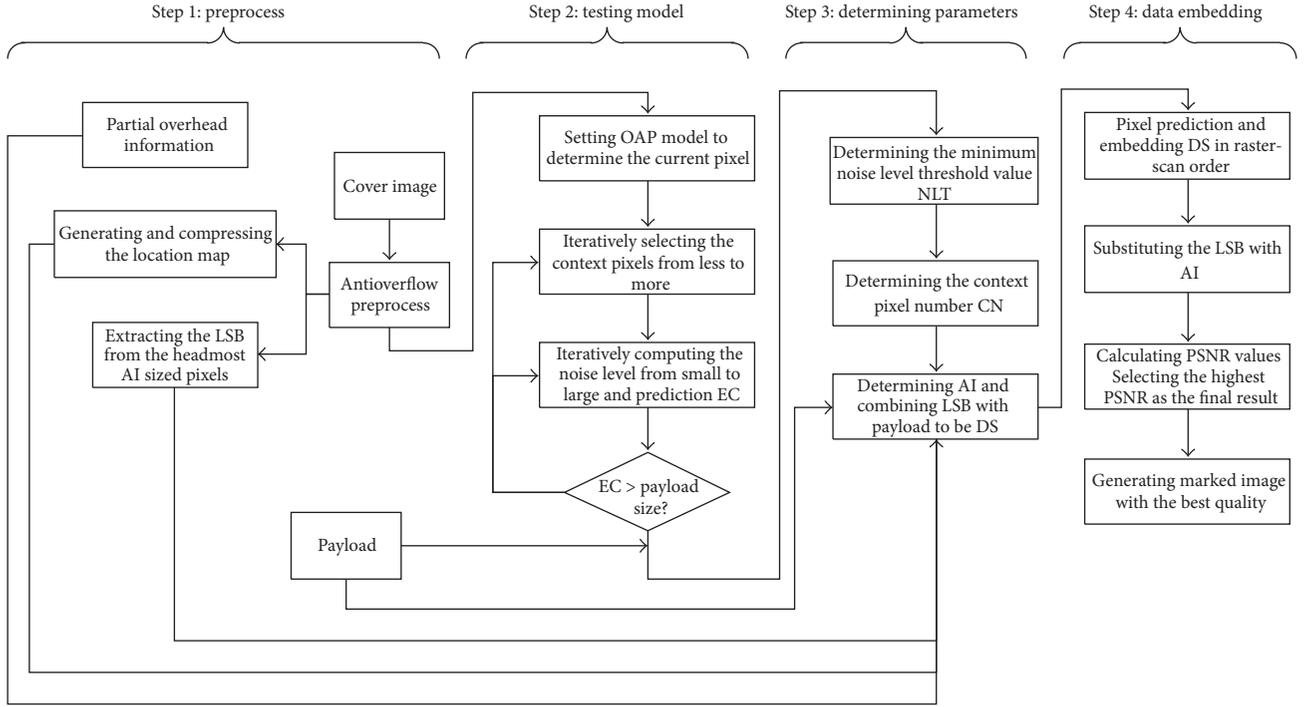


FIGURE 6: The workflow of embedding procedure.

are denoted as  $\max(C)$  and  $\min(C)$  separately. And so, the current pixel is predicted by

$$x_{i,j}^p = \begin{cases} \min(C), & \text{if } \min(C) \neq \max(C), x_{i,j} \leq \min(C) \\ \max(C), & \text{if } \min(C) \neq \max(C), x_{i,j} \geq \max(C) \\ 254, & \text{if } \min(C) = \max(C) = 254, x_{i,j} = 254 \\ VC, & \text{if } \min(C) = \max(C) = VC, VC < 254, x_{i,j} \leq VC. \end{cases} \quad (11)$$

The output prediction error is denoted as

$$pe = x_{i,j} - x_{i,j}^p. \quad (12)$$

A data bit shall be hidden into this current pixel according to the following expanded error

$$PE = \begin{cases} pe - b, & \text{if } \min(C) \neq \max(C), x_{i,j} = \min(C) \\ pe - 1, & \text{if } \min(C) \neq \max(C), x_{i,j} < \min(C) \\ pe + b, & \text{if } \min(C) \neq \max(C), x_{i,j} = \max(C) \\ pe + 1, & \text{if } \min(C) \neq \max(C), x_{i,j} > \max(C) \\ pe + b, & \text{if } \min(C) = \max(C) = 254, x_{i,j} = 254 \\ pe - b, & \text{if } \min(C) = \max(C) = VC, VC < 254, x_{i,j} = VC \\ pe - 1, & \text{if } \min(C) = \max(C) = VC, VC < 254, x_{i,j} < VC, \end{cases} \quad (13)$$

where  $b$  is the to-be-hidden data bit.

Pixels becomes the following values:

$$x_{i,j}^E = x_{i,j}^p + PE. \quad (14)$$

Embed all the DS bits into the cover image according to (11)–(14). Substitute the LSB location with AI to get the marked image. Calculate PSNR values and select the

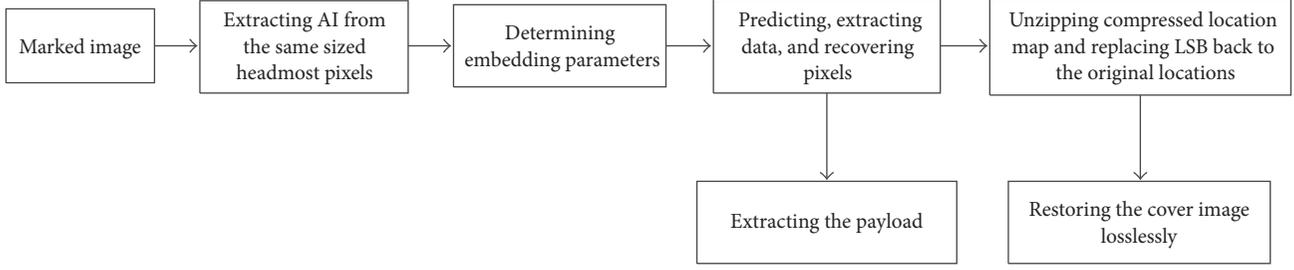


FIGURE 7: The workflow of data extraction and image restoration procedure.

highest one as the final result. Of course, this means the corresponding best embedding quality.

**3.2.2. Data Extraction and Image Restoration Procedure.** Figure 7 describes the data extraction and image restoration procedure with the following four steps.

*Step 1* (extracting AI for embedding parameters). AI has fixed length. Extract the LSB of the AI sized headmost pixels to AI. Thus, four parameters including the length of compressed location map, NLT, CN, and the location of the last embedded pixel shall be obtained.

*Step 2* (extracting DS). Predict pixels with the same method in (11) from the first embedding location to the last embedding pixel. Suppose  $\bar{x}_{i,j}$  is the pixel in marked image. In order to distinguish with the embedding procedure, prediction value is denoted as  $\bar{x}_{i,j}^p$ .

The output prediction error is denoted as

$$pe' = \bar{x}_{i,j} - \bar{x}_{i,j}^p. \quad (15)$$

A data bit shall be extracted from the current pixel according to the following expanded error:

$$PE' = \begin{cases} pe' + b, & \text{if } \min(C) \neq \max(C), \bar{x}_{i,j} = \min(C) \text{ or } \bar{x}_{i,j} = \min(C) - 1 \\ pe' + 1, & \text{if } \min(C) \neq \max(C), \bar{x}_{i,j} < \min(C) - 1 \\ pe' - b, & \text{if } \min(C) \neq \max(C), \bar{x}_{i,j} = \max(C) \text{ or } \bar{x}_{i,j} = \max(C) + 1 \\ pe' - 1, & \text{if } \min(C) \neq \max(C), \bar{x}_{i,j} < \max(C) + 1 \\ pe' - b, & \text{if } \min(C) = \max(C) = 254, \bar{x}_{i,j} = 254 \text{ or } \bar{x}_{i,j} = 255 \\ pe' + b, & \text{if } \min(C) = \max(C) = VC, VC < 254, \bar{x}_{i,j} = VC \text{ or } \bar{x}_{i,j} = VC - 1 \\ pe' + 1, & \text{if } \min(C) = \max(C) = VC, VC < 254, \bar{x}_{i,j} < VC - 1, \end{cases} \quad (16)$$

where  $b$  is the to-be-restored data bit.

Pixels become the following values:

$$x'_{i,j} = \bar{x}_{i,j} + PE'. \quad (17)$$

If pixel  $x'_{i,j}$  is equal to the original  $x_{i,j}$ , a reversible data hiding scheme is acquired.

*Step 3* (unzipping compressed location map). Extracted data includes location map and it must be unzipped with the same lossless decoding unzip method.

*Step 4* (restoring the cover image). According to the unzipped location map, recover the modified pixels back to original values. Hence, blind extraction and image restoration are accomplished.

**3.3. Prediction Factors.** For balancing embedding capacity and image fidelity, it is advisable that the pixel block should

be too small or too large. From literatures [34–37, 40], block size is proposed between  $2 \times 2$  and  $5 \times 5$ . By Definition 2, the minimum size dissatisfies OAP condition and the maximum size block may be a little complicated. Therefore, a  $4 \times 4$  sized block shall be taken as an example for further analysis.

OAP-I and OAP-II are both depicted in Figure 8. Only pixels behind the current pixel are involved in context vector  $C$  and the number is equal to its dimension  $D$ . These two parameters of RAP, OAP-I, and OAP-II are denoted as

$$\begin{aligned} \overrightarrow{C}_{\text{RAP}} &= \{c_{11}, c_{12}, c_{13}, c_{21}, c_{22}, c_{23}, c_{24}, c_{25}, c_{31}, c_{32}, c_{33}, c_{34}, c_{35}, \\ &\quad c_{36}, c_{37}\}, \quad D_{\overrightarrow{C}_{\text{RAP}}} = 15, \\ \overrightarrow{C}_{\text{OAP-I}} &= \{c_{11}, c_{12}, c_{13}, c_{14}, c_{21}, c_{22}, c_{23}, c_{24}, c_{25}, c_{26}, c_{31}, c_{32}, c_{33}, \\ &\quad c_{34}\}, \quad D_{\overrightarrow{C}_{\text{OAP-I}}} = 14, \\ \overrightarrow{C}_{\text{OAP-II}} &= \{c_{11}, c_{12}, c_{13}, c_{14}, c_{21}, c_{22}, c_{23}, c_{24}, c_{25}, c_{31}, c_{32}, c_{33}, \\ &\quad c_{34}\}, \quad D_{\overrightarrow{C}_{\text{OAP-II}}} = 13. \end{aligned} \quad (18)$$

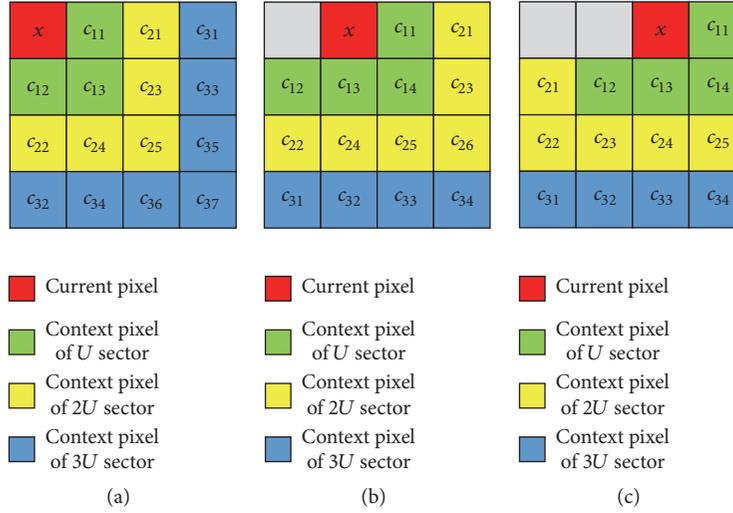


FIGURE 8: Context pixels in different prediction sector radii. (a) RAP; (b) OAP-I; (c) OAP-II.

Thus,  $D_{C_{OAP-II}}^{\overrightarrow{\quad}} < D_{C_{OAP-I}}^{\overrightarrow{\quad}} < D_{C_{RAP}}^{\overrightarrow{\quad}}$ ; the quantity advantage of context pixels decreased systematically from RAP and OAP-I to OAP-II.

Observing from Figures 5 and 8, the current pixel and the two side boundary lines form a sector, which can be named by prediction sector. Sector angle and radii feature the context pixel distribution. For further comparison of prediction capability, sector radii need to be defined explicitly in Definition 5.

**Definition 5** (prediction sector radii (PSR)). PSR refers to the approaching degree of the current pixel and the active context pixels. It is marked as  $R_s$  and the unit is  $U$ . The amount of  $U$  is different from previous pixel unit distance  $u$  and it actually represents a range of distance. Here,  $|U| \geq u$ .

In blocks of Figure 8, red is the current pixel. The close neighboring green pixels are called  $U$  radii context pixels  $R_s^U$ . The next closing yellow pixels are  $2U$  radii context pixels  $R_s^{2U}$ . And the external closing pixels are  $3U$  radii context pixels  $R_s^{3U}$ . From this, the context vectors of RAP in different sector radii can be obtained as

$$\begin{aligned} \overrightarrow{C_{RAP}^U} &= \{c_{11}, c_{12}, c_{13}\}, \quad D_{C_{RAP}^U}^{\overrightarrow{\quad}} = 3, \\ \overrightarrow{C_{RAP}^{2U}} &= \{c_{21}, c_{22}, c_{23}, c_{24}, c_{25}\}, \quad D_{C_{RAP}^{2U}}^{\overrightarrow{\quad}} = 5, \\ \overrightarrow{C_{RAP}^{3U}} &= \{c_{31}, c_{32}, c_{33}, c_{34}, c_{35}, c_{36}, c_{37}\}, \quad D_{C_{RAP}^{3U}}^{\overrightarrow{\quad}} = 7. \end{aligned} \quad (19)$$

The context vectors of OAP-I in different sector radius are

$$\begin{aligned} \overrightarrow{C_{OAP-I}^U} &= \{c_{11}, c_{12}, c_{13}, c_{14}\}, \quad D_{C_{OAP-I}^U}^{\overrightarrow{\quad}} = 4, \\ \overrightarrow{C_{OAP-I}^{2U}} &= \{c_{21}, c_{22}, c_{23}, c_{24}, c_{25}, c_{26}\}, \quad D_{C_{OAP-I}^{2U}}^{\overrightarrow{\quad}} = 6, \\ \overrightarrow{C_{OAP-I}^{3U}} &= \{c_{31}, c_{32}, c_{33}, c_{34}\}, \quad D_{C_{OAP-I}^{3U}}^{\overrightarrow{\quad}} = 4. \end{aligned} \quad (20)$$

The context vectors of OAP-II in different sector radius are

$$\begin{aligned} \overrightarrow{C_{OAP-II}^U} &= \{c_{11}, c_{12}, c_{13}, c_{14}\}, \quad D_{C_{OAP-II}^U}^{\overrightarrow{\quad}} = 4, \\ \overrightarrow{C_{OAP-II}^{2U}} &= \{c_{21}, c_{22}, c_{23}, c_{24}, c_{25}\}, \quad D_{C_{OAP-II}^{2U}}^{\overrightarrow{\quad}} = 5, \\ \overrightarrow{C_{OAP-II}^{3U}} &= \{c_{31}, c_{32}, c_{33}, c_{34}\}, \quad D_{C_{OAP-II}^{3U}}^{\overrightarrow{\quad}} = 4. \end{aligned} \quad (21)$$

Predicting image Lena following (11) works out the active context vector dimensions within different sector radius. If  $R_s^p = R_s^u$ , there is

$$\begin{aligned} D_{RAP}^p &= D_{C_{RAP}^U}^{\overrightarrow{\quad}} = 3, \\ D_{OAP-I}^p &= D_{C_{OAP-I}^U}^{\overrightarrow{\quad}} = 4, \\ D_{OAP-II}^p &= D_{C_{OAP-II}^U}^{\overrightarrow{\quad}} = 4. \end{aligned} \quad (22)$$

If  $R_s^p = R_s^u + R_s^{2u}$ , there is

$$\begin{aligned} D_{RAP}^p &= D_{C_{RAP}^U}^{\overrightarrow{\quad}} + D_{C_{RAP}^{2U}}^{\overrightarrow{\quad}} = 8, \\ D_{OAP-I}^p &= D_{C_{OAP-I}^U}^{\overrightarrow{\quad}} + D_{C_{OAP-I}^{2U}}^{\overrightarrow{\quad}} = 10, \\ D_{OAP-II}^p &= D_{C_{OAP-II}^U}^{\overrightarrow{\quad}} + D_{C_{OAP-II}^{2U}}^{\overrightarrow{\quad}} = 9. \end{aligned} \quad (23)$$

If  $R_s^p = R_s^u + R_s^{2u} + R_s^{3u}$ , there is

$$\begin{aligned} D_{RAP}^p &= D_{C_{RAP}^U}^{\overrightarrow{\quad}} + D_{C_{RAP}^{2U}}^{\overrightarrow{\quad}} + D_{C_{RAP}^{3U}}^{\overrightarrow{\quad}} = 15, \\ D_{OAP-I}^p &= D_{C_{OAP-I}^U}^{\overrightarrow{\quad}} + D_{C_{OAP-I}^{2U}}^{\overrightarrow{\quad}} + D_{C_{OAP-I}^{3U}}^{\overrightarrow{\quad}} = 14, \\ D_{OAP-II}^p &= D_{C_{OAP-II}^U}^{\overrightarrow{\quad}} + D_{C_{OAP-II}^{2U}}^{\overrightarrow{\quad}} + D_{C_{OAP-II}^{3U}}^{\overrightarrow{\quad}} = 13. \end{aligned} \quad (24)$$

TABLE 1: Pixel block sizes within different PSR.

Sector radii	RAP	OAP-I	OAP-II
$R_s^p = R_s^u$	$2 \times 2$	$2 \times 3$	$2 \times 3$
$R_s^p = R_s^u + R_s^{2u}$	$3 \times 3$	$3 \times 4$	$3 \times 4$
$R_s^p = R_s^u + R_s^{2u} + R_s^{3u}$	$4 \times 4$	$4 \times 4$	$4 \times 4$

TABLE 2: Comparisons in terms of shifting rate and reference vector dimension  $N_{\text{cap}}/D$ .

Sector radii	$N_{\text{cap}}/D$ of RAP	$N_{\text{cap}}/D$ of OAP-I	$N_{\text{cap}}/D$ of OAP-II
$R_s^p = R_s^u$	<b>44210/3</b>	36044/4	35943/4
$R_s^p = R_s^u + R_s^{2u}$	<b>28653/8</b>	21672/10	21996/9
$R_s^p = R_s^u + R_s^{2u} + R_s^{3u}$	<b>20267/15</b>	17854/14	17954/13

Table 1 illustrates pixel block sizes within different PSR. Table 2 lists the embedding capacities in different PSR. Four features deserve to be noted as below.

- In all the three practical sector radii, the context pixel quantity and maximum prediction angle in RAP are both inferior to those in OAP-I and OAP-II. Accordingly, the former's EC is larger than that in the latter two. Comparing OAP-I with OAP-II, the former has bigger quantity and the latter has a little larger maximum prediction angle. Thus, the two methods output EC at almost the same level.
- Along with the larger sector radii, embedding capacities are all decreased progressively.
- When the active radii equal the minimum one  $R_s^p = R_s^u$ , OAP-I and OAP-II have the same context pixels and the maximum prediction angle. Yet, the former's embedding capacity is larger than the latter's. This may be caused by the different location of the current pixel.
- When the active radii are equal to the middle one  $R_s^p = R_s^u + R_s^{2u}$  or the maximum one  $R_s^p = R_s^u + R_s^{2u} + R_s^{3u}$ , OAP-I has the more context pixels than OAP-II, while the former's embedding capacity is lower than the latter's. This should be attributed by the more prepositive pixels in  $2U$  sector radii of OAP-II.

The forgoing conclusions declare that the context pixel quantity, the maximum prediction angle, and the current pixel location all have influence on the pixel prediction power. Probable prediction method should be adopted on request for payload size in reality.

Referring to literature [34], shifting bit ratio  $R_{\text{sft}}$  can generally measure the embedding distortion to the host image

$$R_{\text{sft}} = \frac{\#\{\text{shifted pixels}\}}{\#\{\text{expanded and shifted pixels}\}} \quad (25)$$

$$= \frac{N_{\text{sft}}}{N_{\text{cap}} + N_{\text{sft}}}$$

TABLE 3: Comparisons in terms of shifting rate and reference vector dimension  $R_{\text{sft}}/D$ .

Sector radii	$R_{\text{sft}}/D$ of RAP	$R_{\text{sft}}/D$ of OAP-I	$R_{\text{sft}}/D$ of OAP-II
$R_s^p = R_s^u$	0.7196/3	0.6581/4	<b>0.6581/4</b>
$R_s^p = R_s^u + R_s^{2u}$	0.6977/8	0.6312/10	<b>0.6085/9</b>
$R_s^p = R_s^u + R_s^{2u} + R_s^{3u}$	0.6894/15	0.6249/14	<b>0.5998/13</b>

Here, “#” denotes the cardinal number of some set.  $N_{\text{cap}}$  denotes the maximum capable embedding bits and  $N_{\text{sft}}$  denotes the maximum shifted bits. The less  $R_{\text{sft}}$  is, the better fidelity is.

Table 3 compares the shifted ratio  $R_{\text{sft}}$  in different sector radius. Some conclusions can be summarized as follows.

- In all the three practical sector radii, the context pixel quantity and maximum prediction angle in RAP are both inferior to those in OAP-I and OAP-II. Accordingly, the former's shifting ratio is larger than that of the latter two. Comparing OAP-I with OAP-II, the former has bigger quantity and the latter has a little larger maximum prediction angle. Thus, the two methods output shifting ratio at almost the same level.
- Along with the larger sector radii, the shifting ratios are all decreased progressively.
- When the active radii equal the minimum one  $R_s^p = R_s^u$ , OAP-I and OAP-II have the same context pixels and the maximum prediction angle. Thus, they get almost the same shifting ratios. As you see, the different locations of the current pixel of the two methods make negligible influence on shifting ratio. This variation is different from that of embedding capacity in Table 2.
- When the active radii are equal to the middle one  $R_s^p = R_s^u + R_s^{2u}$  or the maximum one  $R_s^p = R_s^u + R_s^{2u} + R_s^{3u}$ , OAP-I has more context pixels than OAP-II, while the former's shifting ratio is higher than latter's. This should be attributed by the more prepositive pixels in  $2U$  sector radii of OAP-II.

To sum up, in the minimum sector radii, the current pixel location has nearly no impact on shifting ratio. However, in the middle and maximum sector radius, obvious results appear.

**3.4. Prediction Factor Evaluation.** Prediction power is mainly determined by three factors, including the context vector dimension, the maximum prediction angle, and the location of the current pixel. For measuring the factor weight, prediction power is computed by

$$H = \begin{bmatrix} \mu \\ \lambda \\ k \end{bmatrix} \cdot [D \ \Theta \ \Gamma]. \quad (26)$$

TABLE 4: Factor weights in different prediction sector radius.

Sector radii	$\delta_1$	$\delta_2$	$\delta_3$
$R_s^P = R_s^u$	0.2405	<b>0.5879</b>	0.1716
$R_s^P = R_s^u + R_s^{2u}$	0.0819	<b>0.6510</b>	0.2671
$R_s^P = R_s^u + R_s^{2u} + R_s^{3u}$	0.1489	<b>0.5467</b>	0.3044

Here,  $H$  indicates the prediction power,  $D$  is the context vector dimension, theta  $\Theta$  is the maximum prediction angle, and gamma  $\Gamma$  is the location measurement of the current pixel. By experience, the variance location measurements from the first to the third current pixel take values  $\Gamma = [2.2 \ 3.3 \ 4.4]^T$ . The other three symbols  $\mu$ ,  $\lambda$ , and  $k$  are the factor coefficients. Therefore, a weight vector is derived by (23)

$$\begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{bmatrix} = \begin{bmatrix} \frac{\mu}{(\mu + \lambda + k)} \\ \frac{\lambda}{(\mu + \lambda + k)} \\ \frac{k}{(\mu + \lambda + k)} \end{bmatrix}. \quad (27)$$

Variances  $\delta_1$ ,  $\delta_2$ , and  $\delta_3$  denote the factor weight, respectively.  $\delta_1 \in [0, 1]$ ,  $\delta_2 \in [0, 1]$ ,  $\delta_3 \in [0, 1]$ . The bigger the values, the greater the impacts. On the contrary, the factor contributes less to prediction power.

Take the data in Table 1, for example, to calculate the prediction powers and factor weights. For facility to express the predicted embedding capacities,  $H$  is formulated by  $H = 10^{-4} \times EC$ . In the case of minimum radii  $R_s^P = R_s^u$ , prediction satisfies

$$\begin{bmatrix} 4.4210 \\ 3.6044 \\ 3.5943 \end{bmatrix} = \begin{bmatrix} \mu \\ \lambda \\ k \end{bmatrix} \cdot \begin{bmatrix} 3 & \frac{\pi}{2} & 2.2 \\ 4 & \frac{3\pi}{4} & 3.3 \\ 4 \arctan(2) + \frac{\pi}{2} & & 4.4 \end{bmatrix}. \quad (28)$$

$$\text{Hence, } \begin{bmatrix} \mu \\ \lambda \\ k \end{bmatrix} = \begin{bmatrix} 6.0542 \\ 14.7972 \\ 4.3190 \end{bmatrix} \text{ and } \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{bmatrix} = \begin{bmatrix} 0.2405 \\ 0.5879 \\ 0.1716 \end{bmatrix}.$$

Similarly, as shown in Table 4, factor weights in the middle and the maximum radius can be obtained.

Table 4 gives the following results.

- Regardless of the sector radii size, all three factors impact on embedding capacity. What is more, the maximum prediction angle has the largest weight. In the three cases shown in the table, this indicator works greater than 0.5.
- The location of the current pixel acts on embedding capacity increasingly along with the sector radii filling out.
- In the minimum sector radii, the location measurement is the least important factor. However, in the middle and the maximum sector radius, it exceeds the context pixel dimension.

TABLE 5: Factor weights in different prediction sector radius.

Sector radii	$\delta_1$	$\delta_2$	$\delta_3$
$R_s^P = R_s^u$	0.2506	<b>0.5798</b>	0.1696
$R_s^P = R_s^u + R_s^{2u}$	0.0929	<b>0.6436</b>	0.2635
$R_s^P = R_s^u + R_s^{2u} + R_s^{3u}$	0.2402	<b>0.5042</b>	0.2556

Take data in Table 2 as the case to declare the prediction power for shifting ratio. Considering the minimum sector radii, prediction power is calculated by

$$\begin{bmatrix} 0.7196 \\ 0.6581 \\ 0.6581 \end{bmatrix} = \begin{bmatrix} \mu \\ \lambda \\ k \end{bmatrix} \cdot \begin{bmatrix} 3 & \frac{\pi}{2} & 2.2 \\ 4 & \frac{3\pi}{4} & 3.3 \\ 4 \arctan(2) + \frac{\pi}{2} & & 4.4 \end{bmatrix}. \quad (29)$$

$$\text{Hence, } \begin{bmatrix} \mu \\ \lambda \\ k \end{bmatrix} = \begin{bmatrix} 0.8426 \\ 1.9500 \\ 0.5704 \end{bmatrix} \text{ and } \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{bmatrix} = \begin{bmatrix} 0.2506 \\ 0.5798 \\ 0.1696 \end{bmatrix}.$$

Factor weights in the middle and the maximum sector radius can be achieved in likewise manner and they shall be listed in Table 5.

The following deductions can be drawn from Table 5.

- No matter what the sector radii is, three factors operate on shifting ratio prediction as well. The maximum prediction angles, having the greatest weights, are superior to 0.5.
- Along with the predicted sector radii increasing, the prediction vector dimensions started to decrease and then lifted. In the minimum and the maximum radius, it is about 0.25 and in the middle radii, it decreases to be smaller than 0.10.
- Coupling with the sector radii extending, location measurement for the current pixel goes up first and drops down afterwards. In the minimum radii, it is about 0.17 and in the middle and the maximum radius, the same index becomes larger than 0.25.

**3.5. Prediction Power Test Procedure.** As shown in Figure 9, the four steps of prediction power test procedure are listed below.

*Step 1.* In order to acquire large embedding capacity and high image fidelity, block should be determined to be a proper size  $S_B = \text{rows} \times \text{columns}$ . The bigger block size having much more context pixels shall lead to more accuracy prediction and low distortion. Meanwhile, the smaller block size shall predict pixels with more probability and this may produce larger embedding capacity. In this paper, the selected minimum block is  $2 \times 2$  and the maximum block is  $4 \times 4$ .

*Step 2* (confirming the location of the current pixel). According to alternative methods, the current pixel should be located in the top-left position in RAP, in the next rightward one in OAP-I, and in the third rightward one in OAP-II.

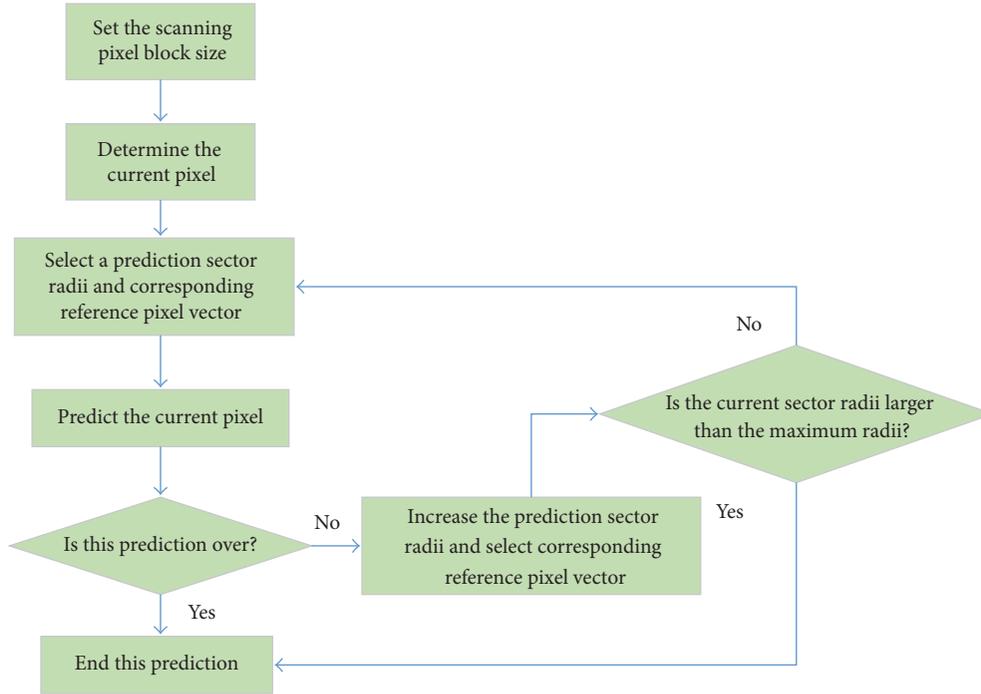


FIGURE 9: Prediction power test procedure.

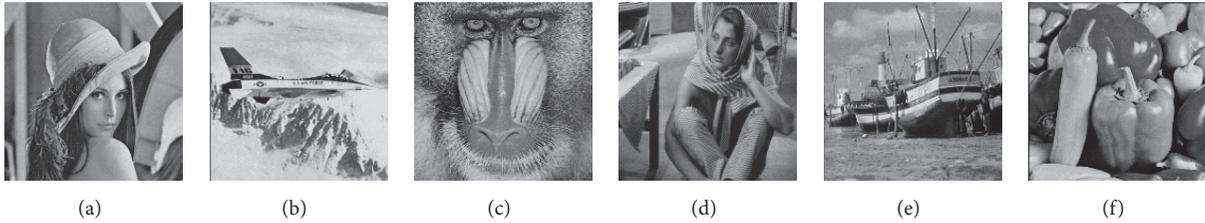


FIGURE 10: Six test gray images of SIPI image set. (a) Lena. (b) Airplane-F16. (c) Baboon. (d) Barbara. (e) Boat. (f) Peppers.

Step 3 (selecting prediction sector radii and corresponding context vector). Enlarge the radii from the minimum value to the maximum one to get pixel prediction values.

Step 4 (embedding payload and auxiliary bit stream and analyzing the factors' weight distribution). After pixel predictions, take the same procedure with PPVO to embed data into the host image. Compare with the original host image to get the output PSNR and then investigate factors' weight by (26) and (27).

#### 4. Experiment Results

Two balancing indexes including EC and PSNR will be used to evaluate the performance of the proposed method. Six test gray images listed in Figure 10, sized  $512 \times 512$ , are all selected from SIPI image set. Payload is the same random bit stream and Matlab 2013a is adopted for all the experiments.

4.1. EC Performance of Full Rank Prediction. Table 6 lists predicted EC with full rank of context pixel vectors in different

sector radius. Table 7 compares the prediction factor weights for EC. In the minimum and the middle sector radius, some likewise conclusions with Table 4 can be summarized. That is, each factor weight is highly consistent and this declares that prediction factors have the same effect regardless of rough or smooth images. However, in the maximum radii case, factor weights fluctuate greatly. And this indicates noticeable specific gravity under this condition. For this reason, Figure 11 portrays the concrete factor weight curves.

In Figure 11, factors are labeled on the horizontal axis and corresponding weights are marked by vertical axis. Coordinate vector (1, 2, 3, 4, 5, 6, 7, 8, 9) maps to the factors  $(\delta_1^{1U}, \delta_2^{1U}, \delta_3^{1U}, \delta_1^{2U}, \delta_2^{2U}, \delta_3^{2U}, \delta_1^{3U}, \delta_2^{3U}, \delta_3^{3U})$  of different sector radius. In the maximum radii, factors have distinct weight values. The most significant factor is the maximum prediction angle, the second-rate one is the current pixel location, and the least one is the context vector dimension. Lena, Barbara, and Peppers are common smooth images having pixel gray value distribution and factor weights are basically the same. Context vector dimensions have the least weight  $\delta_1^{3U} \in [0.14, 0.16]$  and the least variant range  $\sigma \in (0, 0.02)$ . The

TABLE 6: Predicted EC with full rank context vectors.

Image	Method	EC of 1U radii	EC of 2U radii	EC of 3U radii
F16	RAP	67015	45547	32503
	OAP-I	59086	38906	32833
	OAP-II	59002	41426	34818
Baboon	RAP	14152	8971	6383
	OAP-I	12379	7528	6271
	OAP-II	12347	7830	6368
Barbara	RAP	35022	23219	17239
	OAP-I	29882	19067	16296
	OAP-II	29824	20016	16997
Boat	RAP	29427	19179	13359
	OAP-I	25629	16315	13603
	OAP-II	25581	17198	14208
Peppers	RAP	33579	21664	15575
	OAP-I	29780	17955	15001
	OAP-II	29744	18954	15797

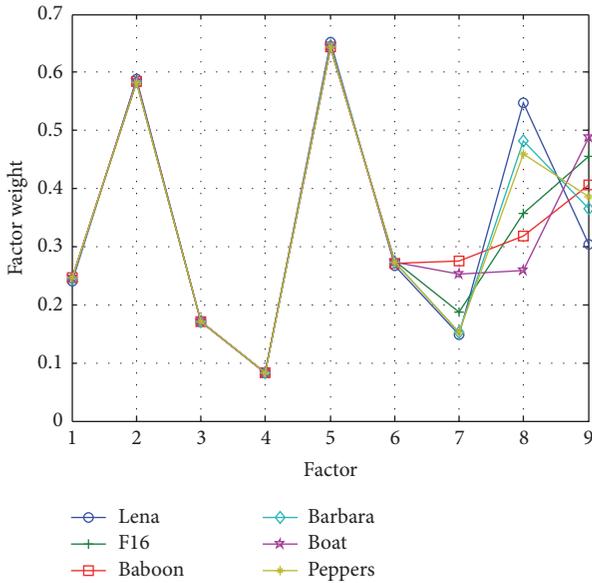


FIGURE 11: Factor weights for predicted EC in the maximum sector radii for different images.

maximum prediction angles have the largest weight  $\delta_2^{3U} \in [0.45, 0.55]$  and the current pixel locations have the middle weight  $\delta_3^{3U} \in [0.30, 0.40]$ . The subsequent two factors vary within almost the same range  $\sigma \in (0, 1)$ . Baboon and Boat are rough images having tremendous changing pixels and complex distributions. Context vector dimensions have the least influence  $\delta_1^{3U} \in [0.27, 0.28]$  varying within  $\sigma \in (0, 0.01)$ . The maximum prediction angles have general values  $\delta_2^{3U} \in [0.25, 0.32]$  on the verge of the former. The current pixel locations have the largest influence  $\delta_3^{3U} \in [0.40, 0.50]$ .

Airplane is smooth image having gentle varying pixels. The least weight is context vector dimension  $\delta_1^{3U} = 0.1874$ , but it is superior to that of Lena, Barbara, and Peppers. The middle weight is the maximum prediction angle  $\delta_2^{3U} = 0.3572$  under that of common smooth images and exceeding that of rough images. The largest weight is the current pixel location  $\delta_3^{3U} = 0.4554$  higher than that of common smooth images.

In brief, four viewpoints can be reported as follows.

- In the minimum and the middle sector radius, factor weights for predicted EC are highly accordant and have nothing to do with image contents.
- In the minimum sector radii, the maximum prediction angle having the largest weight is nearly 0.60 and the context vector dimension having the least weight is about 0.17.
- In the middle sector radii, the most weighted factor is the maximum prediction angle, the second-rate one is the current pixel location measurement and the least weighted one is the context dimension. These three factors are about 0.65, 0.27, and 0.08, respectively.
- In the maximum sector radii, factors become quite different. As far as varying range, the maximum prediction angle plays the most obvious function, the current pixel location takes the second place, and the context vector dimension shows the weakest one.

#### 4.2. Shifting Ratio Performance of Full Rank Prediction.

Table 8 illustrates predicted shifting ratio with full rank of context pixel vectors in different sector radius. Table 9 makes a comparison to the prediction factor weights for shifting ratio. In the minimum and the middle sector radius, some similar outcomes with Table 5 can be summed up. That is, each factor weight is extremely accordant and this proves that factors are not relevant to image content, no matter rough or smooth. However, in the maximum radii case, factor weights change violently. And this means not tolerated gravity at this situation. Therefore, Figure 12 depicts the specific weight curves.

In Figure 12, the horizontal and the vertical coordinates possess the same meaning with those in Figure 11. It is not difficult to find that factors impact on predicted shifting ratio differs widely. In the maximum sector radii case, factors are related to image contents. Big to small, factor weights are in the order of the maximum prediction angle, the context vector dimension, and the current pixel location. Airplane, Lena, Barbara, and Peppers have the alike patterns.

The maximum prediction angles have the largest weight  $\delta_2^{3U} \in [0.57, 0.59]$ , the context vector dimensions take the second place  $\delta_1^{3U} \in [0.24, 0.26]$ , and the current pixel location has the least weight  $\delta_3^{3U} \in [0.16, 0.17]$ . All the three factors vary a little  $\sigma \in (0, 0.02)$ . As for Baboon, the most rough image, from large to small, factors list in the order of the current pixel location  $\delta_3^{3U} = 0.4181$ , the context vector dimension  $\delta_1^{3U} = 0.3766$ , and the maximum prediction angle  $\delta_2^{3U} = 0.2052$ . Apparently, these are approximate indexes. For Boat, a common rough image, factors alter between Baboon

TABLE 7: Factor weights for predicted EC with full rank context vectors.

Image	$\delta_1^{1U}$	$\delta_2^{1U}$	$\delta_3^{1U}$	$\delta_1^{2U}$	$\delta_2^{2U}$	$\delta_3^{2U}$	$\delta_1^{3U}$	$\delta_2^{3U}$	$\delta_3^{3U}$
Lena	0.2405	0.5879	0.1716	0.0819	0.6510	0.2671	0.1489	0.5467	0.3044
F16	0.2469	0.5829	0.1703	0.0822	0.6424	0.2754	0.1874	0.3572	0.4554
Baboon	0.2463	0.5835	0.1703	0.0833	0.6447	0.2720	0.2751	0.3191	0.4058
Barbara	0.2439	0.5852	0.1709	0.0821	0.6455	0.2724	0.1527	0.4812	0.3661
Boat	0.2458	0.5838	0.1705	0.0828	0.6431	0.2741	0.2522	0.2594	0.4884
Peppers	0.2474	0.5824	0.1702	0.0820	0.6447	0.2733	0.1532	0.4602	0.3866

TABLE 8: Shifting ratios of full rank prediction.

Image	Method	EC of 1U radii	EC of 2U radii	EC of 3U radii
F16	RAP	0.5886	0.5570	0.5479
	OAP-I	0.5310	0.5040	0.5019
	OAP-II	0.5310	0.5005	0.4993
Baboon	RAP	0.8977	0.8810	0.8666
	OAP-I	0.8821	0.8668	0.8558
	OAP-II	0.8822	0.8673	0.8582
Barbara	RAP	0.7711	0.7389	0.7265
	OAP-I	0.7160	0.6830	0.6732
	OAP-II	0.7159	0.6825	0.6739
Boat	RAP	0.8013	0.7705	0.7656
	OAP-I	0.7716	0.7462	0.7435
	OAP-II	0.7716	0.7546	0.7521
Peppers	RAP	0.7745	0.7375	0.7360
	OAP-I	0.7175	0.6820	0.6765
	OAP-II	0.7175	0.6846	0.6772

and other smooth images. Then, the least weight is context vector dimension  $\delta_1^{3U} = 0.2584$ , the middle weight is the maximum prediction angle  $\delta_2^{3U} = 0.3592$ , and the largest weight is the current pixel location  $\delta_3^{3U} = 0.3824$ .

From above, the following opinions come into being.

- In the minimum and the middle sector radius, factor weights for predicted shifting ratio are almost uniform and have no relationship to image contents.
- In the minimum sector radii, the maximum prediction angle having the largest weight is nearly 0.60 and the context vector dimension having the least weight is about 0.17. The current pixel location weights about 0.25.
- In the middle sector radii, the most weighted factor is the maximum prediction angle, the second-rate one is the current pixel location measurement, and the least weighted one is the context dimension. These three factors are about 0.65, 0.27, and 0.09, respectively.
- In the maximum sector radii, factors vary quite a lot. Differently from the scales of aforementioned factor

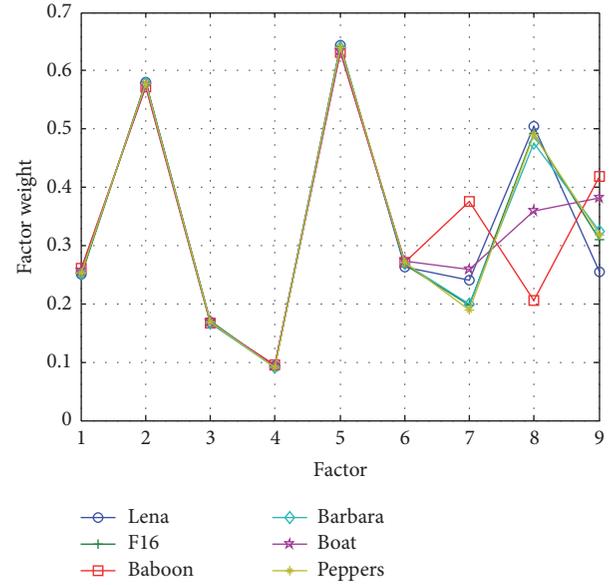


FIGURE 12: Factor weights for shifting ratio in the maximum sector radii for different images.

scales for EC, the maximum prediction angle still acts the most significant part, but the sequent order becomes from the context vector dimension to the current pixel location.

**4.3. Optimal Performance.** In this section, the optimal PSNR performance will be compared for six outstanding RDH methods, including four preexisting approaches from literatures [34, 37–39] and two new proposed ones. For ensuring objectivity, tested payloads are selected as the same bit stream for different methods. OAP-I and OAP-II refer to predict pixels in obtuse angle for suitable block size from  $2 \times 2$  to  $4 \times 4$  as soon as possible. Hence, Qu et al.'s RAP, OAP-I, and OAP-II have the least 2 pixels and the most 15, 14, and 13 pixels, respectively. The highest PSNR value is identified to be the optimal value.

The top embedding capacities of PVO, RAP, OAP-I, and OAP-II are listed in Table 10. On the whole, EC of OAP-I or OAP-II is a little less than that of RAP. Yet, their EC values are much the same. Obviously, PVO outputs the least EC of them. Specifically, the average EC enhancements versus PVO

TABLE 9: Factor weights for predicted shifting ratio with full rank context vectors.

Image	$\delta_1^{1U}$	$\delta_2^{1U}$	$\delta_3^{1U}$	$\delta_1^{2U}$	$\delta_2^{2U}$	$\delta_3^{2U}$	$\delta_1^{3U}$	$\delta_2^{3U}$	$\delta_3^{3U}$
Lena	0.2506	0.5798	0.1696	0.0929	0.6436	0.2635	0.2402	0.5042	0.2556
F16	0.2490	0.5810	0.1700	0.0902	0.6418	0.2680	0.1976	0.4914	0.3110
Baboon	0.2608	0.5719	0.1673	0.0957	0.6323	0.2721	0.3766	0.2052	0.4181
Barbara	0.2524	0.5784	0.1692	0.0909	0.6395	0.2695	0.2002	0.4759	0.3238
Boat	0.2575	0.5745	0.1680	0.0930	0.6337	0.2733	0.2584	0.3592	0.3824
Peppers	0.2521	0.5786	0.1693	0.0905	0.6392	0.2703	0.1902	0.4909	0.3189

TABLE 10: Comparisons in view of optimal embedding capacities (/bits).

Image	PVO	RAP	OAP-I	OAP-II
Lena	31941	46930	46419	46315
Airplane	38896	69048	68773	68619
Baboon	13144	14772	14735	14702
Barbara	25767	36628	36544	36484
Boat	24126	30368	30238	30167
Peppers	27976	33596	33531	33466
Average	26775	39549	39342	39257

to RAP, OAP-I, and OAP-II are 12774 bits, 12567 bits, and 12482 bits.

Tables 11 and 12 compare the optimal PSNR values for 10000 bits and 20000 bits payloads. EC of Baboon is not illustrated in Table 12 for smaller space than 20000 bits. Inspecting Table 11, when the payload is 10000 bits, OAP-I improves RAP by 0.34 dB, 0.05 dB,  $-0.45$  dB, 0.58 dB, 0.04 dB, and 0.32 dB, with the average of 0.14 dB. OAP-II improves RAP by 0.41 dB, 0.01 dB,  $-0.48$  dB, 0.53 dB, 0.11 dB, and 0.23 dB, with the average of 0.09 dB. Referenced by Table 12, when the payload is 20000 bits, OAP-I improves RAP by 0.03 dB, 0.05 dB, 0.01 dB, 0.22 dB, and 0.29 dB, with the average of 0.32 dB. OAP-II improves RAP by 0.12 dB, 0.02 dB,  $-0.04$  dB,  $-0.08$  dB, and 0.20 dB, with the average of 0.25 dB. Thus, OAP-I prevails a little over OAP-II.

Figure 13 compares the optimal PSNR performance. The minimum payload is selected as 5000 bits and the step is 2000. Results are described in four aspects.

- (a) All the tested methods produce good embedding image fidelity. PSNR values are dependent on image contents. From Figures 13(a) to 13(e), Luo et al. [38] yield the lowest purple curve and Hong [39] produces a bit over cyan curve, especially in low-medium payloads. Qu and Kim's RAP [37] portrays the green curve exceeding them in all payload sizes. The black curve generated by Li et al.'s PVO [34] is near to RAP curve in low payload and decreases quickly. For Lena, Barbara, Boat, and Peppers, Li et al.'s PVO gets higher PSNR than Hong's method in high-medium payload. For Barbara and Peppers, PVO is higher than RAP. Striking colors are used

for painting the red OAP-I and the blue OAP-II curves. Experiment results demonstrate that OAP type methods improve optimal PSNR values visibly especially in high-medium payload. In Figure 13(f), a cross appears at the nearly 20000 bits' point for Hong's and Luo et al.'s methods. And then, the former declines more quickly than the latter.

- (b) PSNR varies in the same tendency while it differs in severity with uniformly changing payload. Linearity keeps well in small payload and nonlinearity performs prominently in large payload. PVO comes down turning into medium payload and then transfers into gentle stages. For Lena, Airplane, Barbara, Boat, and Peppers, PVO descends 0.18 dB, 0.08 dB, 0.50 dB, 0.32 dB, and 0.48 dB individually when the payload size is 23000 bits, 27000 bits, 19000 bits, 19000 bits, and 21000 bits.
- (c) OAP-I and OAP-II deserve the overwhelming advantages especially in moderate and large payload size. In such conditions, RAP is improved visibly.
- (d) For rough image, such as Baboon, different methods cause results quite a lot. OAP-I, OAP-II, RAP, and Hong hold fine linearity in small payload and tend to coincide in large-moderate payload size. PVO acts the supreme in the beginning while falling down quickly in the middle and later periods. Though Luo et al.'s method has the lowest measurement values, it is still in excess of 50 dB with gentle variation.

## 5. Conclusion

In this paper, an OAP prediction scheme well exploiting the location relevance of context pixels is presented. The main contributions can be summarized by two key points. First, OAP develops the pixel-mannered prediction method for more pixel distribution coherence. Results demonstrate higher PSNR performance than conventional PPVO method. Second, a mathematical model for evaluating pixel prediction ability is constructed. Here, prediction power specified the evaluating index. In detail, three factors are analyzed, including the context vector dimension, the prediction angle, and the current pixel location. Furthermore, comparing with other four state-of-the-art methods, the proposed OAP scheme performs prominently.

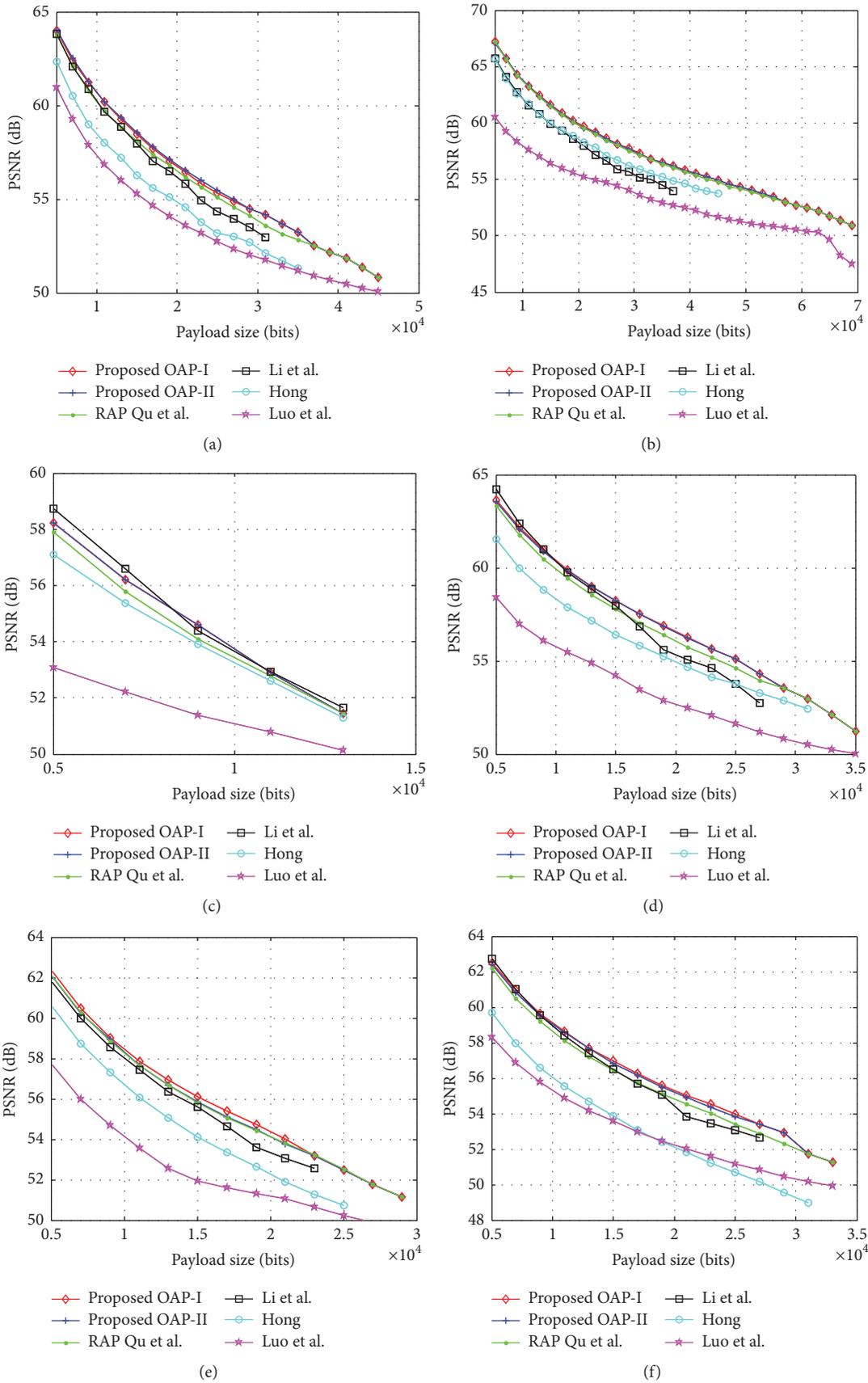


FIGURE 13: The optimal PSNR performance comparisons for the six test images of SIPI image data set. (a) Lena. (b) Airplane. (c) Baboon. (d) Barbara. (e) Boat. (f) Peppers.

TABLE 11: Comparisons in view of PSNR (dB) for [32, 34, 37, 38], OAP-I, and OAP-II with the payload of 10000 bits.

Image	Luo et al.	Hong	Li et al.	RAP	OAP-I	OAP-II
Lena	57.31	58.50	59.86	60.31	60.65	60.72
Airplane	57.97	62.08	61.61	63.72	63.77	63.73
Baboon	51.06	56.13	53.50	54.20	53.75	53.72
Barbara	55.74	58.33	59.98	59.82	60.40	60.35
Boat	54.06	56.63	58.11	58.43	58.47	58.32
Peppers	55.29	56.10	56.12	58.81	59.13	59.04
Average	55.24	57.96	58.20	59.22	59.36	59.31

TABLE 12: Comparisons in view of PSNR (dB) for [34, 37–39], OAP-I, and OAP-II with the payload of 20000 bits.

Image	Luo et al.	Hong	Li et al.	RAP	OAP-I	OAP-II
Lena	53.83	54.90	56.22	56.70	56.73	56.82
Airplane	55.44	58.59	58.14	59.92	59.97	59.94
Barbara	52.66	54.94	54.73	55.61	56.62	56.57
Boat	51.19	52.29	53.31	54.20	54.42	54.12
Peppers	52.21	52.15	54.72	55.03	55.32	55.23
Average	53.07	54.57	55.42	56.29	56.61	56.54

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

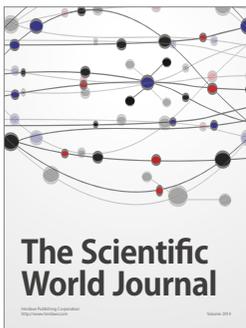
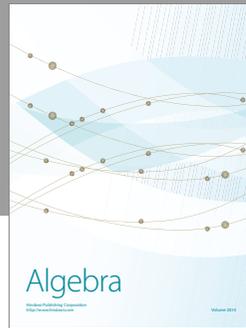
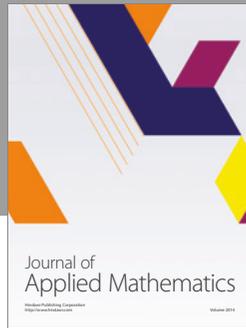
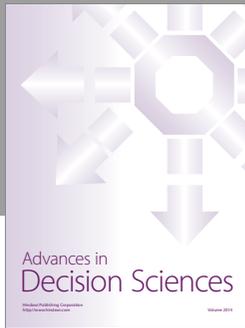
## Acknowledgments

This work was supported by National Natural Science Foundation of China (no. 61302106 and no. 61201393), the Fundamental Research Funds for the Central Universities (no. 2014MS105 and no. 13MS66), and Hebei Province Natural Science Foundation of China, Youth Science Fund (no. E2013502267). Here, the authors are obliged to the above foundations, their responsible persons, and all their lab associates.

## References

- [1] J. M. Barton, "Method and apparatus for embedding authentication information within digital data," U.S. Patent 5 646 997, 1997.
- [2] C. W. Honsinger, P. W. Jones, M. Rabbani, and J. C. Stoffel, "Lossless recovery of an original image containing embedded data," U.S. Patent 6 278 791, 2001.
- [3] S. Y. Chen, H. Tong, Z. Wang, S. Liu, M. Li, and B. Zhang, "Improved generalized belief propagation for vision processing," *Mathematical Problems in Engineering*, vol. 2011, Article ID 416963, 12 pages, 2011.
- [4] K. Hwang and D. Li, "Trusted cloud computing with secure resources and data coloring," *IEEE Internet Computing*, vol. 14, no. 5, pp. 14–22, 2010.
- [5] G. Coatrieux, C. Le Guillou, J.-M. Cauvin, and C. Roux, "Reversible watermarking for knowledge digest embedding and reliability control in medical images," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 2, pp. 158–165, 2009.
- [6] F. Bao, R. H. Deng, B. C. Ooi, and Y. Yang, "Tailored reversible watermarking schemes for authentication of electronic clinical atlas," *IEEE Transactions on Information Technology in Biomedicine*, vol. 9, no. 4, pp. 554–563, 2005.
- [7] D. Coltuc and I. Caciula, "Stereo embedding by reversible watermarking: further results," in *Proceedings of the International Symposium on Signals, Circuits and Systems (ISSCS '09)*, pp. 1–4, IEEE, Iași, Romania, July 2009.
- [8] X. F. Tong, G. C. Shen, G. R. Xuan et al., "Stereo image coding with histogram-pair based reversible data hiding," in *Digital-Forensics and Watermarking: 13th International Workshop, IWDFW 2014, Taipei, Taiwan, October 1–4, 2014. Revised Selected Papers*, vol. 9023 of *Lecture Notes in Computer Science*, pp. 201–214, Springer International Publishing, Cham, Switzerland, 2014.
- [9] J. Fridrich, M. Goljan, and R. Du, "Invertible authentication," in *Security and Watermarking of Multimedia Contents III*, Proceedings of SPIE, pp. 197–208, San Jose, Calif, USA, January 2001.
- [10] M. Goljan, J. J. Fridrich, and R. Du, "Distortion-free data embedding for images," in *Proceedings of the 4th ACM Information Hiding and Multimedia Security Workshop*, vol. 2137, pp. 27–41.
- [11] J. Fridrich, M. Goljan, and R. Du, "Lossless data embedding—new paradigm in digital watermarking," *EURASIP Journal on Applied Signal Processing*, no. 2, pp. 185–196, 2002.
- [12] G. Xuan, J. Zhu, J. Chen, Y. Q. Shi, Z. Ni, and W. Su, "Distortionless data hiding based on integer wavelet transform," *Electronics Letters*, vol. 38, no. 25, pp. 1646–1648, 2002.
- [13] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Lossless generalized-LSB data embedding," *IEEE Transactions on Image Processing*, vol. 14, no. 2, pp. 253–266, 2005.
- [14] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890–896, 2003.
- [15] J. Tian, "Wavelet-based reversible watermarking for authentication," in *Security and Watermarking of Multimedia Contents IV*, vol. 4675 of *Proceedings of SPIE*, pp. 679–690, April 2002.

- [16] L.-T. Ko, J.-E. Chen, Y.-S. Shieh, M. Scalia, and T.-Y. Sung, "A novel fractional-discrete-cosine-transform-based reversible watermarking for healthcare information management systems," *Mathematical Problems in Engineering*, vol. 2012, Article ID 757018, 17 pages, 2012.
- [17] A. M. Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE Transactions on Image Processing*, vol. 13, no. 8, pp. 1147–1156, 2004.
- [18] D. Coltuc and J.-M. Chassery, "Very fast watermarking by reversible contrast mapping," *IEEE Signal Processing Letters*, vol. 14, no. 4, pp. 255–258, 2007.
- [19] S. Weng, Y. Zhao, J.-S. Pan, and R. Ni, "Reversible watermarking based on invariability and adjustment on pixel pairs," *IEEE Signal Processing Letters*, vol. 15, pp. 721–724, 2008.
- [20] X. Wang, X. Li, B. Yang, and Z. Guo, "Efficient generalized integer transform for reversible watermarking," *IEEE Signal Processing Letters*, vol. 17, no. 6, pp. 567–570, 2010.
- [21] D. Coltuc, "Low distortion transform for reversible watermarking," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 412–417, 2012.
- [22] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, vol. 2, pp. II912–II915, May 2003.
- [23] A. van Leest, M. van der Veen, and F. Bruekers, "Reversible image watermarking," in *Proceedings of the IEEE International Conference on Image Processing*, vol. 2, pp. 731–734, September 2003.
- [24] Y.-S. Juang, L.-T. Ko, J.-E. Chen, Y.-S. Shieh, T.-Y. Sung, and H. C. Hsin, "Histogram modification and wavelet transform for high performance watermarking," *Mathematical Problems in Engineering*, vol. 2012, Article ID 164869, 14 pages, 2012.
- [25] M. Fallahpour and M. H. Sedaaghi, "High capacity reversible image watermarking based on integer transform," *IEICE Electronics Express*, vol. 4, no. 7, pp. 205–210, 2007.
- [26] S.-K. Lee, Y.-H. Suh, and Y.-S. Ho, "Reversible image authentication based on watermarking," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '06)*, vol. 1, pp. 1321–1324, July 2006.
- [27] G. Xuan, Y. Q. Shi, P. Chai, X. Cui, Z. Ni, and X. Tong, "Optimum histogram pair based image lossless data embedding," in *Proceedings of the 7th International Workshop on Digital Watermarking*, vol. 5041, pp. 264–278, 2007.
- [28] X. Li, B. Li, B. Yang, and T. Zeng, "General framework to histogram-shifting-based reversible data hiding," *IEEE Transactions on Image Processing*, vol. 22, no. 6, pp. 2181–2191, 2013.
- [29] L.-T. Ko, J.-E. Chen, H.-C. Hsin, Y.-S. Shieh, and T.-Y. Sung, "Haar-wavelet-based just noticeable distortion model for transparent watermark," *Mathematical Problems in Engineering*, vol. 2012, Article ID 635738, 14 pages, 2012.
- [30] D. M. Thodi and J. J. Rodríguez, "Expansion embedding techniques for reversible watermarking," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 721–730, 2007.
- [31] Y. Hu, H.-K. Lee, and J. Li, "DE-based reversible data hiding with improved overflow location map," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 2, pp. 250–260, 2009.
- [32] W. Hong, T.-S. Chen, and C.-W. Shiu, "Reversible data hiding for high quality images using modification of prediction errors," *Journal of Systems and Software*, vol. 82, no. 11, pp. 1833–1842, 2009.
- [33] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y. Q. Shi, "Reversible watermarking algorithm using sorting and prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 7, pp. 989–999, 2009.
- [34] 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.
- [35] F. Peng, X. Li, and B. Yang, "Improved PVO-based reversible data hiding," *Digital Signal Processing*, vol. 25, no. 1, pp. 255–265, 2014.
- [36] B. Ou, X. Li, Y. Zhao, and R. Ni, "Reversible data hiding using invariant pixel-value-ordering and prediction-error expansion," *Signal Processing: Image Communication*, vol. 29, no. 7, pp. 760–772, 2014.
- [37] 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.
- [38] L. X. Luo, Z. Y. Chen, M. Chen, X. Zeng, and Z. Xiong, "Reversible image watermarking using interpolation technique," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 1, pp. 187–193, 2010.
- [39] W. Hong, "An efficient prediction-and-shifting embedding technique for high quality reversible data hiding," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, Article ID 104835, 12 pages, 2010.
- [40] X. Li, W. Zhang, X. Gui, and B. Yang, "Efficient reversible data hiding based on multiple histograms modification," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 9, pp. 2016–2027, 2015.



# Hindawi

Submit your manuscripts at  
<https://www.hindawi.com>

