CNN Prediction Based Reversible Data Hiding

Tables

1. Introduction

- How to predict images is an important issue in the reversible data hiding (RDH) community
- There still exists room for RDH by making full use of more neighboring pixels as the context of a to-be-predicted pixel
- Experimental results show that the CNNP can promote prediction performance due to the use of more surrounding pixels as the context

embed one bit $b = 1$

$$l = \left\lfloor \frac{206 + 201}{2} \right\rfloor = \left\lfloor \frac{407}{2} \right\rfloor = 203,\ h = 206 - 201 = 5$$

$$h' = \underbrace{2 \times h}_{\text{shifting}} + \underbrace{b}_{\text{embedding}} = 2 \times 5 + 1 = 11.$$

$$x' = 203 + \left\lfloor \frac{11 + 1}{2} \right\rfloor = 209,\ y' = 203 - \left\lfloor \frac{11}{2} \right\rfloor = 198.$$

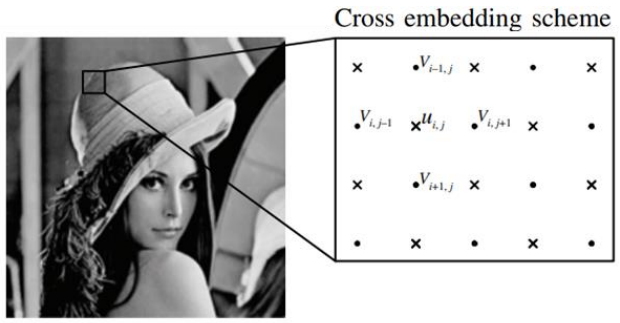
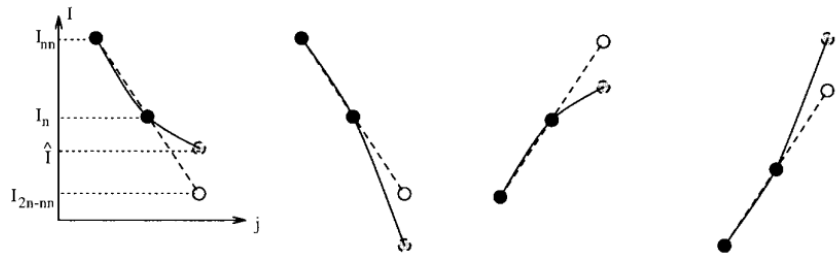


Fig. 1. Prediction pattern. The pixel value u of the Cross set can be predicted by using the four neighboring pixel values of the Dot set and expanded to hide one bit of data.



1. Introduction

- How to predict images is an important issue in the reversible data hiding (RDH) community
- There still exists room for RDH by making full use of more neighboring pixels as the context of a to-be-predicted pixel
- Experimental results show that the CNNP can promote prediction performance due to the use of more surrounding pixels as the context

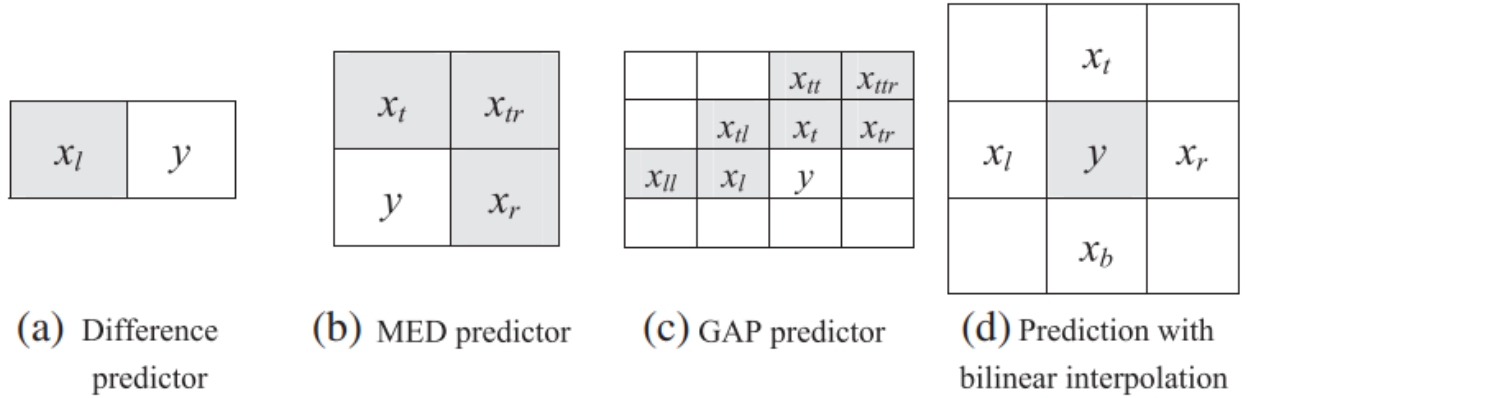


Fig. 1 Prediction context of a pixel. **a** Difference prediction with one causal pixel. **b** MED predictor with three causal pixels. **c** GAP predictor with seven causal pixels. **d** Noncausal prediction using bilinear interpolation with two past pixels and two future pixels

2. Proposed Method

A. *Pre-Processing Images*

- For the “Cross” set image, the pixel values of the positions belong to the “Dot” set are assigned to 0,
- and so does the “Dot” set image

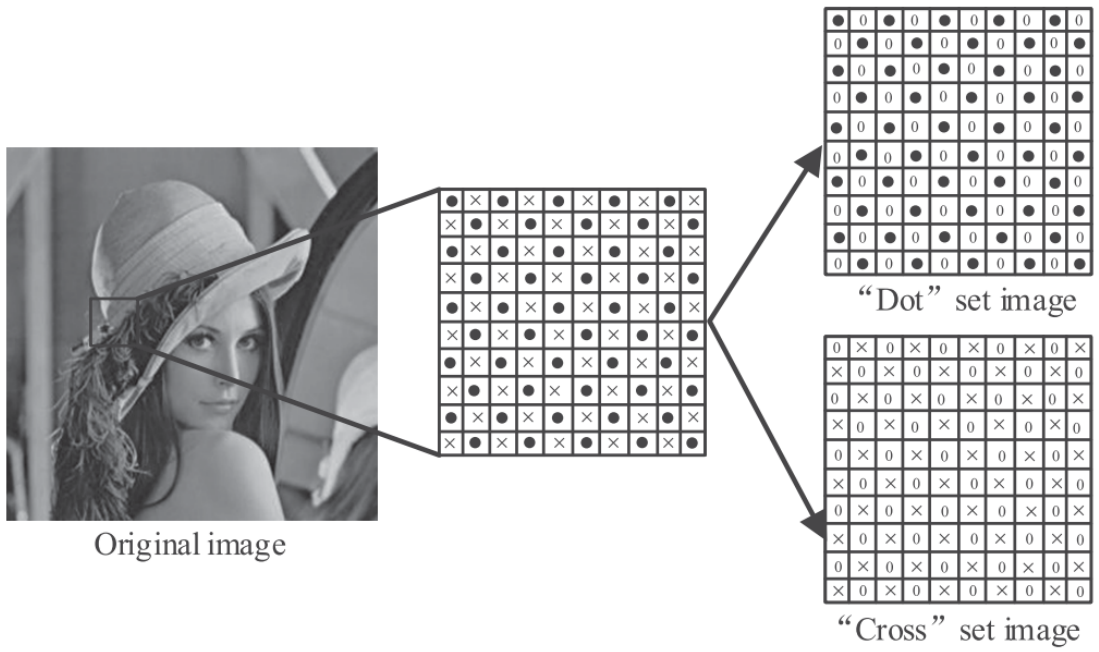


Fig. 1. Illustration to divide an image to “Dot” and “Cross” set images.

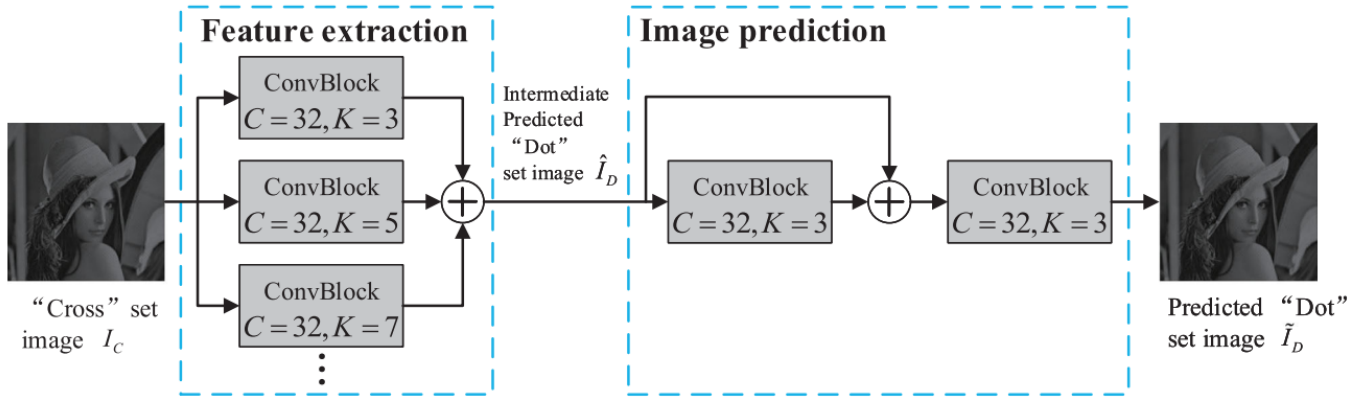
2. Proposed Method

B. Architecture Overview

- two main steps:
- the feature extraction step (for the use of the multi receptive fields)
- the image prediction step (for the use of global optimization)

$$\tilde{I}_D(x,y) = \sum_{i,j=1}^{K} I_C(x+i, y+j) \cdot w(i,j) + b,$$

input is the "Cross" set image I_C
the target is the "Dot" set image I_D
output of the proposed CNNP is the predicted "Dot" set image $\tilde{I}_D$



(a) Architecture of the proposed CNNP

(b) Structure of the convolution block in (a)

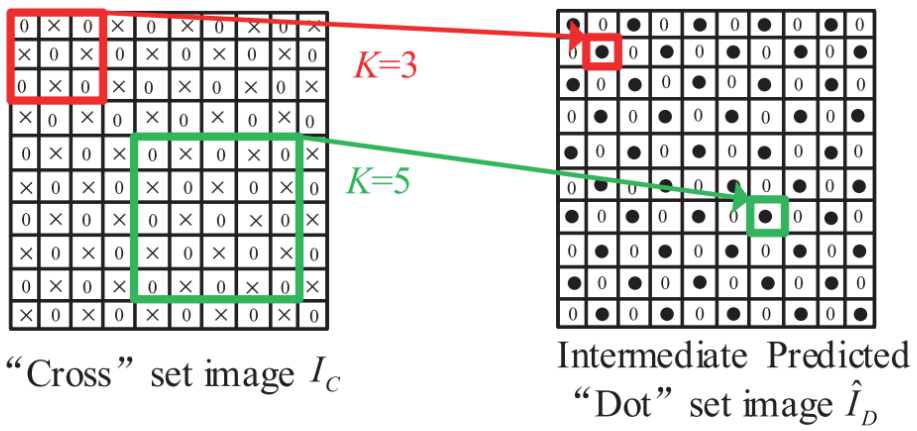
Fig. 2. Overview of the proposed CNNP.

2. Proposed Method

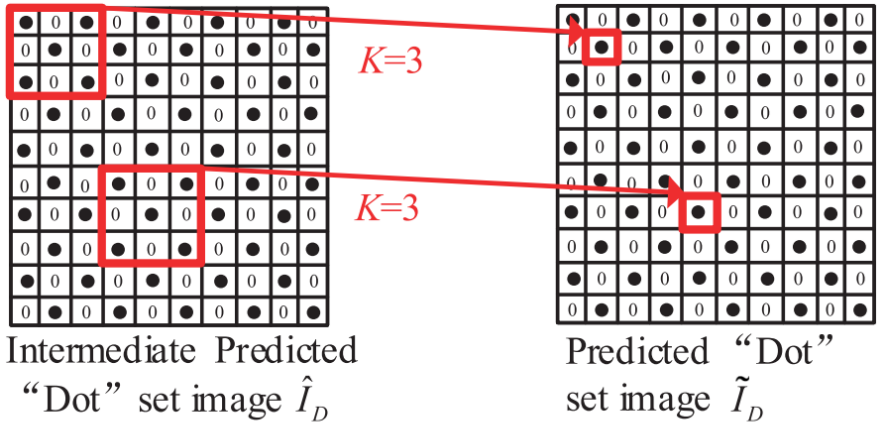
C. Training

$\lfloor K^2/2 \rfloor$ cross set pixels to predict the central dot set pixel

$$\text{loss} = \frac{1}{P}\sum_{i=1}^{P}\left(\tilde{I}_D - I_D\right)^2 + \lambda \left\|\omega\right\|_2^2,$$



(a) When $K = 3$ and $K = 5$ in the feature extraction step



(b) When $K = 3$ in the image prediction step

Fig. 3. Illustration on the use of convolution layers in Fig. 2(a).

2. Proposed Method

D. CNNP Based RDH Method

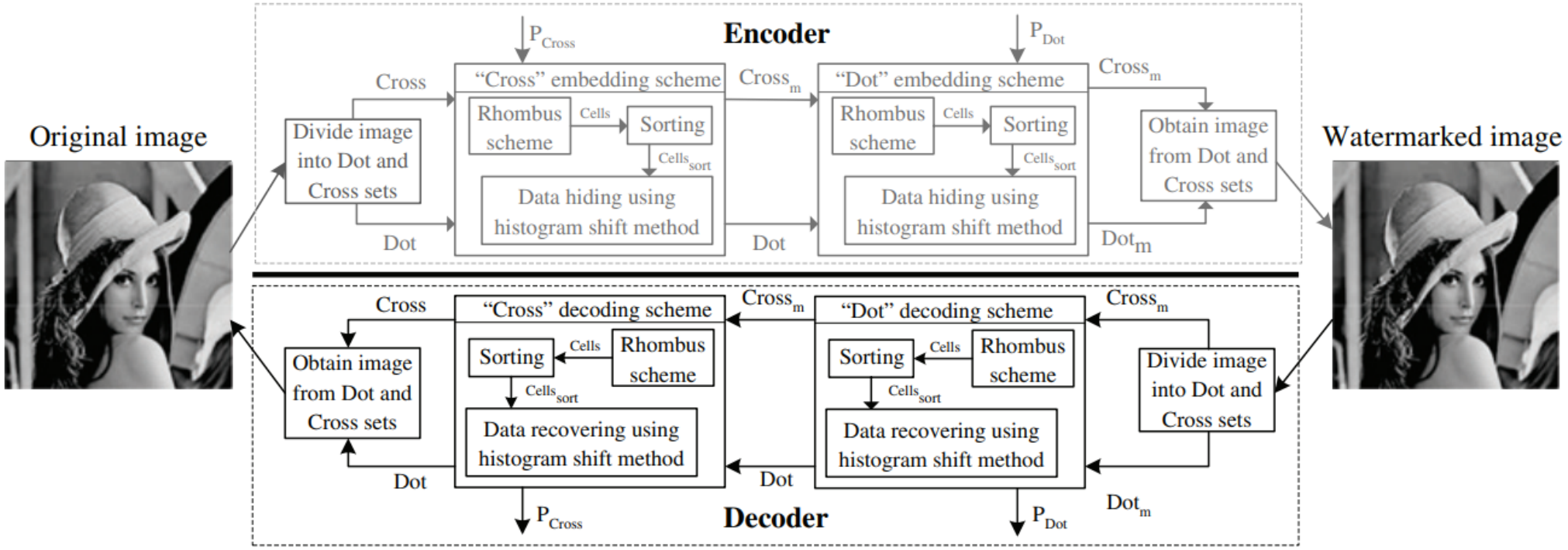


Fig. 7. Framework of the double encoding and decoding scheme.

2. Proposed Method

D. CNNP Based RDH Method

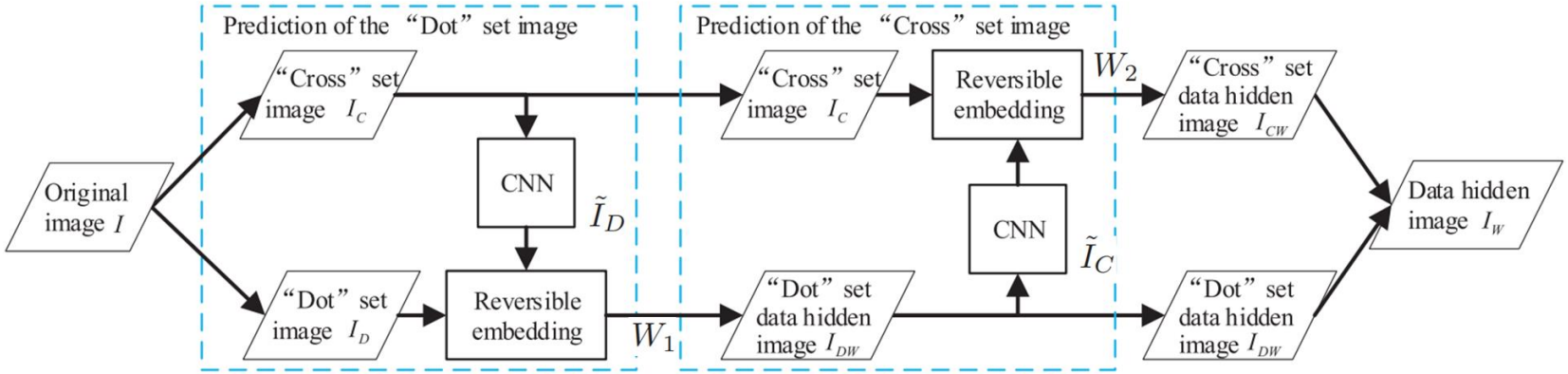


Fig. 4. The proposed reversible data embedding scheme.

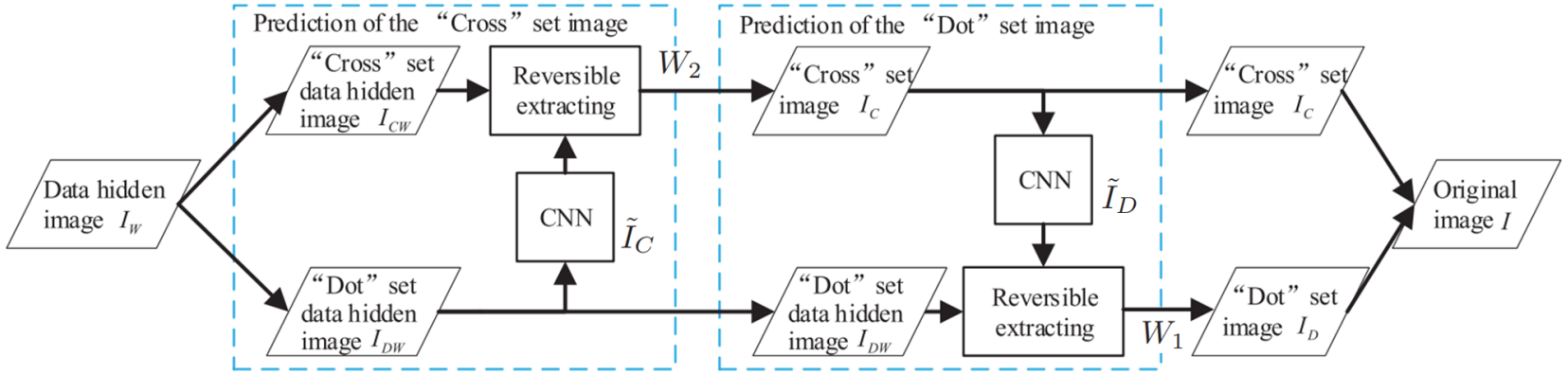


Fig. 5. The proposed reversible data extracting scheme.

3. Experimental Results

A. *Prediction Accuracy*

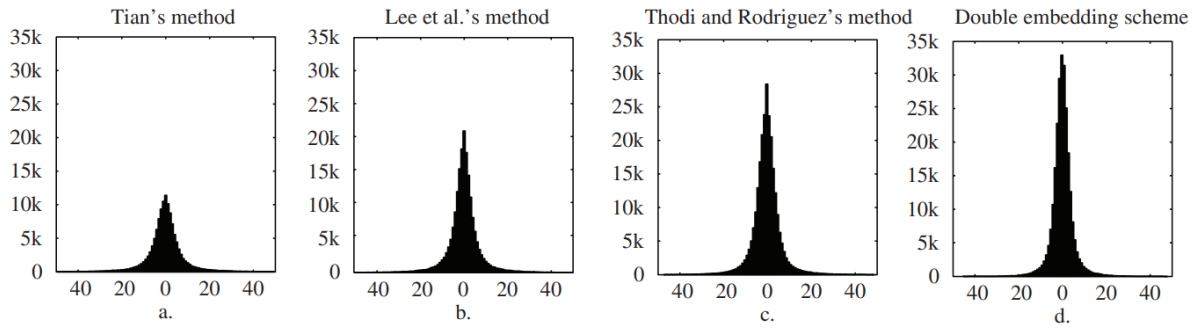


Fig. 2. (a) Histogram of differences between neighboring pixels. (b) Histogram of high-frequency wavelet coefficients. (c) Histogram of JPEG-LS prediction errors. (d) Histogram of prediction errors for Double embedding scheme and (e) for Lena image.

TABLE I
AVERAGE MSE, ABSOLUTE MEAN AND VARIANCE OF THE PREDICTION ERRORS IN 100 IMAGES FOR FIVE DIFFERENT PREDICTORS

Predictor	CNNP	BIP	MEDP	GAP	DP
MSE	99.4	154.8	234.2	231.9	230.8
Mean	4.77	6.25	7.37	9.86	5.13
Variance	66.9	100.5	161.3	167.6	196.6

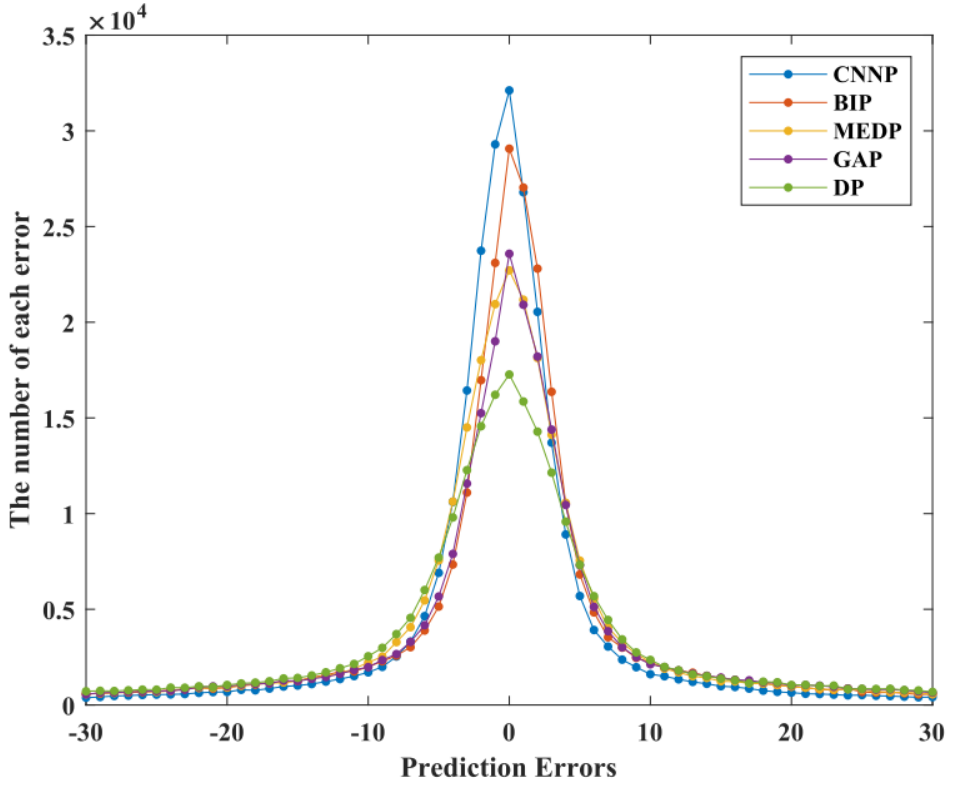


Fig. 6. Histograms of the image *Lena* under five different predictors.

3. Experimental Results

B. *Embedding Performance*

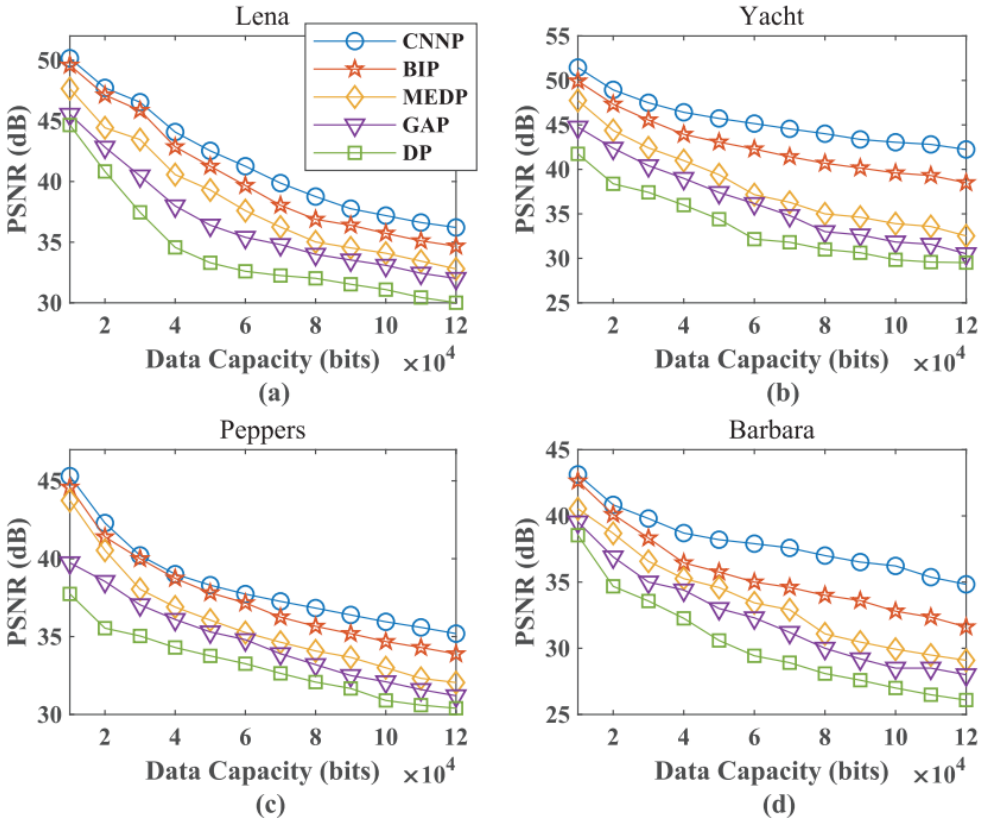


Fig. 7. Comparison of five RDH methods by using four benchmark images.

3. Experimental Results

B. Embedding Performance

prediction error $d_{i,j}$

$$d_{i,j} = u_{i,j} - u'_{i,j}. \tag{2}$$

difference expansion

$$D_{i,j} = 2d_{i,j} + b \tag{3}$$

After data hiding

$$U_{i,j} = D_{i,j} + u'_{i,j}. \tag{4}$$

$$D_{i,j} = \begin{cases} 2d_{i,j} + b, & \text{if } d_{i,j} \in [T_n; T_p] \\ d_{i,j} + T_p + 1, & \text{if } d_{i,j} > T_p \text{ and } T_p \geq 0 \\ d_{i,j} + T_n, & \text{if } d_{i,j} < T_n \text{ and } T_n < 0. \end{cases} \tag{9}$$

3. Experimental Results

B. *Embedding Performance*

TABLE II
AVERAGE PSNR (IN dB) OF 100 IMAGES FOR FIVE DIFFERENT PREDICTORS BY USING EXPANSION EMBEDDING TECHNIQUE IN [8]

bits	CNNP	BIP	MEDP	GAP	DP
10,000	47.9	46.4	44.2	41.7	41.3
20,000	44.7	43.2	40.3	38.2	37.8
30,000	42.4	41.1	37.9	36.0	35.7
40,000	40.9	39.4	36.0	34.1	33.7
50,000	39.9	38.3	35.4	33.5	32.5
60,000	38.7	37.3	34.7	32.6	31.4
70,000	38.0	36.6	33.5	31.7	30.9
80,000	37.3	35.6	32.7	31.2	30.3
90,000	36.5	35.2	32.1	30.7	29.9
100,000	35.9	34.8	31.6	30.2	29.3
110,000	35.3	34.3	31.0	29.5	28.7
120,000	34.7	33.7	29.5	28.7	28.2

TABLE III
AVERAGE PSNR (IN dB) OF 100 IMAGES OF THE PROPOSED CNNP-BASED METHOD AND THE METHOD [11]

bits	CNNP	BIP
10,000	58.4	56.8
20,000	55.1	53.8
30,000	52.9	51.6
40,000	51.2	50.1
50,000	49.8	48.8
60,000	48.5	47.5
70,000	47.3	46.4
80,000	46.3	45.4
90,000	45.5	44.7
100,000	44.8	44.0
110,000	43.9	43.2
120,000	43.0	42.3

4. Conclusion