HORIZONTAL PAIRWISE REVERSIBLE WATERMARKING

Ioan-Catalin Dragoi, Dinu Coltuc and Ion Caciula

Electrical Engineering Dept.
Valahia University of Targoviste, Romania
Email: {dragoi, coltuc, caciula}@valahia.ro

ABSTRACT

The best results for very low embedding capacity reported so far are provided by a pairwise reversible watermarking scheme that uses paired embedding with rhombus prediction and context smoothness control. Since the horizontal (or vertical) adjacent pixels are more correlated than the diagonal ones, this paper proposes such a pairing for reversible watermarking. The rhombus prediction should be modified in order to match the horizontal/vertical pairing. An improved measure of the context smoothness is also introduced. The proposed horizontal pairing based reversible watermarking scheme outperforms the state of the art diagonal pairing one. Experimental results for standard graylevel test images are provided.

Index Terms— pairwise reversible watermarking, histogram shifting, prediction-error expansion

1. INTRODUCTION

Reversible watermarking extracts the embedded data and exactly recovers the original host image without any distortion. Three major reversible watermarking approaches have been developed so far. They are based on lossless compression, difference expansion (DE) and histogram shifting (HS). The reversible watermarking by lossless compression (based on substituting part of the host with a compressed version of itself and the watermark) is rarely used today.

DE reversible watermarking, introduced by Tian in [1], creates space for a hidden data bit by increasing two times the difference between two adjacent pixels. Thodi et. al, [2], adapted the DE algorithm for a pixel and its predicted value. This approach is known as prediction-error expansion (PEE). Significant research was devoted to reducing this prediction-error by developing increasingly efficient predictors [3]– [7]. Other notable improvements to PEE reversible watermarking are: local complexity based capacity control [8], reduced distortion through context insertion [9], efficient embedding using pixel pairs and a 2D prediction-error histogram [10], etc.

This work was supported by UEFISCDI Romania, Grant PN-II-PT-PCCA-2013-4-1762.

HS based reversible watermarking was introduced in [11] by Ni et. al and considered the graylevel histogram of the image. This approach selects one histogram bin and creates space for data embedding in an adjacent bin by shifting part of the histogram. The graylevel histogram was latter replaced by the prediction-error histogram (see [12]). This approach is known as prediction-error based histogram shifting (PE-HS). Embedding into the bin ”0” of the prediction-error histogram with PE-HS produces the same results as PEE. Although they were developed in parallel, DE can be considered a generalized form of HS.

As far as we know, the pairwise algorithm of [10] is the best algorithm proposed so far for very low embedding capacity reversible watermarking. This paper proposes an improved pairwise algorithm. While in [10] the pairing of pixels is limited by the predictor, in the proposed scheme the predictor was chosen after the optimal pairing was determined, ensuring good correlation between the paired pixels. Most aspects of the embedding scheme were modified to suit the new pairing. We remind that in [10] the pairwise algorithm is presented in the frame of PEE reversible watermarking. In this paper we introduce the pairwise algorithm in a PE-HS framework.

The outline of the paper is as follows. The basic PE-HS scheme of [12] and the pairwise PEE approach of [10] are briefly discussed in Section 2. The proposed pairwise reversible watermarking scheme is presented in Section 3. Experimental results are provided in Section 4. Finally, the conclusions are drawn in Section 5.

2. RELATED WORK

As said above, we introduce the pairwise reversible watermarking of [10] not as a PEE scheme, but as a PE-HS one. We believe that the pairwise scheme can be more easily understood in the PE-HS framework. The basic principle of PE-HS is first discussed in Section 2.1 and then the pairwise scheme is described in Section 2.2.

978-0-9928626-3-3/15/$31.00 ©2015 IEEE

2.1. Basic PE-HS

The basic PE-HS embedding scheme processes the image pixel by pixel in a selected order, usually raster-scan. Let $\hat{x}_i$ be the predicted value for the x_i pixel. The corresponding prediction error is:

$$e_i = x_i - \hat{x}_i \tag{1}$$

Based on the required embedding capacity, two prediction error values (L and R, with $L < R$) are selected and x_i is modified according to:

$$x'_i = \begin{cases} x_i + b_i, & \text{if } e_i = R \\ x_i - b_i, & \text{if } e_i = L \\ x_i + 1, & \text{if } e_i > R \\ x_i - 1, & \text{if } e_i < L \end{cases} \tag{2}$$

where $b_i \in \{0, 1\}$ is the hidden data bit.

There must be sufficient pixels with $e_i \in \{L, R\}$ to allow the embedding of the entire payload. The graylevel values of the pixels with $e_i < L$ and $e_i > R$ are shifted one position left/right in order to prevent them from overlapping with the marked values. The remaining pixels (with $L < e_i < R$) are left unchanged.

Equation (2) can produce values outside the valid graylevel range of the image (overflow/underflow). The problem is solved by leaving unchanged the pixels where the overflow/underflow can appear and by recording their position in a map. Since the map and the L, R values must be available at decoding, they are inserted into the image by LSB substitution. The substituted LSB values are added to the payload.

At decoding, the modified prediction error is first computed as:

$$e'_i = x'_i - \hat{x}_i \tag{3}$$

The hidden bit is then extracted from the marked pixels and all the modified pixels are restored to their original values:

$$b_i = \begin{cases} 0, & \text{if } e'_i = L \text{ or } e'_i = R \\ 1, & \text{if } e'_i = L - 1 \text{ or } e'_i = R + 1 \end{cases} \tag{4}$$

$$x_i = \begin{cases} x'_i + 1, & \text{if } e_i < L \\ x'_i - 1, & \text{if } e_i > R \end{cases} \tag{5}$$

2.2. Pairwise PEE

The image pixels can be grouped and processed as a pair. Let (x_1, x_2) be a pixel pair. Each pixel is individually modified (marked or shifted) forming four possible cases: both pixels are marked (mm), x_1 is marked and x_2 is shifted (ms), x_1 is shifted and x_2 is marked (sm) or both are shifted (ss). Equation (6) shows the RR part (when $e_1, e_2 \geq R$) of the 2D equivalent of (2). The remaining parts (RL, LR, LL) can be easily deduced from (6).

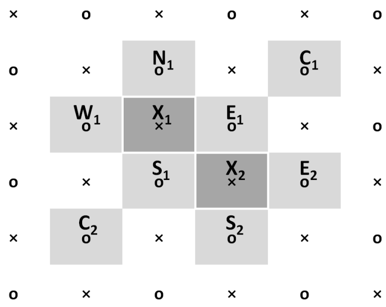

Fig. 1. A pair of cross pixels and the corresponding dot pixels used by [10] for determining local complexity.

In equation (6), for (mm) pairs, 2 bits are inserted into (x_1, x_2). An original idea of [10] is to eliminate the insertion of two bits of "1". By embedding only three combinations of the four possible, the embedding capacity decreases from two bits to $\log_2 3$ bits per pair. The advantage is a reduction of the watermarking distortion. The new embedding procedure is shown in equation (7). By eliminating the sequence "11" from direct embedding, the hiding of one bit into the pairs of pixels where both prediction errors are equal to R+1 becomes possible. This corresponds to the (mm^*) pairs in equation (7), pairs treated as (ss) in equation (6). The extra capacity gained from (mm^*) exceeds the one lost by (mm). It should be noticed that for (mm^*) pairs the marking by equation (7) introduces less distortion than the shifting by (6). Furthermore, for (ms) and (sm) pairs both (6) and (7) operate identically. Finally, as previously mentioned, there are less pixels in (ss) for (7).

The watermarking scheme of [10] uses the rhombus average of [8] for predicting the (x_1, x_2) pair and a similar capacity/distortion control. The paired pixels must be correlated in order to ensure good results. Furthermore, x_1 and x_2 must be embedded as a pair. Since they should be detected as a pair too, the two pixels cannot be part of each others prediction context. With the rhombus average, one has:

$$\begin{aligned} \hat{x}_1 &= \lfloor (N_1 + W_1 + E_1 + S_1)/4 + 0.5 \rfloor \\ \hat{x}_2 &= \lfloor (E_1 + S_1 + E_2 + S_2)/4 + 0.5 \rfloor \end{aligned} \tag{8}$$

Since the four nearest neighbors of each pixel are reserved for prediction, the diagonal pairing appears to be the logical solution.

The pixels are also split into two sets: dot and cross, forming a chessboard pattern (Fig. 1). Note that the paired pixels belong to the same set. The entire cross set is embedded before processing the dot set. At the detection stage the order is reversed: all dot pixels are restored before decoding the cross set. This allows for an unaltered prediction context for the cross set (less important for low embedding capacities) and context based sorting.

The prediction performance varies wildly in complex im-

$$(x_1', x_2') = \begin{cases} (x_1, x_2) + (b_1, b_2), & \text{if } e_1 = R \text{ and } e_2 = R \quad (mm) \\ (x_1, x_2) + (b_i, 1), & \text{if } e_1 = R \text{ and } e_2 > R \quad (ms) \\ (x_1, x_2) + (1, b_i), & \text{if } e_1 > R \text{ and } e_2 = R \quad (sm) \\ (x_1, x_2) + (1, 1), & \text{if } e_1 > R \text{ and } e_2 > R \quad (ss) \end{cases} \tag{6}$$

where $(b_1, b_2) \in \{(0,0), (0,1), (1,0), (1,1)\}$.

$$(x_1', x_2') = \begin{cases} (x_1, x_2) + (b_1, b_2), & \text{if } e_1 = R \text{ and } e_2 = R \quad (mm) \\ (x_1, x_2) + (b_i, b_i), & \text{if } e_1 = R+1 \text{ and } e_2 = R+1 \quad (mm^*) \\ (x_1, x_2) + (b_i, 1), & \text{if } e_1 = R \text{ and } e_2 > R \quad (ms) \\ (x_1, x_2) + (1, b_i), & \text{if } e_1 > R \text{ and } e_2 = R \quad (sm) \\ (x_1, x_2) + (1, 1), & \text{if } (e_1 > R+1, e_2 > R) \text{ or } (e_1 > R, e_2 > R+1) \quad (ss) \end{cases} \tag{7}$$

where $(b_1, b_2) \in \{(0,0), (0,1), (1,0)\}$ and $(b_i, b_i) \in \{(0,0); (1,1)\}$.

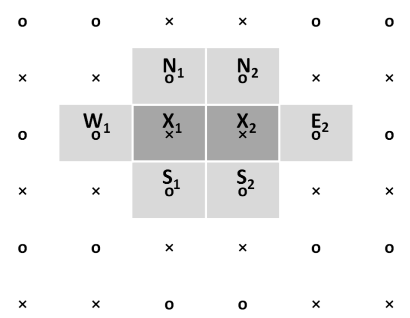

Fig. 2. The proposed pixel pair selection with the neighbors used for prediction and for local complexity computation.

age regions. It is preferable to prioritize the embedding in uniformed areas than to risk using a region containing only pixels that must be shifted. As was previously mentioned the shifting operation is performed in order to avoid overlapping between marked and unmarked pixels. The shifted pixels do not provide embedding capacity, but are distorted. For each (x_1, x_2) pair in the current set the local complexity is evaluated based on neighboring pixels from the other set:

$$\begin{aligned} l_i \; &= |N_1 - W_1| + |W_1 - S_1| + |S_1 - E_1| + |E_1 - N_1| \\ &+ |S_1 - S_2| + |S_2 - E_2| + |E_2 - E_1| + |E_1 - C_1| \\ &+ |S_1 - C_2| \end{aligned} \tag{9}$$

A complexity threshold T is then used to control the number of pixels considered for watermarking. Only the pairs of pixels with $l_i < T$ from each set are considered for embedding, the rest are left unchanged. The same l_i value is computed at the decoding stage.

Before going any further we stress that, as far as we know, the results in [10] are the best reported so far in literature for very low capacity reversible watermarking.

3. PROPOSED HORIZONTAL PAIRWISE SCHEME

The proposed scheme uses the same watermarking framework described in Section 2.2: the image is split in two sets, a complexity threshold is used to determine the local smoothness and the pixels are processed in pairs by using (7). The improvement comes from prioritizing (x_1, x_2) correlation over prediction precision. A better method to determine regional complexity is also introduced.

As previously mentioned, the horizontal/vertical neighbors that form the rhombus context are closer to the central pixel than their diagonal counterparts. Thus, it can be expected that the horizontal/vertical pairing provides better results than the diagonal one used in [10]. The horizontal pairing presented in Fig. 2 is investigated.

It immediately appears that the rhombus average can not be used for prediction. A slightly weaker predictor is considered:

$$\begin{aligned} \hat{x}_1 &= \left\lfloor (N_1 + W_1 + \tfrac{N_2 + W_1 + E_2 + S_2}{4} + S_1)/4 + 0.5 \right\rfloor \\ \hat{x}_2 &= \left\lfloor (N_2 + \tfrac{N_1 + W_1 + E_2 + S_1}{4} + E_2 + S_2)/4 + 0.5 \right\rfloor \end{aligned} \tag{10}$$

Note that the performance of the pairwise PEE scheme is due to the use of l_i and to the correlation between x_1 and x_2. Since for very low bit-rates the required capacity can be easily provided by most predictors, we consider that the distortion control is much more important. A modified complexity measure is proposed next.

The horizontal pixels are first paired and then, not the pixels, but the pairs are split between the cross and dot sets. The chess board pattern is maintained with pairs instead of single pixels (Fig. 2). For each pair in a set, the local complexity is computed:

$$\begin{aligned} l_{i,j} \; &= |N_1 - W_1| + |W_1 - S_1| + |S_1 - S_2| + |S_2 - E_2| \\ &+ |E_2 - N_2| + |N_2 - N_1| \end{aligned} \tag{11}$$

where i and j are the row/column of the pixel pair.

By considering the complexity value of neighboring pairs from the same set, the regional complexity can be determined:

$$r_{i,j} = l_{i,j} + l_{i-1,j-1} + l_{i-1,j+1} + l_{i+1,j-1} + l_{i+1,j+1} \tag{12}$$

Fig. 3. Test images: *Peppers*, *Elaine Boat*, *Lake*, *Barbara*, *Lena*, *Woman* and *Mandrill*

The first/last line/row of pixels cannot be predicted by using (10) and are not considered for watermarking. The pairing starts from line 2. Some pairs do not have all the four required neighbors for computing $r_{i,j}$ as above. They can be still considered for watermarking by computing $r_{i,j}$ using only the available local complexity values and weighting them accordingly (for example: $r_{1,1} = 3l_{1,1} + 2l_{2,2}$).

4. EXPERIMENTAL RESULTS

In this section, experimental results for the horizontal pairwise reversible watermarking scheme are presented. Its performance is evaluated by computing the peak signal-to-noise ratio (PSNR) between the original host image and its watermarked version. The PSNR value is used to determine the distortion introduced by the reversible watermarking scheme for a given embedding capacity. Eight graylevel 512×512 images extensively used in reversible watermarking are considered, namely: *Peppers*, *Elaine*, *Boat*, *Lake*, *Barbara*, *Lena*, *Woman* and *Mandrill*. The test images are presented in Fig. 3.

Before going any further, the case of vertical pairing must be discussed. On seven of the eight test images, vertical pairing had similar or inferior results compared to its horizontal counterpart. The inferior results were obtained on *Boat* (with an average 0.6 dB difference in PSNR) and *Barbara* (0.2 dB). On the eighth image, *Mandrill*, vertical pairing obtained an average 0.2 dB increase in PSNR over the horizontal approach, but was outperformed by the diagonal one. Based on this results, horizontal pairing is recommended over its vertical counterpart.

Another aspect that must be mentioned is the benefit of using $r_{i,j}$ instead of the simpler $l_{i,j}$ for regional complexity. On the eight test images, (12) brings an average increase in PSNR of 0.11 dB. This gain is large enough to justify its use. Equation (12) was also adapted for the watermarking schemes of [8] and [10], using their specific local complexity equations. The average improvement for [8] was 0.18 dB for bit-rates of up to 1 bpp and 0.48 dB when considering only low embedding capacities. For [10] the average gain was 0.09 dB.

Next, the proposed pairwise PE-HS scheme is compared with the sorting approach of [8] and the diagonal pairing of [10]. The results are presented in Fig. 4. The proposed scheme outperforms the one of [8] on all test images. The gain in PSNR was between 0.67 dB (on *Mandrill*, one of only two improvements bellow 1 dB, the other was on *Woman*) and 1.9 dB (on *Pappers*). The average for the entire set is 1.22 dB.

The most noticeable improvement compared to [10] is obtained on *Peppers* with an average gain in PSNR of 0.72 dB. Good results are also obtained on *Elaine* (0.46 dB), *Lake* (0.43 dB), and *Boat* (0.4 dB). On the next three images, *Barbara*, *Lena* and *Woman*, the gain in PSNR obtained for capacities bellow 20,000 bits is balanced out by the loss in performance for capacities above 40,000 bits. The large capacities have been obtained for $L = -1$, $R = 0$. For this images, the pixels of smooth regions are horizontally correlated, but as more pixels need to be used, they begin to be outnumbered by the diagonally correlated ones from more complex regions. On *Mandrill*, the diagonal pairing of [10] provides the best results. Thus, the proposed scheme is outperformed by an average of 0.4 dB. Textured images like *Boat* and *Mandrill*, with very limited smooth areas, can favor a certain pairing. For such images, texture orientation can be used to determine the most suitable pairing direction before the embedding process.

5. CONCLUSIONS

An improved pairwise reversible watermarking scheme has been proposed. Pixels are horizontally paired in order to take advantage of the correlation between adjacent pixels. The embedding process is adapted accordingly. An improved method for determining the local smoothness has also been introduced. The proposed scheme outperforms the recently published pairwise embedding scheme (which has the best results reported so far in literature for low capacity reversible watermarking).

REFERENCES

[1] J. Tian, ”Reversible Data Embedding Using a Difference Expansion”, *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 890–896, 2003.

[2] D. M. Thodi and J. J. Rodriguez, ”Expansion Embedding Techniques for Reversible Watermarking”, *IEEE Trans. Image Process.*, vol. 15, pp. 721–729, 2007.

[3] X. Li, B. Yang and T. Zeng, ”Efficient Reversible Watermarking Based on Adaptive Prediction-Error Expansion and Pixel Selection”, *IEEE Trans. on Image Process.*, vol. 20, no. 12, pp. 3524–3533, 2011.

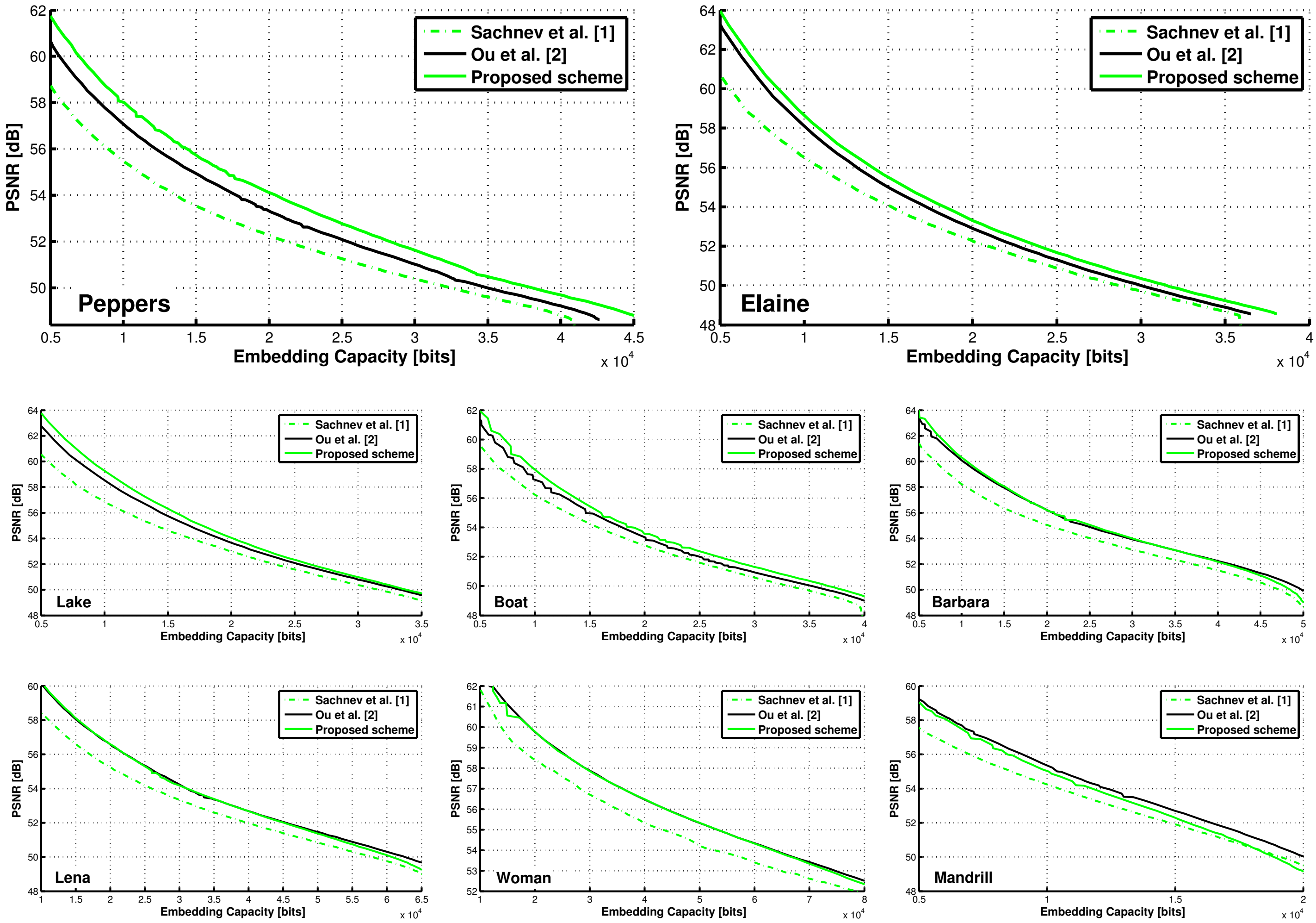

Fig. 4. Experimental results for the proposed pairwise PEE scheme.

[4] I.-C. Dragoi and D. Coltuc, Improved Rhombus Interpolation for Reversible Watermarking by Difference Expansion, *Proc. 20th European Conf. on Signal. Process., EUSIPCO2012*, pp. 1688–1692, 2012.

[5] W.-J. Yang, K.-L. Chung, H.-Y. M. Liao and W.-K. Yu, "Efficient reversible data hiding algorithm based on gradient-based edge direction prediction", *Journal of Systems and Software*, vol. 86, no. 2, pp. 567–580, 2013.

[6] X. Li, B. Li, B. Yang and T. Zeng, "General Framework to Histogram-Shifting-Based Reversible Data Hiding", *IEEE Trans. on Image Process.*, vol. 22, no. 6, pp. 2181–2191, 2013.

[7] I.-C. Dragoi and D. Coltuc, "Local Prediction Based Difference Expansion Reversible Watermarking", *IEEE Trans. on Image Processing*, vol. 23, no. 4, pp. 1779–1790, 2014.

[8] V. Sachnev, H. J. Kim, J. Nam, S. Suresh and Y. Q. Shi, "Reversible Watermarking Algorithm Using Sorting and Prediction", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, pp. 989–999, 2009.

[9] D. Coltuc, "Improved Embedding for Prediction Based Reversible Watermarking", *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 873–882, 2011.

[10] B. Ou, X. Li, Y. Zhao, R. Ni and Y.-Q. Shi, "Pairwise Prediction-Error Expansion for Efficient Reversible Data Hiding", *IEEE Trans. on Image Processing*, vol. 22, no. 12, pp. 5010–5021, 2013.

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

[12] P. Tsai, Y.-C. Hu, and H.-L. Yeh, "Reversible image hiding scheme using predictive coding and histogram shifting", *Signal Process.*, vol. 89, no. 6, pp. 11291143, 2009.