

Unified entropy-based sorting for reversible data hiding

Jiajia Xu¹ · Weiming Zhang¹ · Ruiqi Jiang¹ ·
Nenghai Yu¹

Received: 19 April 2016 / Revised: 10 September 2016 / Accepted: 16 September 2016 /
Published online: 14 October 2016
© Springer Science+Business Media New York 2016

Abstract Reversible data hiding schemes compete against each other for a sharply distributed prediction error histogram, usually realized by utilizing prediction strategies together with sorting technique that aims to estimate the local context complexity for each pixel to optimize the embedding order. Sorting techniques benefit prediction a lot by picking out pixels located in smooth areas. In this paper, a novel entropy-based sorting (EBS) scheme is proposed for reversible data hiding, which uses entropy measurement to characterize local context complexity for each image pixel. Furthermore, by extending the EBS technique to the two-dimensional case, it shows generalized abilities for multi-dimensional RDH scenarios. Additionally, a new gradient-based tracking and weighting (GBTW) pixel prediction method is introduced to be combined with the EBS technique. Experimental

This work was supported in part by the Natural Science Foundation of China under Grant 61170234 and Grant 61572452, in part by the Strategic Priority Research Program through the Chinese Academy of Sciences under Grant XDA06030601, and in part by the Science and Technology on Information Assurance Laboratory under Grant KJ-13-003.

Copyright (c) 2013 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org

✉ Weiming Zhang
weimingzhang@yahoo.cn; zhangwm@ustc.edu.cn

Jiajia Xu
xujiajia@mail.ustc.edu.cn

Ruiqi Jiang
[jq123@mail.ustc.edu.cn](mailto:jrq123@mail.ustc.edu.cn)

Nenghai Yu
ynh@ustc.edu.cn

¹ CAS Key Laboratory of Electro-magnetic Space Information, University of Science and Technology of China, Hefei 230026, China

results apparently indicate that our proposed method outperforms the previous state-of-arts counterparts significantly in terms of both the prediction accuracy and the overall embedding performance.

Keywords Reversible data hiding · Entropy-based sorting · Gradient-based tracking and weighting

1 Introduction

Reversible data hiding (RDH) [15, 18, 26] as a new embedding technique has drawn much attention in the past few years with advantages that not only the secret message can be precisely extracted, but also the cover itself can be restored losslessly. This important data hiding technique has been found to be useful in many fields such as medical imagery, military imagery and legal imagery, where the cover cannot be damaged during data extraction. A framework of RDH for digital images is illustrated in Fig. 1.

Until now, in order to facilitate the efficiency of RDH, researchers have proposed many methods in the past decade. In general, RDH algorithms consist of embedding and extracting. There are three key steps in the process of embedding. The first step is predicting, which focuses on how to exploit inter-pixel correlations better for deriving a sharp distribution. The second step is sorting technique, which makes the most of the correlation among neighboring pixels for optimizing embedding order. The third step involves reversibly embedding the message into the prediction-error (PE) sequence by modifying its histogram [20, 28–30].

The improvement of the prediction is essential for RDH [3–8, 21]. Some predictors have proposed such as median edge detection (MED) and gradient adjusted prediction (GAP) [21]. The MED is a non-linear predictor that uses three neighboring pixels to estimate unknown pixel. It detects horizontal or vertical edges using three neighboring pixels to estimate the predicted value of the current pixels accordingly. The GAP uses seven neighboring pixels and selects the output based not only on the existence of a horizontal or vertical edge, but also on its strength. Recently, C. Dragoi and D. Coltuc [6, 7] have proposed extended gradient-based selection and weighting (EGBSW) and local prediction (LP) for RDH. The EGBSW algorithm uses a set of four simple linear predictors associated with the four principal directions and computes the output value as a weighted sum between the predicted values corresponding to the selected gradients. Then predicted value is obtained. For the LP [7], a least square predictor is computed on a square block centered on the pixel and the corresponding prediction error is obtained, which results in good performance.

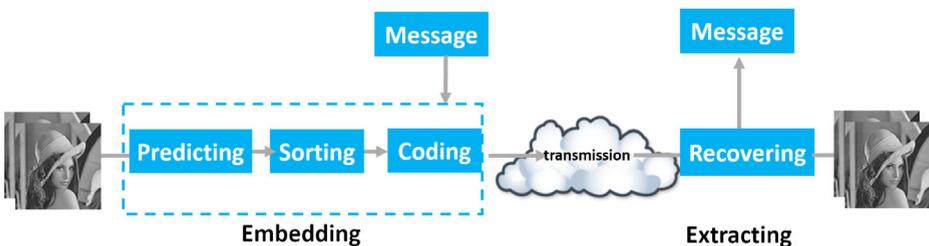


Fig. 1 Framework of RDH embedding/extraction

As a fundamental step to exploit the correlation between neighboring pixels for optimizing embedding order, sorting [1, 23] algorithm is also essential to enhance the embedding payload and visual quality. By introducing sorting algorithm, Kamstra and Heijmans [12] has gained a great improvement in reducing the location map size by sorting pairs of pixels over previous methods. Sachnev et al. [23] has used local variance values to sort the predicted errors. They sort the cells in ascending order of the local variance values, which first embeds the smoother cells with lower local variance values. But in some cases, it does not work accurately. Mahsa Afsharizadeh [1] extend Sachnev's method and proposed a new efficient sorting technique, which generating a more accurate sorting procedure. Ou et al. [19] proposed a simple but efficient sorting method by calculating its local complexity, which is the sum of absolute differences between diagonal blank pixels in the 4×4 sized neighborhood. A small local complexity indicates that the pair is located in a smooth image region where the pixels can be accurately predicted. However, the characteristic of prediction-error distribution was not considered in the above-mentioned algorithms.

In this paper, first of all, we propose a new gradient-based tracking and weighting (GBTW) method based on Rad and Attar's work [7] for predicting, which uses two or four neighboring pixels to estimate the unknown pixel. Secondly, we propose a generalized normal distribution entropy-based sorting for reversible data hiding. Then, we extend the one-dimensional entropy-based sorting for two-dimensional (2D) RDH, and deduce the uniform entropy-based sorting for multi-dimensional RDH.

The paper is organized as follows. Section 2 describes the proposed GBTW algorithm for reversible data hiding. Section 3 deals with the details of generalized normal distribution entropy-based sorting. The experimental results are presented in Section 4. In Section 5, conclusions are briefly drawn based on the experimental results.

2 Prediction using gradient-based tracking and weighting (GBTW)

Naturally, images have an important feature, whose adjacent pixels are highly correlated, which is continuous tone. However, edges also play a critical role in human visual system (HVS), which is revealing with jump in intensity. How to use the property to predict pixel is very important. Rad and Attar [21] have proposed gradient-based tracking and adapting method (GBTW) that tracks edges in more directions precisely. When the main direction and sub direction are the same, the predictor just uses one neighboring pixel for prediction. When they are different, it uses two neighboring pixels. However GBTW did not make full use of the feature of high correlation between adjacent pixels.

All pixels divided into two sets: the shadow pixel set and the blank set. The shadow pixels and blank pixels appear alternately. And the number of pixels in the two sets are equal. In the first round, the shadow set is used for embedding data and blank set for computing predictions. While in the second round, the blank set is used for embedding and shadow set for computing predictions. Henceforth, this scheme will be called the double-layered embedding scheme. So here is a new gradient-based tracking and weighting (GBTW) method for predicting based on GBTW, which using eight neighboring pixels ($\hat{N}W, N, \hat{N}E, E, \hat{S}E, S, \hat{S}W$ and W) to estimate current pixel. Note that the pixel P in shadow set has eight neighboring pixels, where N, E, S, W belong to blank set, and $\hat{N}W, \hat{N}E, \hat{S}E, \hat{S}W$ can be predicted by blank set as follows: $\hat{N}W = (NNW + N + W + NWW)/4$, $\hat{N}E = (NNE + N + E + NEE)/4$, $\hat{S}W = (W + S + W + SSW + SWW)/4$, and $\hat{S}E = (E + SEE + SSE + S)/4$. Only the pixels in blank set are used to predict the pixels in shadow set and embedding is done in shadow set, and blank embedding is similar

to shadow embedding, which can ensure the algorithm reversible. During data hiding in one set, pixels from the other set are not to be modified. The decoding procedure is an inverse of the encoding scheme.

By estimating main direction (denoted by θ_{m-d}) gradients in 4 directions ($45^\circ, 90^\circ, 135^\circ$ and 180°) and sub-direction (denoted by θ_{s-d}) gradients in 11 directions ($\theta_{s-d} = \theta_{m-d} - 45^\circ, \theta_{m-d}$ or $\theta_{m-d} + 45^\circ$) as shown in Fig. 2, it tries to track the most correlation pixels and weight the neighboring pixels according to local gradient to predict current pixel.

GBTW algorithm first computes some parameters to earn the gradient orientation. Main directions θ_{m-d} are computed by using adjacent closest pixels as follows:

$$m - d_{max} = \max\{|P - \hat{N}E|, |P - N|, |P - \hat{N}W|, |P - W|\} \tag{1}$$

$$\theta_{m-d} = \begin{cases} 45^\circ & m - d_{max} = |P - \hat{N}E| \\ 90^\circ & m - d_{max} = |P - N| \\ 135^\circ & m - d_{max} = |P - \hat{N}W| \\ 180^\circ & m - d_{max} = |P - W| \end{cases} \tag{2}$$

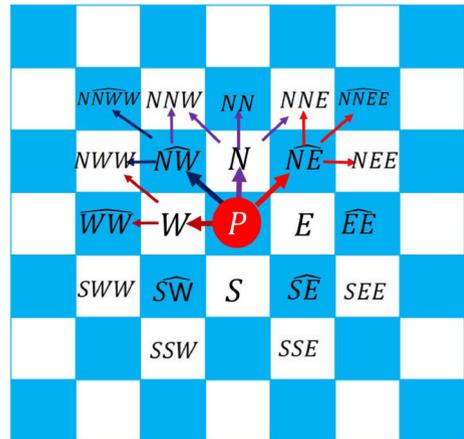
After computing, we can get main direction θ_{m-d} .

Secondly, the sub-direction θ_{s-d} among $\theta_{m-d} - 45^\circ, \theta_{m-d}$ or $\theta_{m-d} + 45^\circ$ is selected, sub-directions are produced to be more exact in determining the edges. As for, how to get sub-direction θ_{s-d} . For instance, in the Fig. 2, from the beginning of the pixel $\hat{N}E$ there are three sub-directions, identification with three red lines. If the absolute $|NNE - \hat{N}E|$ is maximum, then the θ_{s-d} is 45° . In the same way, we can get other θ_{s-d} . In the predicting step, two directions which are main direction and sub-direction in term of maximum change level, are both taken into consideration. That is to say, final direction not only relies on the main direction but also relies on sub-direction. GBTW produces the prediction in a way that the pixels in the orthogonal direction of main-direction have more contribution to the formation of final prediction than the pixels in the orthogonal direction of sub-direction. GBTW uses four of eight neighboring pixels for final predicting.

For instance, as shown in Fig. 3a, θ_{m-d} is 45° and θ_{s-d} is 0° . $\hat{N}W$ and $\hat{S}E$ are vertical to θ_{m-d} , and N and S are vertical to θ_{s-d} . Then the prediction for pixel is computed as:

$$\hat{P} = \frac{[\omega_{m-d} \times (\hat{N}W + \hat{S}E) + \omega_{s-d} \times (N + S)]}{2 \times (\omega_{m-d} + \omega_{s-d})} \tag{3}$$

Fig. 2 Main direction (short of θ_{m-d}) gradients in 4 directions and sub-direction (short of θ_{s-d}) gradients in 11 directions



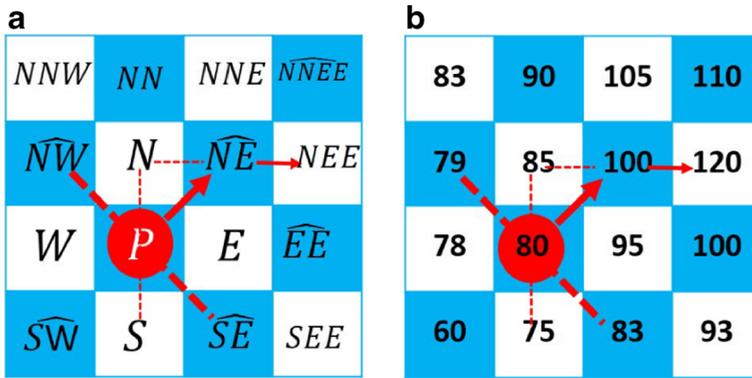


Fig. 3 **a** is main direction (θ_{m-d} is 45°) and sub-direction (θ_{s-d} is 0°). **b** is an example

Where ω_{m-d} is the weight of \widehat{NW} and \widehat{SE} , ω_{s-d} is the weight of N and S . According to the experimental results, we suggest that $\omega_{m-d} : \omega_{s-d}$ is equal to 2:1. We give an example Fig. 3b. The traditional rhombus prediction algorithm is that center pixel P of the cell can be predicted from the four neighboring pixels N, E, S, W . The predicted value \hat{P} is computed as $\hat{P} = (N + E + S + W)/4 = (85 + 95 + 75 + 78)/4 = 83.25$. But the traditional rhombus prediction algorithm does not making full use of image texture characteristics. Suppose that the main direction is 45° and the sub-direction is 0° (see above Fig. 3b), then prediction $\hat{P} = [\omega_{m-d} \times (79 + 83) + \omega_{s-d} \times (85 + 75)]/2 \times (\omega_{m-d} + \omega_{s-d}) = [2\omega_{s-d} \times (79 + 83) + \omega_{s-d} \times (85 + 75)]/2 \times (2\omega_{s-d} + \omega_{s-d}) = 80.67$. Obviously, our algorithm is better than the traditional algorithm.

Generally speaking, the unified formula of prediction is as follows

$$\hat{P} = \frac{1}{2(\omega_{m-d} + \omega_{s-d})} [\omega_{m-d} \times (P_{m-d-orth1} + P_{m-d-orth2}) + \omega_{s-d} \times (P_{s-d-orth1} + P_{s-d-orth2})], \tag{4}$$

where $P_{m-d-orth1}$ and $P_{m-d-orth2}$ are the pixels in the vertical direction to θ_{m-d} , $P_{s-d-orth1}$ and $P_{s-d-orth2}$ are the pixels in the vertical direction to θ_{s-d} . In the same way, we can get other predictors. In detail, all of the predictors about direction are shown in Table 1.

As shown in Table 1, according to the local gradient estimation, the predictor performs dynamically. For each pixel prediction just two or four neighbor pixels participate in final predictor. When the main direction and sub-direction are the same, the predictor estimates the unknown pixel value to be same as the pixel which keeps the intensity change level and just uses two neighbor pixels for prediction. When the main and sub-direction are different, the predictors are much more complex than previous. In these circumstances, four neighbor pixels participate to form the predictor. According to GBTW algorithm premises, the pixels which participate in order to show the effect of main direction in keeping maximum change level weighted ω_{m-d} and the other one which participates for effect of sub-direction weighted ω_{m-d} and divided by $2 \times (\omega_{m-d} + \omega_{s-d})$ to form the final predictor.

Next, the prediction-error is computed by:

$$e = P - \hat{P} \tag{5}$$

Finally, the prediction-error sequence $\mathbf{e} = (e_1, \dots, e_N)$ is derived.

Table 1 All predictors in different directions

Main-direction	Sub-direction	Predictor
$\theta_{m-d} = 45^\circ$	$\theta_{s-d} = 0^\circ$	$\hat{P} = \frac{[\omega_{m-d} \times (\hat{N}W + \hat{S}E) + \omega_{s-d} \times (N+S)]}{2 \times (\omega_{m-d} + \omega_{s-d})}$
$\theta_{m-d} = 45^\circ$	$\theta_{s-d} = 45^\circ$	$\hat{P} = \frac{\hat{N}W + \hat{S}E}{2}$
$\theta_{m-d} = 45^\circ$	$\theta_{s-d} = 90^\circ$	$\hat{P} = \frac{[\omega_{m-d} \times (\hat{N}W + \hat{S}E) + \omega_{s-d} \times (W+E)]}{2 \times (\omega_{m-d} + \omega_{s-d})}$
$\theta_{m-d} = 90^\circ$	$\theta_{s-d} = 45^\circ$	$\hat{P} = \frac{[\omega_{m-d} \times (W+E) + \omega_{s-d} \times (\hat{N}W + \hat{S}E)]}{2 \times (\omega_{m-d} + \omega_{s-d})}$
$\theta_{m-d} = 90^\circ$	$\theta_{s-d} = 90^\circ$	$\hat{P} = \frac{W+E}{2}$
$\theta_{m-d} = 90^\circ$	$\theta_{s-d} = 135^\circ$	$\hat{P} = \frac{[\omega_{m-d} \times (W+E) + \omega_{s-d} \times (\hat{N}E + \hat{S}W)]}{2 \times (\omega_{m-d} + \omega_{s-d})}$
$\theta_{m-d} = 135^\circ$	$\theta_{s-d} = 90^\circ$	$\hat{P} = \frac{[\omega_{m-d} \times (\hat{N}E + \hat{S}W) + \omega_{s-d} \times (W+E)]}{2 \times (\omega_{m-d} + \omega_{s-d})}$
$\theta_{m-d} = 135^\circ$	$\theta_{s-d} = 135^\circ$	$\hat{P} = \frac{\hat{N}E + \hat{S}W}{2}$
$\theta_{m-d} = 135^\circ$	$\theta_{s-d} = 180^\circ$	$\hat{P} = \frac{[\omega_{m-d} \times (\hat{N}E + \hat{S}W) + \omega_{s-d} \times (N+S)]}{2 \times (\omega_{m-d} + \omega_{s-d})}$
$\theta_{m-d} = 180^\circ$	$\theta_{s-d} = 135^\circ$	$\hat{P} = \frac{[\omega_{m-d} \times (N+S) + \omega_{s-d} \times (\hat{N}E + \hat{S}W)]}{2 \times (\omega_{m-d} + \omega_{s-d})}$
$\theta_{m-d} = 180^\circ$	$\theta_{s-d} = 180^\circ$	$\hat{P} = \frac{N+S}{2}$

3 Unified entropy-based sorting

In order to hide more data with less visual degradation, we need to change the order to hide data into the prediction-error. However, how to exploit the correlation among neighboring pixels for optimizing embedding order? In information theory [24], the entropy refers to disorder or uncertainty. So it can be used to characterize the randomness of pixels in a certain image region. For example, if the entropy is higher, which means the pixels in image region are more random or unpredictability. Consequently, the pixels are hard to predict accurately in this region. Thus, the prediction-errors can be rearranged by sorting according to entropy.

3.1 Entropy of generalized normal distribution for sorting

The generalized error distribution is a generalized form of the Normal, possesses a natural multivariate form, has a parametric kurtosis that is unbounded above and possesses special cases that are identical to the Normal and the double exponential distributions [9]. Given that the probability density function (PDF) of prediction-error follows generalized normal distribution or gaussian distribution, we consider using this model to describe the prediction-error in Fig. 3. Generalized normal distribution density function is defined by Nadarajah [17] (Fig. 4).

$$f(e|u, \alpha, \beta) = \frac{\beta}{2\alpha\Gamma(\frac{1}{\beta})} \exp \left\{ - \left| \frac{(e-u)}{\alpha} \right|^\beta \right\}, \tag{6}$$

where e is prediction-error with mean u and variance σ^2 . $\alpha = \sqrt{\sigma^2\Gamma(1/\beta)/\Gamma(3/\beta)}$ is a scale parameter, playing the role of a variance that determines the width of the PDF, while $\beta > 0$, called the shape parameter, controls the fall-off rate in the vicinity of the mode (the higher β , the lower the fall-off rate). $\Gamma(\cdot)$ denotes the Gamma function such that $\Gamma(t) = \int_0^\infty x^{t-1} \exp(-x) dx$. It is easy to see that the (6) reduces to the normal distribution for $\beta = 2$, and Laplacian distribution for $\beta = 1$. Figure 4 illustrates some possible shapes of (6). The effect of the parameters can easily be seen from these graphs. Similar plots can be drawn for others values of the parameters.

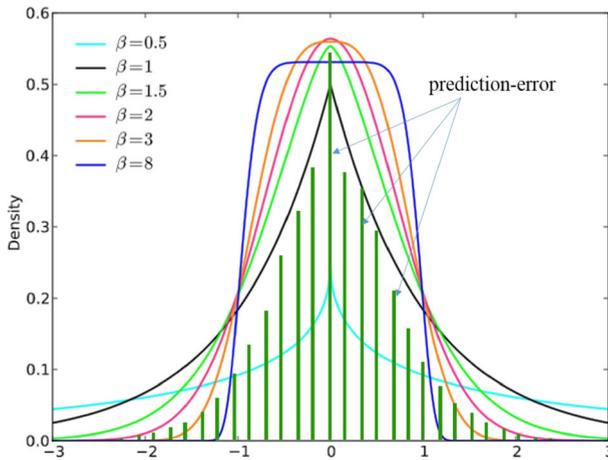


Fig. 4 The probability density plots of generalized normal distributions and prediction-error distribution of lena

An entropy of a random variable e is a measure of variation of the uncertainty. Following Rényi [22], the Rényi entropy is defined by

$$\begin{aligned}
 R(\gamma) &= \frac{1}{1-\gamma} \log_2 \left\{ \int [f(e|u, \alpha, \beta)]^\gamma de \right\} \\
 &= \frac{1}{1-\gamma} \log_2 \left\{ \frac{\beta^\gamma}{(2\alpha)^\gamma (\Gamma(1/\beta))^\gamma} \frac{2\alpha\gamma^{-1/\beta} \Gamma(1/\beta)}{\beta} \right\} \\
 &= \frac{\log_2 \gamma}{\beta(\gamma-1)} - \log_2 \left\{ \frac{\beta}{2\alpha\Gamma(1/\beta)} \right\}, \tag{7}
 \end{aligned}$$

where $\gamma > 0, \gamma \neq 1, \alpha > 0$ and $\beta > 0$. According to Shannon [24], the entropy of the generalized normal random variable e is given by

$$H(e|u, \alpha, \beta) = - \int_{-\infty}^{+\infty} f(e|u, \alpha, \beta) \log_2(f(e|u, \alpha, \beta)) de, \tag{8}$$

which is the particular case of Rényi entropy for $\gamma \Rightarrow 1$. According to L'Hospital's rule, if $\lim_{\gamma \rightarrow c} f(\gamma) = \lim_{\gamma \rightarrow c} g(\gamma) = 0$ or $\pm\infty$, and $\lim_{\gamma \rightarrow c} [f'(\gamma)/g'(\gamma)]$ exists, and $g'(\gamma) \neq 0$ for all γ then $\lim_{\gamma \rightarrow c} [f(\gamma)/g(\gamma)] = \lim_{\gamma \rightarrow c} [f'(\gamma)/g'(\gamma)]$. Thus, in the limit when $\gamma \Rightarrow 1$ and using L'Hospital's rule

$$\lim_{\gamma \rightarrow 1} \frac{\log_2 \gamma}{\beta(\gamma-1)} = \frac{1}{\ln 2} \tag{9}$$

Shannon entropy is easily obtained from the expression for Rényi entropy as follows:

$$H(e|u, \alpha, \beta) = \frac{1}{\beta \ln 2} - \log_2 \left\{ \frac{\beta}{2\alpha\Gamma(1/\beta)} \right\} \tag{10}$$

where $\alpha > 0, \beta > 0$. The entropy depends only on the shape parameter β and the scale parameter α . We assume e follows zero-mean generalized normal distribution. The

generalized normal distribution is symmetric with respect to u , hence the odd central moments are zero. Thus, the n -th moment is given by [2]

$$E\{|e|^n\} = \left\{ \frac{\sigma^2 \Gamma(1/\beta)}{\Gamma(3/\beta)} \right\}^{n/2} \frac{\Gamma((n+1)/\beta)}{\Gamma(1/\beta)} \tag{11}$$

In order to estimate the parameters β and α , we consider the first two absolute moments. So we can get $Ra(\beta)$, which is known as generalized Gaussian function ratio

$$Ra(\beta) = \frac{E^2[|e|]}{E[|e|^2]} = \frac{\Gamma^2(2/\beta)}{\Gamma(1/\beta)\Gamma(3/\beta)} \tag{12}$$

It seems to be clear that the function $Ra(\beta)$ and the inverse function $Ra^{-1}(\beta)$ cannot be inverted in an explicit form. Hence, we propose a numeric-analytic or approximation procedure to solve the problem. We should notice that $Ra(\beta)$ is a function of products of gamma functions with arguments depending on β . Hence, the Stirling approximation [2] is well-behaved for values close to the origin, which is $\Gamma(x) = \sqrt{2\pi}x^{x-\frac{1}{2}}e^{-x}[1 + O(x^{-1})]$. Then, we can obtain the following approximate equation

$$\Gamma(x) \simeq \tilde{\Gamma}(x) = \sqrt{2\pi}x^{x-\frac{1}{2}}e^{-x} \tag{13}$$

Then, by taking the above approximate equation, we propose an approximation such that it can be inverted and close enough to the actual function as follow

$$Ra(\beta) \simeq \tilde{Ra}(\beta) = \frac{\tilde{\Gamma}^2(2/\beta)}{\tilde{\Gamma}(1/\beta)\tilde{\Gamma}(3/\beta)} = (1/4)^{3^{\frac{1}{2}} \frac{\beta-6}{\beta} 2^{\frac{4+\beta}{\beta}}} \tag{14}$$

The corresponding inverse function for $Ra^{-1}(x)$ is given by

$$\tilde{Ra}^{-1}(x) = \frac{4 - 3 \log_2 3}{\log_2(\log_2(x)) - \frac{1}{2} \log_2 3} \tag{15}$$

from which it is possible to find an approximated method of moments estimator for β and α .

$$\hat{\beta} = \tilde{Ra}^{-1} \left\{ \frac{E^2[|e|]}{E[|e|^2]} \right\} \tag{16}$$

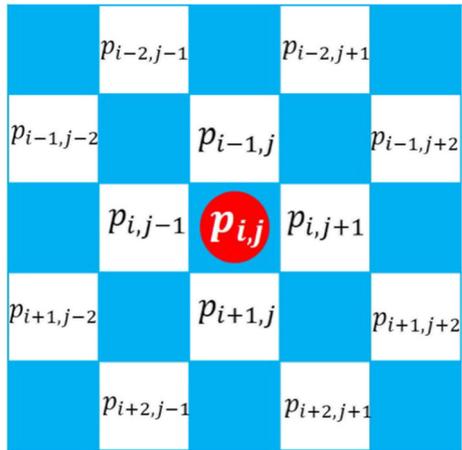
$$\hat{\alpha} = \sqrt{\frac{\sigma^2 \Gamma(1/\hat{\beta})}{\Gamma(3/\hat{\beta})}} \tag{17}$$

Now, we discuss how to utilize the entropy-based sorting in RDH. Sorting is possible only when cells are independent. That is to say, embedding data into one cell should not affect the other cells. Note that the shadow pixel set and the blank set are independent each other. In the first round, the shadow set is used for embedding data and blank set for sorting. While in the second round, the blank set is used for embedding and shadow set for sorting. Specific details are as follows, we can exploit the correlation between neighboring pixels to estimate the value of u in Fig. 5.

$$e_{i-1,j} = \frac{p_{i-2,j-1} + p_{i-2,j+1} + p_{i,j-1} + p_{i,j+1}}{4} - p_{i-1,j}$$

$$e_{i,j+1} = \frac{p_{i-1,j} + p_{i-1,j+2} + p_{i+1,j} + p_{i+1,j+2}}{4} - p_{i,j+1} \tag{18}$$

Fig. 5 5×5 template for the estimation of the mean and variance



According to the Fig. 5, we can calculate $e_{i+1,j}$, $e_{i,j-1}$, $e_{i-2,j-1}$, $e_{i-2,j+1}$, $e_{i-1,j+2}$, $e_{i+1,j+2}$, $e_{i+2,j+1}$, $e_{i+2,j-1}$, $e_{i+1,j-2}$ and $e_{i-1,j-2}$ in the same way. Based on the above, we can get the absolute central moments by

$$E[|e|] = \frac{1}{12} \sum_{n,m} |e_{i+n,i+m}| \tag{19}$$

$$E[|e|^2] = \frac{1}{12} \sum_{n,m} |e_{i+n,i+m}|^2 \tag{20}$$

where $n, m \in [-2, -1, 1, 2]$. Then, by using above equation, we can get $\hat{\beta}$ and $\hat{\alpha}$.

Based on information theory, the entropy of prediction-error in cell (5×5) region can be given by

$$H(e|u, \hat{\alpha}, \hat{\beta}) = \frac{1}{\hat{\beta} \ln 2} - \log_2 \left\{ \frac{\hat{\beta}}{2\hat{\alpha}\Gamma(1/\hat{\beta})} \right\} \tag{21}$$

Further, consider a much bigger region **region** ($\geq 5 \times 5$ pixels) of the image in Fig. 5. The entropy of prediction-error can be estimated as:

$$H_{\text{region}}(e|u, \hat{\alpha}, \hat{\beta}) = \sum_{\text{region}} H(e|u, \hat{\alpha}, \hat{\beta}) \tag{22}$$

Observed from (21) and (22), named as one-dimensional entropy-based sorting (EBS). The EBS has several features. First of all, this value of entropy remains unchanged after data hiding. Second, if the value of entropy is much lower, which means the pixels in a smooth image region or regularity region, and it can be predicted accurately. That is to say, entropy with smaller variance values are better for data hiding. Consequently, the region should be used preferentially for data embedding.

By setting a threshold λ , the entropy satisfying $H(e|u, \hat{\alpha}, \hat{\beta}) \leq \lambda$ are utilized in data embedding while the others are skipped. For a specic payload R , λ is determined as the smallest value such that it can ensure the enough payload. So, threshold is determined by the specific payload. In the experiment, we firstly set a very small initial threshold λ . If the prediction-error sequence, which meet the condition $H(e|u, \hat{\alpha}, \hat{\beta}) \leq \lambda$, can not accommodate the payload. Then the threshold λ is increased, until the payload is completely

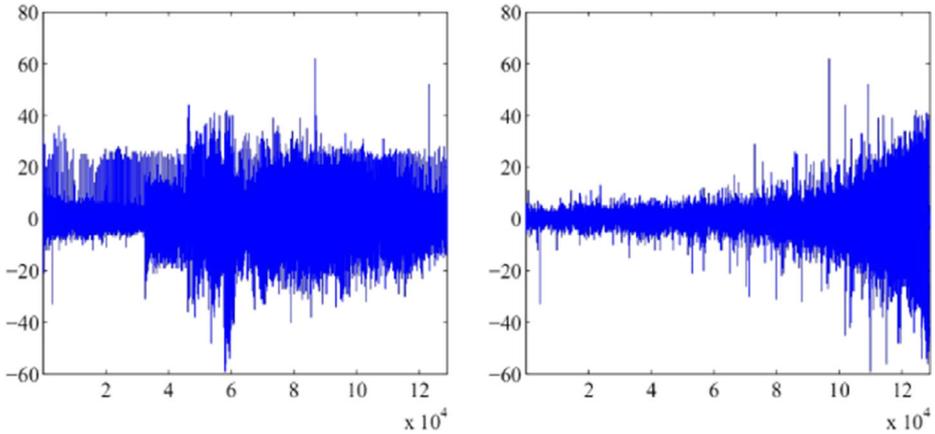


Fig. 6 The prediction-error of lena

embedded. Prediction-errors are sorted in ascending order of the entropy. Thus, the embedding process starts from the prediction-error with the smallest entropy value in the sorted row, and moves on to the next prediction-error until the last bit of data is embedded. As shown in Fig. 6, the left is the prediction-error of the lena image before sorting and the error margin is very high. The right is sorted by our EBS and the results can be clearly seen that both error and entropy are small being sorted in front. The image quality can be improved significantly, because the message is embedded in the appropriate prediction-error.

3.2 Entropy of multivariate normal distribution model

The prior reversible data hiding (RDH) methods are 1-dimension. Ou et al. [19] propose 2-dimensions RDH, which take every two adjacent prediction-errors as a unit to generate a sequence consisting of prediction-error pairs, then obtain a 2-dimensional prediction-error histogram (2D PEH), and finally, embed data by using pairwise prediction-error expansion(PEE). Ou et al. [19] have proposed scheme outperforms state-of-the-art 1-dimension RDH algorithms. In this paper, we propose a 1-dimension entropy-based sorting for reversible data hiding, which uses entropy measurement to characterize local context complexity for optimizing embedding order, and can significantly improve the performance of RHD. Then, we extend the 1-dimension entropy-based sorting to 2-dimensions entropy-based sorting, which can improve the performance of 2-dimensions RDH. Furthermore, we deduce the uniform d-dimensions entropy-based sorting framework for d-dimensions RDH scenarios.

In this section, we deduce the uniform entropy-based sorting for d -dimensional RDH. Suppose that prediction-error $\mathbf{e} = (e_1, e_2, \dots, e_d)'$ follows the d -dimensional generalized normal distribution, and the density function of the multivariate generalized normal distribution is defined by [17]

$$\begin{aligned}
 f(\mathbf{e}|\mathbf{u}, \Sigma, \beta) &= \frac{\Gamma(\frac{d}{2})}{\pi^{\frac{d}{2}} \Gamma(\frac{d}{2\beta}) 2^{\frac{d}{2\beta}} |\Sigma|^{\frac{1}{2}}} \frac{\beta}{\beta} \\
 &\times \exp \left\{ -\frac{1}{2} [(\mathbf{e} - \mathbf{u})' \Sigma^{-1} (\mathbf{e} - \mathbf{u})]^\beta \right\} \tag{23}
 \end{aligned}$$

where β is the shape parameter, with mean u and covariance matrix $(2^{1/\beta} \Gamma(\frac{m+2}{2\beta}) \Sigma) / (m \Gamma(\frac{m}{2\beta}))$. In addition, noting that the covariance matrix is given by Σ only in the case of the Gaussian distribution ($\beta = 1$). According to the formula of entropy, there does not appear to exist a closed form expression for the entropy of multivariate generalized normal distribution. However, in order to simplify and reduce the complexity of the problem, we take $\beta = 1$ as a particular case of multivariate generalized normal. When $\beta = 1$ is corresponding to the multivariate normal distribution

$$f(\mathbf{e}|\mathbf{u}, \Sigma) = 2\pi^{-\frac{d}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{e} - \mathbf{u})' \Sigma^{-1}(\mathbf{e} - \mathbf{u})\right) \tag{24}$$

where the mean is given by

$$\mathbf{u} = (u_1, u_2, \dots, u_d)' \tag{25}$$

and the covariance is defined by

$$Cov(\mathbf{e}, \mathbf{e}) = E[\mathbf{e} - \mathbf{u}][\mathbf{e} - \mathbf{u}]' = \Sigma \tag{26}$$

$$\begin{aligned} \Sigma &= E[(\mathbf{e} - \mathbf{u})(\mathbf{e} - \mathbf{u})'] \\ &= \begin{pmatrix} c_{1,1} & \dots & c_{1,i} & \dots & c_{1,d} \\ \dots & \dots & \dots & \dots & \dots \\ c_{i,1} & \dots & c_{i,i} & \dots & c_{i,d} \\ \dots & \dots & \dots & \dots & \dots \\ c_{d,1} & \dots & c_{d,i} & \dots & c_{d,d} \end{pmatrix} \end{aligned} \tag{27}$$

In particular, $c_{i,j} > 0$. When $d = 1$, $c_{1,1} = \sigma^2$ in the usual notation. The symmetric matrix $\Sigma := (c_{i,j})$ is positive-definite, since the matrix $(\langle x_i, x_j \rangle)$ is positive-definite for any finite set of linearly independent x_i in a real inner product space $(X, \langle \cdot, \cdot \rangle)$.

$$\begin{aligned} \Sigma^{-1} &= (E[(\mathbf{e} - \mathbf{u})(\mathbf{e} - \mathbf{u})'])^{-1} \\ &= \begin{pmatrix} \alpha_{1,1} & \dots & \alpha_{1,i} & \dots & \alpha_{1,d} \\ \dots & \dots & \dots & \dots & \dots \\ \alpha_{i,1} & \dots & \alpha_{i,i} & \dots & \alpha_{i,d} \\ \dots & \dots & \dots & \dots & \dots \\ \alpha_{d,1} & \dots & \alpha_{d,i} & \dots & \alpha_{d,d} \end{pmatrix} \end{aligned} \tag{28}$$

we have $c_{:,i} = E[(e_i - u_i)(\mathbf{e} - \mathbf{u})']$ and $\alpha_{j,:} \cdot c_{:,i} = \delta_{i,j}$.

So the entropy of multivariate normal can be calculated as

$$\begin{aligned} H(\mathbf{e}|\mathbf{u}, \Sigma) &= - \int_{-\infty}^{+\infty} \dots \int_{-\infty}^{+\infty} f(\mathbf{e}|\mathbf{u}, \Sigma) \log_2(f(\mathbf{e}|\mathbf{u}, \Sigma)) d\mathbf{e} \\ &= - \int f(\mathbf{e}|\mathbf{u}, \Sigma) \left[\frac{1}{2} \log_2(2\pi)^d |\Sigma| \right. \\ &\quad \left. + \log_2 \epsilon \cdot \left(-\frac{1}{2}(\mathbf{e} - \mathbf{u})' \Sigma^{-1}(\mathbf{e} - \mathbf{u}) \right) \right] d\mathbf{e} \\ &= \frac{1}{2} \log_2(2\pi)^d |\Sigma| + \frac{1}{2} \log_2 \epsilon \cdot E[(\mathbf{e} - \mathbf{u})' \Sigma^{-1}(\mathbf{e} - \mathbf{u})] \\ &= \frac{1}{2} \log_2(2\pi)^d |\Sigma| + \frac{1}{2} \log_2 \epsilon \cdot d \\ &= \frac{1}{2} \log_2[(2\pi\epsilon)^d |\Sigma|] \end{aligned} \tag{29}$$

Because $\mathbf{e} = (e_1, e_2, \dots, e_d)'$ is a random vector, and $\Sigma = E[(\mathbf{e} - \mathbf{u})(\mathbf{e} - \mathbf{u})']$. Then we proof $E[(\mathbf{e} - \mathbf{u})'\Sigma^{-1}(\mathbf{e} - \mathbf{u})] = d$ as follows

$$\begin{aligned}
 (\mathbf{e} - \mathbf{u})'\Sigma^{-1}(\mathbf{e} - \mathbf{u}) &= (\mathbf{e} - \mathbf{u})' \begin{pmatrix} \alpha_{1,:} \\ \alpha_{j,:} \\ \dots \\ \alpha_{d,:} \end{pmatrix} (\mathbf{e} - \mathbf{u}) \\
 &= (e_1 - u_1)\alpha_{1,:} \cdot (\mathbf{e} - \mathbf{u}) \\
 &\quad + (e_2 - u_2)\alpha_{2,:} \cdot (\mathbf{e} - \mathbf{u}) + \dots \\
 &\quad + (e_d - u_d)\alpha_{d,:} \cdot (\mathbf{e} - \mathbf{u})
 \end{aligned} \tag{30}$$

$$\begin{aligned}
 E[(\mathbf{e} - \mathbf{u})'\Sigma^{-1}(\mathbf{e} - \mathbf{u})] &= \alpha_{1,:} \cdot E[(e_1 - u_1)(\mathbf{e} - \mathbf{u})] \\
 &\quad + \alpha_{2,:} \cdot E[(e_2 - u_2)(\mathbf{e} - \mathbf{u})] + \dots \\
 &\quad + \alpha_{d,:} \cdot E[(e_d - u_d)(\mathbf{e} - \mathbf{u})] \\
 &= \alpha_{1,:} \cdot c_{:,1} + \alpha_{2,:} \cdot c_{:,2} \dots + \alpha_{d,:} \cdot c_{:,d} \\
 &= d
 \end{aligned} \tag{31}$$

Then we get the entropy of multivariate normal distribution, which fit for d -dimensional RDH.

Now we use a simple 2-dimensional RDH example to illustrate the d -dimensional processes of the method described above. Ou et al. [19] have proposed pairwise prediction-error expansion technique used for 2-dimensional RDH, by considering every two adjacent prediction-errors together $\{(e_{1,1}, e_{2,2}), \dots, (e_{i,j}, e_{i+1,j+1}), \dots\}$. In this case, 1-dimensional entropy-based sorting can not work well, so we need to design sorting algorithm for 2-dimensional RDH. For example, considering this situation, the entropy of $e_{i,j}$ is high, but the entropy of $e_{i+1,j+1}$ is low. If we use 1-dimensional entropy-based sorting to determine whether to embed, the result is $e_{i+1,j+1}$ suitable for embedding rather than $e_{i,j}$. But it does not fit well with algorithm proposed in paper [19]. Consequently, we take every two adjacent prediction-errors $e_{i,j}$ and $e_{i+1,j+1}$ as a whole. The entropy of $e_{i,j}$ and $e_{i+1,j+1}$ are both low having priority. In fact, adjacent prediction-errors are usually highly correlated. And the density function of 2-dimensional normal distribution can be given by

$$f(\mathbf{e}|\mathbf{u}, \Sigma) = 2\pi^{-1} |\Sigma|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{e} - \mathbf{u})'\Sigma^{-1}(\mathbf{e} - \mathbf{u})\right) \tag{32}$$

Where $e = (e_{i,j}, e_{i+1,j+1})'$, $u = (u_1, u_2)'$, Σ is covariance.

$$u_1 = \frac{e_{i-1,j} + e_{i,j+1} + e_{i+1,j} + e_{i,j-1}}{4} \tag{33}$$

$$u_2 = \frac{e_{i,j+1} + e_{i+1,j+2} + e_{i+2,j+1} + e_{i+1,j}}{4} \tag{34}$$

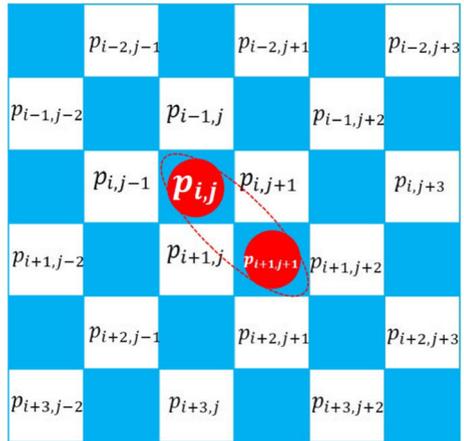
$$e_{i-1,j} = \frac{p_{i-2,j-1} + p_{i-2,j+1} + p_{i,j-1} + p_{i,j+1}}{4} - p_{i-1,j} \tag{35}$$

$$e_{i+2,j+1} = \frac{p_{i+1,j} + p_{i+1,j+2} + p_{i+3,j} + p_{i+3,j+2}}{4} - p_{i+2,j+1} \tag{36}$$

Then we can calculate $e_{i,j-1}$, $e_{i,j+1}$, $e_{i+1,j}$ and $e_{i+1,j+2}$ in the same way in Fig. 7.

$$\Sigma = \begin{pmatrix} c_{1,1} & c_{1,2} \\ c_{2,1} & c_{2,2} \end{pmatrix} = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix} \tag{37}$$

Fig. 7 6×6 template for the estimation of the mean and variance



So the differential entropy of 2-dimensional normal distribution can be given by

$$\begin{aligned}
 H(\mathbf{e}|\mathbf{u}, \Sigma) &= \frac{1}{2} \log_2(2\pi\epsilon)^2|\Sigma| \\
 &= \frac{1}{2} \log_2[(2\pi\epsilon)^2\sigma_1^2\sigma_2^2(1 - \rho^2)] \tag{38}
 \end{aligned}$$

Further, consider a much bigger region $\mathbf{R} (\geq 6 \times 6$ pixels) of the image. The entropy of prediction-error can be estimated as:

$$H_{\mathbf{R}}(\mathbf{e}|\mathbf{u}, \Sigma) = \sum_{\mathbf{R}} \frac{1}{2} \log_2[(2\pi\epsilon)^2\sigma_1^2\sigma_2^2(1 - \rho^2)] \tag{39}$$

where, ρ is the correlation within two adjacent prediction-errors e_i and e_{i+1} , which is computed by

$$\rho = \frac{|\sum_{i=1}^{N-1} (e_i - \bar{e})(e_{i+1} - \bar{e})|}{\sqrt{(\sum_{i=1}^{N-1} (e_i - \bar{e})^2)}\sqrt{(\sum_{i=1}^{N-1} (e_{i+1} - \bar{e})^2)}} \tag{40}$$

where \bar{e} is the mean of $e = (e_1, \dots, e_N)$, and ρ ranges from 0 to 1, and the larger the value, the more correlated the two prediction-errors.

In 2-dimensional RDH example, a novel way is presented in this paper to determine whether a pairwise should be embedded to calculate its local entropy. We take every two adjacent prediction-errors as a unit to generate two-dimensional differential entropy for sorting. A small $H_{\mathbf{R}}(\mathbf{e}|\mathbf{u}, \Sigma)$ indicates that the 2-dimensional prediction-error is located in a smooth image region and should be used preferentially for data embedding. The simplicity and efficiency are the main advantages of entropy-based sorting strategy. The EBS technique also work well in d -dimensional RDH scenarios, so we call it unified entropy-based sorting.

4 Application, experiment and analysis

In this section, we apply GBTW and EBS algorithms to the Sachnev et al. [23], Ou et al. [19] and Zhang et al. [31] method. It is stressed that the embedding and extraction procedures are

same with the algorithms in [19, 23, 31]. We just replace or add the prediction and sorting algorithm in experiments. Then experimental results of the proposed GBTW and EBS for RDH scheme are presented.

4.1 The GBTW predicting and EBS for Sachnev et al. [23]

Double-layered embedding method have been proposed by Sachnev et al. [23], with all pixels divided into two sets: the shadow pixel set and the blank set (see Fig. 8). In the first round, the shadow set is used for embedding data and blank set for computing predictions. While in the second round, the blank set is used for embedding and shadow set for computing predictions. Since the two layers embedding processes are similar in nature, we only take the shadow layer for illustration.

We implemented these methods on the computer with Intel core i3 and 4 GB RAM. The program developing environment is MATLAB R2011b based on Microsoft Windows 7 operating system. In the experiment, in order to simplify the complexity of GBTW, we that $\omega_{m-d} : \omega_{s-d} = 2 : 1$. The size of R region is 5×5 pixels. Test image is shown in Fig. 9.

Our method is being compared with other six recent works of Sachnev et al. [23], Hu et al. [11], Luo et al. [16], Wang et al. [27], Lee et al. [13], Thodi et al. (P3) [25]. The comparison results are shown in Fig. 10a and b. For our method, we vary the embedding rate from 0.1 BPP to 0.6 BPP with step size 0.1. From the Fig. 10a and b, one observes that our proposed method outperforms the previous state-of-arts counterparts significantly. Our algorithm consists of two parts. One is the prediction algorithm, and the other is the sorting algorithm. In order to demonstrate the performance of our GBTW prediction algorithm and EBS sorting algorithm, we design contrast experiment as shown in Fig. 10c and d. The combination of our GBTW prediction algorithm and EBS sorting algorithm with Sachnev et al. [23], there will be four cases as Sachnev+EBS, Sachnev+GBTW, Sachnev+GBTW+EBS (short for proposed I). Observing from Fig. 10c and d, we can see that if embedding rate is smaller, the effect achieved by EBS algorithm is more obvious. For Sachnev+EBS and Sachnev+GBTW method, better results are earned compared with the Sachnev et al. [23] method, respectively, in Fig. 10c and d. Both of these two cases indicate that the GBTW and EBS strategy for RDH are efficient. Compared with Sachnev et al. [23] method, the Sachnev+GBTW+EBS (proposed I) method has a higher PSNR about 3.5 dB to 5 dB on average, especially under low embedding rates.

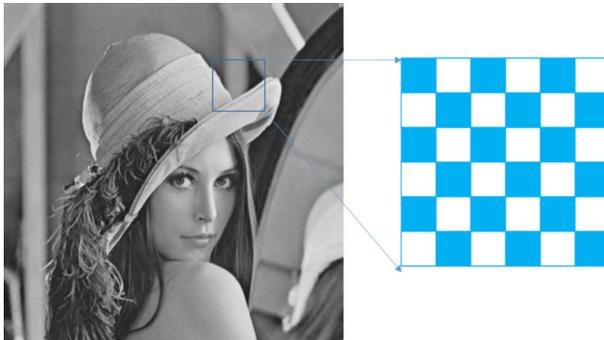


Fig. 8 The image divided into two sets: the shadow pixel set and the blank set



Fig. 9 Test image Lena, Boat, Baboon, Pepper, Barbara and Tiffany

4.2 Recursive histogram modification using GBTW and EBS in RDH

Zhang et al. [31] has proposed a histogram modification method for RDH, which embeds the message by recursively utilizing the decompression and compression processes of an entropy coder. In this paper, we add the GBTW and EBS in Zhang et al. [31] algorithm.

Embedding Procedures

- Firstly, use the GBTW prediction strategy in Section 2, and a prediction-error sequence $\mathbf{e} = \{e_1, \dots, e_i, \dots, e_d\}$ is obtained from the cover.
- Secondly, for each e_i , using 1-dimensional normal distribution entropy-based sorting in Section 3, calculate its entropy $H_{\mathbf{R}}(e|u, \sigma)$, find the smallest value λ , so that there are enough pairs to embed the payload.
- Thirdly, according to the OHM [10, 22] data embedding step, process the prediction-error pairs satisfying $H_{\mathbf{R}}(e|u, \sigma) < \lambda$ to embed the payload. Then the message can be optimally embedded in the marked sequence.

Extracting Procedures

- Firstly, we use the GBTW prediction and scan order strategy to obtain the marked prediction-error $e = \{e'_1, \dots, e'_i, \dots, e'_N\}$.
- Secondly, for each e'_i , we compute its normal distribution entropy $H_{\mathbf{R}}(e'|u, \sigma)$. If another other is the same as the one used in the embedding phase, calculating its entropy.
- Finally, processing the pairs that satisfying $H_{\mathbf{R}}(e'|u, \sigma) < \lambda$, and according to the OHM [22] data extraction step to recover the original image.

Test image is shown in Fig. 9. Observing from Fig. 11, one can find our method is being compared with the other six recent works of Zhang et al. [31], Hu et al. [11], Luo et al. [16], Wang et al. [27], Lee et al. [13], Thodi et al. (P3) [25]. The comparison results are shown in Fig. 11a and b. The combination of our GBTW prediction algorithm and

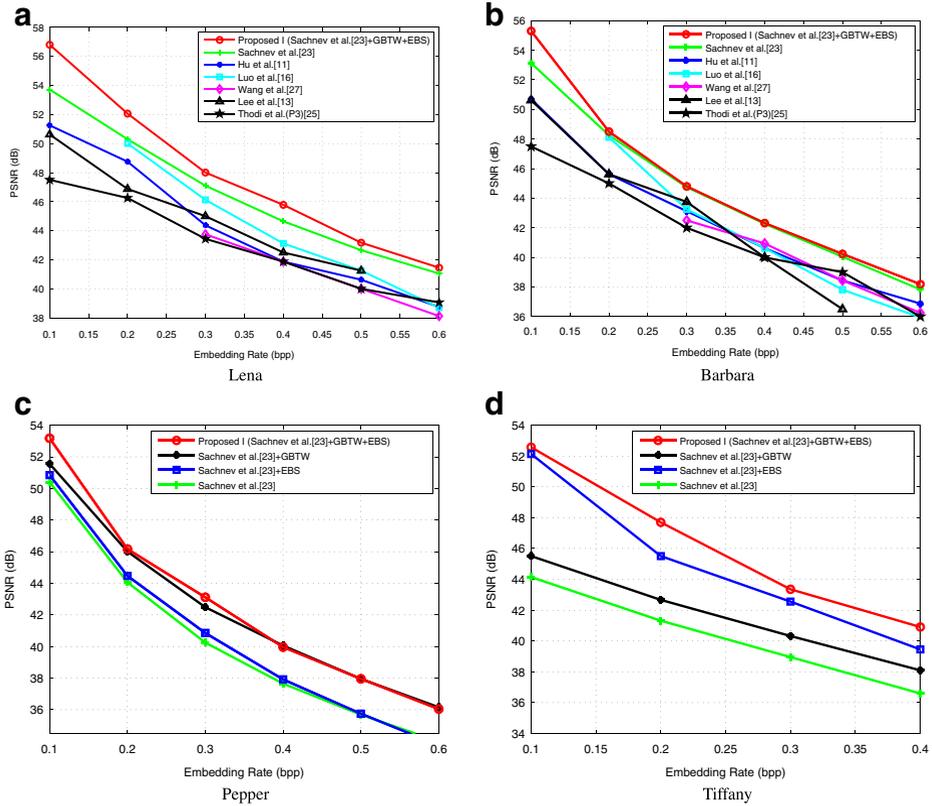


Fig. 10 **a** and **b** is performance comparison between our method and six methods of Sachnev et al. [23], Hu et al. [11], Luo et al. [16], Wang et al. [27], Lee et al. [13], Thodi et al. (P3) [25]. **c**, **d** is embedding performance comparisons with Sachnev et al. [23], Sachnev+GBTW, Sachnev+GBTW+EBS, proposed I (Sachnev+GBTW+EBS)

EBS sorting algorithm with Zhang et al. [31], there will be four cases as Zhang +EBS, Zhang+GBTW, Zhang+GBTW+EBS (short for proposed II). For Zhang+GBTW+EBS method, an average 1.06 dB PSNR gains is earned compared with the Zhang et al. [31] method, and compared with Sachnev et al. [23], the gains of PSNR is much higher, about 2.5 dB to 3.5 dB on average, especially under high embedding rates. For Zhang+EBS and Zhang+GBTW method, an average about 0.3 dB and 0.6 dB PSNR gains is earned compared with the Zhang et al. [31] method, respectively, in Fig. 11c and d. Both of these two cases indicate that, to some extent, the GBTW and EBS strategy for RDH are efficient.

4.3 Entropy-based sorting for pairwise prediction-error expansion RDH [19]

Now we will describe in detail the embedding and extraction procedures. The shadow and blank layers are embedded equally. Notice that here, it is the same as the method of Ou et al. [19].

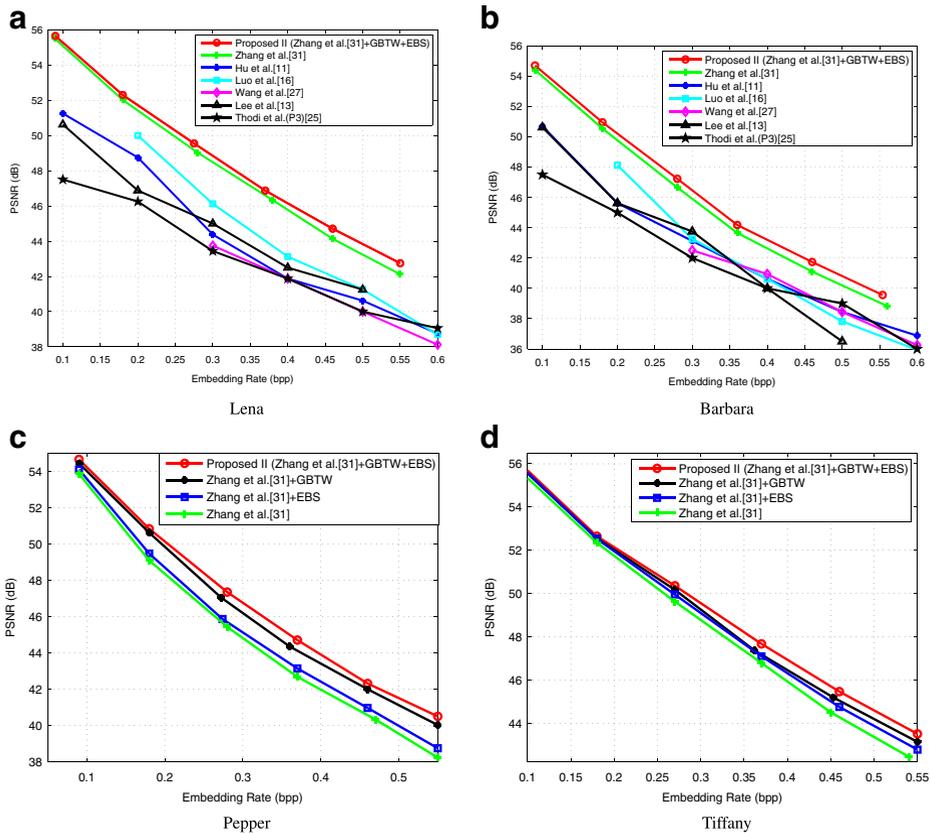


Fig. 11 **a** and **b** is performance comparison between our method and six methods of Zhang et al. [31], Hu et al. [11], Luo et al. [16], Wang et al. [27], Lee et al. [13], Thodi et al. (P3) [25]. **c**, **d**, **e** and **f** is embedding performance comparisons with Zhang et al. [31], Zhang+EBS, Zhang+GBTW, proposed II (Zhang+GBTW+EBS)

Embedding Procedures

- Firstly, predicting shadow pixels in the scan order by GBTW, then we can get the prediction-error pair sequence $(e_{i,j}, e_{i+1,j+1})$ as mentioned above. For each $(e_{i,j}, e_{i+1,j+1})$, we calculate its entropy $H_{\mathbf{R}}(\mathbf{e}|\mathbf{u}, \Sigma)$.
- Secondly, we empty LSBs of some first-line pixels to make room for the embedding of the parameters. Find the smallest value λ , so that there are enough pairs to embed the payload. Using LSB replacement, embed the values of λ , the compressed location map size and the message size into LSBs of some first-line pixels.
- Thirdly, processing the prediction-error pairs that satisfying $H_{\mathbf{R}}(\mathbf{e}|\mathbf{u}, \Sigma) < \lambda$ to embed the payload. The modifications on these pairs are based on the proposed pairwise PEE. After this step, the shadow layer embedding is completed.

Table 2 Experimental results

Methods	Embedding bits	Ou [19]	Ou [19] +EBS	Ou [19] +GBTW	Ou [19] +GBTW +EBS
Lena	10000	59.75	61.05	59.75	60.24
Lake	10000	58.72	58.65	58.64	58.98
Pepper	10000	56.22	58.26	57.40	58.56
Cornfield	10000	61.55	62.77	61.87	62.31

Extracting Procedures

- Firstly, by reading LSBs of some first-line pixels, determining the values of the parameters. Using the same prediction and scanning order to obtain the marked prediction-error pairs sequence $(e'_{i,j}, e'_{i+1,j+1})$. For each $(e'_{i,j}, e'_{i+1,j+1})$, computing its entropy which is the same as the one used in the embedding phase.
- Secondly, processing the pairs that satisfying $H_{\mathbf{R}}(\mathbf{e}'|\mathbf{u}, \Sigma) < \lambda$, we can get entropy. The recovery of these pairs is implemented by the inverse mapping of the proposed pairwise PEE. After the embedded payload is extracted, the location map and replaced LSBs can be obtained.
- Finally, recovering the first-line pixels by the extracted LSBs to recover the original shadow pixels.

The experiment, as shown in Table 2, is enforced on four standard 512×512 sized gray-scale images: Lena, Lake, pepper and cornfield as shown in Fig. 12. For comparison, only low payload cases are considered. That is, the embedding size is 10,000 bits. It should be noted that 10,000 bits are not maximum capacity of our method. The reason is that the

**Fig. 12** Test image Lena, Lake, pepper and cornfield

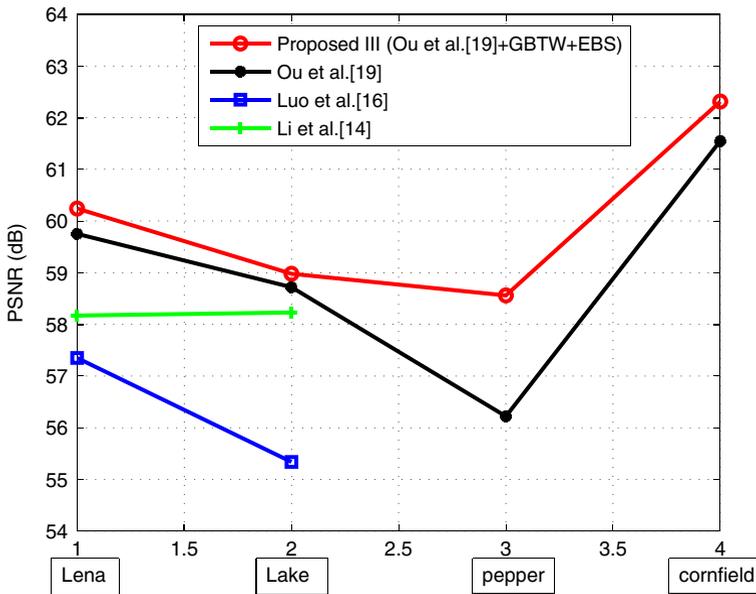


Fig. 13 Embedding performance comparisons with Proposed III (Ou et al. [19] +GBTW+EBS), Ou et al. [19], Luo et al. [16], Li et al. [14]

payload of contrast method Ou et al. [19] is about 50000 bits. So, we choose the same embedding rate to compare the performance of different algorithms. According to Table 2, one can see that the proposed method outperforms these state-of-the-art works. Our method can provide a larger PSNR whatever the test image or payload is. Moreover, experimental results show that proposed III (Ou et al. [19] +GBTW+EBS) method provides an average increase in PSNR of 0.49 dB for Lena, 0.26 dB for Lake, 2.34 dB for Pepper and 0.76 dB for Cornfield. Our method gains 0.96 dB PSNR in average, compared with Ou et al. [19] in Fig. 13. For the test image Lena and Lake, proposed III method yields a superior performance to Luo et al. [16] and Li et al. [14]. Compared the existing embedding techniques, the technique proposed in this paper has a more solid theoretical model. There are two features in our method. Firstly, the sharply distributed prediction error histogram can be obtained by our proposed GBTW prediction method. The sharper the histogram is, the more payload that can be embedded. Therefore, our proposed method have higher payloads. Secondly, the set of sorted prediction errors by our EBS method can be efficiently used for low distortion data hiding. Our method exploited over the sorted prediction errors produces excellent ratio between capacity and distortion.

5 Conclusion

In this paper, we propose a normal distribution entropy-based sorting for reversible data hiding, which uses entropy to characterize the texture of image. Lower entropy means a pixel lies in a smooth image region or regularity region, thus the prediction-error is more accurate. Consequently, the region should be used preferentially for data embedding. Then, we extend the one-dimensional entropy-based sorting to 2-dimensional (2D) RDH, and deduce

the uniform entropy-based sorting for multi-dimensional RDH. Compared with other existing sorting methods, the proposed scheme can achieve higher accuracy of sorting and adapt to different dimensions embedding algorithms. So entropy-based sorting is the key step to enhance the embedding payload and visual quality. Secondly, we propose a new gradient-based tracking and weighting (GBTW) method for predicting, which uses eight neighboring pixels to estimate unknown pixel. It's a highly efficient adaptive predictor. Experimental results show that the proposed method has better results compared to the methods of Sachnev et al. [23], Hu et al. [11], Luo et al. [16], Wang et al. [27], Lee et al. [13], Thodi et al. (P3) [25].

References

1. Afsharizadeh M, Mohammadi M (2013) A reversible watermarking prediction based scheme using a new sorting and technique. In: 10th international conference on information security and cryptology (ISC), pp 98–104
2. Armando Domínguez-Molina J, González-Farías G A practical procedure to estimate the shape parameter in the generalized Gaussian distribution Cimat tech rep.I-01-08-eng.pdf.[Online]<http://www.cimat.mx/reportes/enlinea/I-01-18-eng.pdf>
3. Coltuc D (2012) Low distortion transform for reversible watermarking. *IEEE Trans Image Process* 21(1):412–417
4. Coltuc D, Dragoi I-C (2013) Context embedding for raster-scan rhombus based reversible watermarking. In: Proceedings of the ACM IH&MMSEC, pp 215–220
5. Dragoi C, Coltuc D (2012) Improved rhombus interpolation for reversible watermarking by difference expansion. In: Proceedings of the EUSIPCO, pp 1688–1692
6. Dragoi C, Coltuc D (2014) Gradient based prediction for reversible watermarking by difference expansion. In: IH&MMSEC'14, pp 35–41
7. Dragoi C, Coltuc D (2014) Local-prediction-based difference expansion reversible watermarking. *IEEE Trans Image Process* 23(4):1779–1781
8. Fallahpour M (2008) Reversible image data hiding based on gradient adjusted prediction. *IEICE Electron Exp* 5(20):870–876
9. Giller G (2005) A generalized error distribution. Available at SSRN: <http://ssrn.com/abstract=2265027> or doi:10.2139/ssrn.2265027
10. Hu X, Zhang W, Hu X, Yu N, Zhao X, Li F (2013) Fast estimation of optimal marked-signal distribution for reversible data hiding. *IEEE Trans Inf Forensics Secur* 8(5):779–788
11. Hu Y, Lee H-K, Li J (2009) DE-based reversible data hiding with improved overflow location map. *IEEE Trans Circuits Syst Video Technol* 19(2):250–260
12. Kamstra LHJ, Heijmans AM (2005) Reversible data embedding into images using wavelet techniques and sorting. *IEEE Trans Image Process* 14(12):2082–2090
13. Lee S, Yoo CD, Kalker T (2007) Reversible image watermarking based on integer-to-integer wavelet transform. *IEEE Trans Inform Forensics Secur* 2(3):321–330
14. Li X, Yang B, Zeng T (2011) Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection. *IEEE Trans Image Process* 20(12):3524–3533
15. Lin S-J, Chung W-H (2012) The scalar scheme for reversible information-embedding in gray-scale signals: capacity evaluation and code constructions. *IEEE Trans Inf Forensics Secur* 7(4):1155–1167
16. Luo L, Chen Z, Chen M, Zeng X, Xiong Z (2010) Reversible image watermarking using interpolation technique. *IEEE Trans Inf Forensics Secur* 5(1):187–193
17. Nadarajah S (2005) A generalized normal distribution. *J Appl Stat* 32(7):685–694
18. Ni Z, Shi Y, Ansari N, Wei S (2006) Reversible data hiding. *IEEE Trans Circ Syst Video Technol* 16(3):354–362
19. Ou B, Li X, Zhao Y, Ni R (2013) Pairwise prediction-error expansion for efficient reversible data hiding. *IEEE Trans Image Process* 22(12):5110–5121
20. Qin C, Chang C-C, Huang Y-H, Liao L-T (2013) An inpainting-assisted reversible steganographic scheme using histogram shifting mechanism. *IEEE Trans Circ Syst Video Technol* 23(7):1109–1118
21. Rad RM, Attar A (2013) A predictive algorithm for multimedia data compression. *Multimed Syst* 19(2):103–115
22. Rényi A (1961) On measures of entropy and information. In: Proceedings of the 4th Berkeley symposium on mathematical statistics and probability, vol I, pp 547–561

23. Sachnev V, Kim HJ, Nam J, Suresh S, Shi Y (2009) Reversible watermarking algorithm using sorting and prediction. *IEEE Trans Circuits Syst Video Technol* 19(7):989–999
24. Shannon CE (1948) A mathematical theory of communication. *Bell Syst Tech J* 27:379–423
25. Thodi DM, Rodriguez JJ (2007) Expansion embedding techniques for reversible watermarking. *IEEE Trans Image Process* 16(3):721–730
26. Tian J (2003) Reversible data embedding using a difference expansion. *IEEE Trans Circuits Syst Video Technol* 13(8):890–896
27. Wang X, Li X, Yang B, Guo Z (2010) Efficient generalized integer transform for reversible watermarking. *IEEE Signal Process Lett* 17(6):567–570
28. Wu H-T, Huang J (2012) Reversible image watermarking on prediction errors by efficient histogram modification. *Signal Process* 92(12):3000–3009
29. Zhang W, Chen B, Yu N (2011) Capacity-approaching codes for reversible data hiding. In: *Proceedings of the 13th information hiding conference, LNCS 6958*, pp 255–269
30. Zhang W, Chen B, Yu N (2012) Improving various reversible data hiding schemes via optimal codes for binary covers. *IEEE Trans Image Process* 21(6):2991–3003
31. Zhang W, Hu X, Li X, Yu N (2013) Recursive histogram modification: establishing equivalency between reversible data hiding and lossless data compression. *IEEE Trans Image Process* 22(7):2775–2785



Jiajia Xu received the B.S. degree in 2009 from the Hefei University of Technology (HFUT), and the M.S. degree in 2012 from the University of Science and Technology of China (USTC), Hefei, China. He is now pursuing the Ph.D. degree in USTC. His research interests include cloud security, image and video processing, video compression and information hiding.



Weiming Zhang received the M.S. degree and Ph.D. degree in 2002 and 2005, respectively, from the Zhengzhou Information Science and Technology Institute, Zhengzhou, China. Currently, he is an Associate Professor with the School of Information Science and Technology, University of Science and Technology of China, Hefei, China. His research interests include multimedia security, information hiding and cryptography.



Ruiqi Jiang received the B.S. degree in 2010 from the Haerbin Institute of Technology (HIT), and the M.S. degree in 2013 from the New York University. He is now pursuing the Ph.D. degree in University of Science and Technology of China (USTC), Hefei, China. His research interests include cloud security and information hiding.



Nenghai Yu received the B.S. degree in 1987 from Nanjing University of Posts and Telecommunications, Nanjing, China, the M.E. degree in 1992 from Tsinghua University, Beijing, China, and the Ph.D. degree in 2004 from the University of Science and Technology of China, Hefei, China, where he is currently a Professor. His research interests include multimedia security, multimedia information retrieval, video processing and information hiding.