INTERNATIONAL JOURNAL OF RESEARCH
In Advanced Engineering Technologies

Volume 4, Issue 4 AUG 2015

ANALYTICAL PERFORMANCE OF LOCAL PREDICTION BASED REVERSIBLE WATERMARKING SCHEME FOR ACCURATE ANALYSIS

B. S. SRINIVAS1 Mr. S. CHAKRI SREEDHAR2

Department of ECE (DECS) [1] ***Assistant Professor*** [2]

SREE RAMA ENGINEERING COLLEGE, Tirupati, Andhra Pradesh, INDIA

Abstract

In digital image processing domain security area occupies an important role to protect the data from unauthenticated process. Although tremendous progress has been made in the past years on reversible watermarking, there still exist a number of problems. We believe that the most important one is related to the reversible watermarking such as robustness and the conventional schemes designs are not allowed the recovery of the same predictor at detection, without any additional information. The proposed local prediction is general and it applies regardless of the predictor order or the prediction context. For the particular cases of least square predictors with the same context as the median edge detector, gradient-adjusted predictor or the simple rhombus neighborhood, the local prediction-based reversible watermarking clearly outperforms the state-of-the-art schemes based on the classical counterparts. Experimental results are provided.

. INTRODUCTION

Data Hiding is referred to as a process to hide data (representing some information) into cover media. That is, the data hiding process links two sets of data, a set of the embedded data and another set of the cover media data. The relationship between these two sets of data characterizes different applications. For instance, in covert communications, the hidden data may often be irrelevant to the cover media. In authentication, however, the embedded data are closely related to the cover media. In these two types of applications, invisibility of hidden data is an important requirement.

In most cases of data hiding, the cover media will experience some distortion due to data hiding and cannot be inverted back to the original media. That is, some permanent distortion has occurred to the cover media even after the hidden data have been extracted out. In some applications, such as medical diagnosis and law enforcement, it is critical to reverse the marked media back to the original cover media after the hidden data are retrieved for some legal considerations.

In other applications, such as remote sensing and high-energy particle physical experimental investigation, it is also desired that the original cover media can be recovered because of the required high-precision nature. The marking techniques satisfying this requirement are referred to as reversible, lossless, distortion-free, or invertible data hiding techniques. Reversible data hiding facilitates immense possibility of applications to link two sets of data in such a way that the cover media can be losslessly recovered after the hidden data have been extracted out, thus providing an additional avenue of handling two different sets of data

Obviously, most of the existing data hiding techniques are not reversible. For instance, the widely utilized spread-spectrum based data hiding methods are not invertible owing to truncation (for the purpose to prevent over/underflow) error and round-off error. The well-known least significant bit plane (LSB) based schemes are not lossless owing to bit replacement without “memory.” Another category of data hiding techniques, quantization-index-modulation (QIM) based schemes are not distortion-free owing to quantization error

Recently, some reversible marking techniques have been reported in the literature. The first method is carried out in the spatial domain. It uses modulo 256 addition (assuming here that eight-bit grayscale images are considered) to embed the hash value of the original image for authentication. The embedding formula is , in which denotes the original image, the marked image, and the watermark, where denotes the hash function operated on the original image , and the secret key. Because of using modulo 256 addition, the over/underflow is prevented and the reversibility is achieved. Some annoying salt-and-pepper noise, however, is generated owing to possible grayscale value flipping over between 0 and 255 in either direction during the modulo 256 addition. The second reversible marking technique was developed in the transform domain ,which is based on a lossless multi resolution transform and the idea of patchwork It also uses modulo 256 addition. Note that no experimental results about this technique have been reported. Another spatial domain technique was reported in that losslessly compresses some selected bit plane(s) to leave space for data embedding. Because the necessary bookkeeping data are also embedded in the cover media as an overhead, the method is reversible. Since these techniques at authentication, the amount of hidden data is limited. The capacity of method ,which is based on the idea of patchwork and modulo 256 addition, is also limited except that the hidden data exhibit some robustness against high quality JPEG compression. Since it uses modulo 256 addition, it also suffers from salt-and-pepper noise. As a result, the technique cannot be utilized in many applications .This observation is valid to all lossless data hiding algorithms that use modulo 256 addition to achieve reversibility.

The first reversible marking technique that is suitable for a large amount of data hiding was presented. This technique first segments an image into non overlapping blocks, and then introduces a discriminating function to classify these blocks into three groups: R(egular), S(ingular), and U(nusable). It further introduces a flipping operation, which can convert an R-block to an S-block and vice versa. A U-block remains intact after the flipping operation. By assigning, say, binary 1 to an R-block and binary 0 to an S-block, all R- and S-blocks are scanned in a chosen sequential order, resulting in a biased (meaning that the binary numbers of 1 and 0 are not balanced) binary sequence. This biased binary sequence is losslessly compressed to leave space for data embedding and the compressed bit sequence is embedded into the cover media as an overhead for later reconstruction of the original image.

In data embedding, the R- and S-blocks are scanned once again and the flipping operation is applied whenever necessary to make the changed R- and S-block sequence coincident with the to-be-embedded data followed by the overhead data mentioned above. While it is novel and successful in reversible data hiding, the payload is still not large enough for some applications. Specifi- cally, the embedding capacity estimated by authors ranges from 3 to 41 kb for a 512 512 8 cover grayscale image when the embedding amplitude is 4 (the estimated average PSNR of the marked image versus the original image is 39 dB) . Another problem with the method is that when the embedding strength increases in order to increase the payload, the visual quality of the marked image will drop severely due to annoying artifacts. To increase the payload dramatically, a new lossless data hiding technique based on integer wavelet transform (IWT) , (a second generation wavelet transform, which has avoided round-off errors) was developed recently. Because of the superior decorrelation capability of wavelet transform, the selected bit plane compression of IWT coefficients in high frequency subbands creates more space for data hiding, resulting in a two to five times payload as large .

Specifically, its payload ranges from 15 to 94 kb for a 512 512 8 grayscale image at the same (39 dB) PSNR of the marked images compared with the original images. To achieve reversible data hiding, a histogram modification is applied in its pre-processing to prevent over/underflow. This histogram modification causes, however, a relatively low PSNR of the marked image versus the original image though there are no annoying artifacts. It is noted that reversible data hiding has attracted increasing attention recently, and more algorithms are being developed. A very recent example is the technique reported. The main idea is that in the embedding phase, the host signal is quantized and the residual is obtained. Then the authors adopt the CALIC lossless image compression algorithm, with the quantized values as side information, to efficiently compress the quantization residuals to create high capacity for the payload data . The compressed residual and the payload data are concatenated and embedded into the host signal via generalize grayscale image while the PSNR is 38 dB. Even though the payload is high, the PSNR is still not high enough-LSB modification method. The payload of this technique is from 15 to 143 kb for a 512 512 8 grayscale image while the PSNR is 38 dB. Even though the payload is high, the PSNR is still not high enough.

DIFFERENCE EXPANSION REVERSIBLE WATERMARKING:

We briefly remind the basic principles of the difference expansion with histogram shifting (DE-HS) reversible watermarking for the case of prediction-error expansion (also called prediction-error expansion). The section introduces the LS prediction as well.

Basic Reversible Watermarking Scheme:

Let $\hat{x}_{i,j}$ be the estimated value of the pixel $x_{i,j}$. The prediction error is

$$e_{i,j} = x_{i,j} - \hat{x}_{i,j} \quad \ldots (1)$$

Let T >0 be the threshold. The threshold controls the distortion introduced by the watermark. Thus, if the prediction error is less than the threshold and no overflow or underflow is generated, the pixel is transformed and a bit of data, b, is embedded. The transformed pixel is:

$$x'_{i,j} = x_{i,j} + e_{i,j} + b \ldots\ldots (2)$$

The embedded pixels are also called carrier pixels (see [12]). The pixels that cannot be embedded because $|e_{i,j}| \geq T$ (the non-carriers) are shifted in order to provide, at detection, a greater prediction error than the one of the embedded pixels. These pixels are modified as follows:

$$x'_{i,j} = \begin{cases} x_{i,j} + T, & \text{if } e_{i,j} \geq T \\ x_{i,j} - (T-1), & \text{if } e_{i,j} \leq -T \end{cases} \quad \ldots\ldots (3)$$

The underflow/overflow cases are solved either by creating a map of underflow/overflow pixels or by using flag bits . Let us suppose that, at detection, one gets the same predicted value for the pixel $x_{i,j}$. The prediction error at detection is

$$e'_{i,j} = x'_{i,j} - \hat{x}_{i,j} \quad \ldots\ldots (4)$$

The discrimination between embedded and translated pixels is provided by the prediction error If

$-2T \leq e'_{i,j} \leq 2T+1$ one has an embedded pixel. For the embedded pixels one has $e'_{i,j}=2e'_{i,j}+b$ and b follows as the LSB of $e'_{i,j}$. The original pixel is immediately recovered as

$$x = \frac{x'_{i,j} + \hat{x}_{i,j} - b}{2} \quad \dots (5)$$

For the shifted pixels, the original pixel recovery follows by inverting equation (3). As long as at detection one has the same predicted value, the reversibility of the watermarking scheme is ensured. The same predicted value is obtained if the pixels within the prediction context are recovered before the prediction takes place. Let us suppose that the watermarking proceeds in a certain scan order.

The decoding should proceed in a reverse order. The first pixel restored to its original value is the last embedded one. Obviously, for the last embedded pixel, one has the same prediction context both at detection and at embedding. Once the last embedded pixel has been restored, one recovers the context for the prediction of its predecessor and so on. Usually, anti-causal predictors are used and the embedding is performed in raster scan order, row by row, from the upper left to the lower right pixel. The use of anti-causal predictors with the normal raster scan has the advantage of using for prediction only the original pixel values.

Before going any further, a comment should be made. In fact, as shown , it is not the predicted value that should be exactly recovered at detection, but the expanded prediction error The embedding capacity of the basic DE HS scheme is given by the number of pixels that are embedded with equation (2), namely the pixels having the absolute prediction error lower than the threshold. Obviously, the capacity depends on the prediction error, i.e. on the quality of the prediction.

Linear Prediction

As said above, adaptive predictors can provide better results than fixed predictors like MED, GAP, the average on the four horizontal and vertical neighbors, etc. We shall focus on linear predictors. By linear prediction, image pixels $x_{i,j}$ are estimated by a weighted sum over a certain neighborhood of $x_{i,j}$

In order to simplify the notations, we consider an indexing of the neighborhood (prediction context), p=1,...,k, namely

$x^1_{i,j}$,..., $x^k_{i,j}$,wherek is the order of the predictor. Let v=[x_1,..., x_k] be the column vector with the coefficients of the predictor. Let xi,j be the row vector obtained by ordering the context of xi,j according to the indexing. The predicted pixel can be written in closed form as

$$\hat{x}_{i,j} = XV \quad \dots (6)$$

A rather similar form, used mainly in linear regression, includes also a constant term

$$\hat{x}_{i,j} = v_0 + \sum_{p=1}^{k} v_p \, x^p_{i,j} \quad \dots\dots (7)$$

When the constant term is used, the vector $x_{i,j}$is extended by adding a first element, $x_{i,j}$=1. We shall consider mainly this latter form. The predicted value and the prediction error depend onv. We shall further write $\hat{x}_{i,j}$ (v) and $e'_{i,j}(v)$ .A popular solution to the linear regression problem is the least square (LS) approach. We remind that the LS considers the weights that minimize the sum of the squares of the prediction error. The LS predictor is the one that provides.

$$\min_{v} \sum_i \sum_j e_{i,j}(v)^2 \quad \dots (8)$$

Let y be the column vector obtained by scanning the image along the rows and let X be the matrix whose rows are the corresponding context vectors as defined above. The prediction error vector is y−Xv. Equation (8) corresponds to the minimization of$(y - Xv)'$ (y−Xv),where“ ‘ ” denotes vector/matrix transposition. By taking the partial derivatives of the square error with respect to the components of v and by setting them equal to zero one gets XX’ v=X ‘ y and, finally

$$v = (X'X)^{-1}X'y \quad \dots (9)$$

LOCAL PREDICTION REVERSIBLE WATERMARKING

In order to illustrate the reduction of the prediction error provided by using a distinct predictor for each pixel, a simple example is presented. Let us consider the case of the rhombus context and let us evaluate the mean squared prediction error for local LS prediction computed on a $B \times B$ sliding window. The improvement depends on the image content, namely it is more significant for images with a high content of texture or fine details than for the ones with large uniform areas.

A. Local Prediction

Next, we investigate the computation of a distinct predictor for each pixel. Obviously, the embedding of the predictors coefficients into the image is out of the question. Therefore, instead of computing the predictors on original image blocks, we investigate the computation on blocks containing both original and modified pixels.

Let the pixels be embedded in a raster-scan order, pixel by pixel and row by row, from the upper left corner to the lower right one. Obviously, the decoding proceeds in reverse order, from bottom to top, starting with the last embedded pixel.

If the prediction context has k pixels, the central pixel takes part in k other prediction equations. There are two solutions:

1) The vector corresponding to the central pixel, $\boldsymbol{x_{i,j}}$, as well as the ones that contain the central pixel ($x_{l,m}$, with $\boldsymbol{x_{i,j}}$ $\in x_{l,m}$) are eliminated from $X_{i,j}$and the

central pixel as well as the pixels $x_{l,m}$ are eliminated from $y_{i,j}$;

2) Before the construction of $X_{i,j}$ and $y_{i,j}$, the central pixel of the block $\boldsymbol{x_{i,j}}$ is replaced by an estimate $\tilde{x}_{i,j}$ computed by using a fixed predictor as the one of equation (10).

$$\tilde{x}_{i,j} = \frac{x_{i-1,j} + x_{i+1,j} + x_{i,j-1} + x_{i,j+1}}{4} \quad \dots (10)$$

The first solution is simple, but does not consider the pixels close to the current pixel as sample data for the computation of the current predictor. The second solution eliminates this drawback, even if instead of the true central pixel value we use an estimate.

B. Proposed Scheme

In raster scan order ,the image pixels has been processed which is started from the upper left corner.

The proceedings for each pixel $x_{i,j}$ are as follows :

1). Following or using the local LS prediction scheme or the fixed predictor , we are computing the pixel $x_{i,j}$ as

- The pixel $x_{i,j}$ which is centered for the block $B \times B$ is extracted :
- Having the block which is extracted is replaced with the pixel $\tilde{x}_{i,j}$
- Scanning the block by creating $X_{i,j}$ and $y_{i,j}$
- By solving $y_{i,j} = X_{i,j}v_{i,j}$ the local predictor $v_{i,j}$ is computed
- Compute $\hat{x}_{i,j}$.

2). Compute the prediction error $e_{i,j}$.

3). If $|e_{i,j}| < T$ compute $x'_{i,j}$ with eq (2) or otherwise with eq(3).

4). If $x'_{i,j} \in [0, T-1]\, U[255-T, 255]$, replace $x_{i,j}$ $by\, x'_{i,j}$ if $x'_{i,j} \in [0,255]$.

5). Do not replace $x_{i,j}$ if $x'_{i,j} \nexists [0,255]$ and if the flag bit b=0 insert it to the next embeddable pixel.

SIMULATION RESULTS

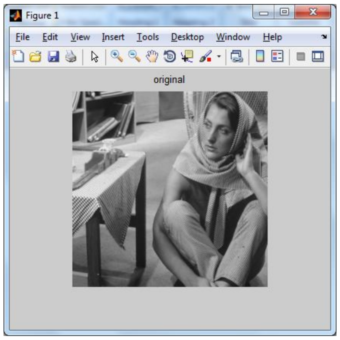

Figure 1: Input image

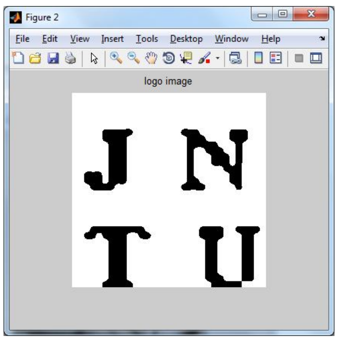

Figure 2: Logo image

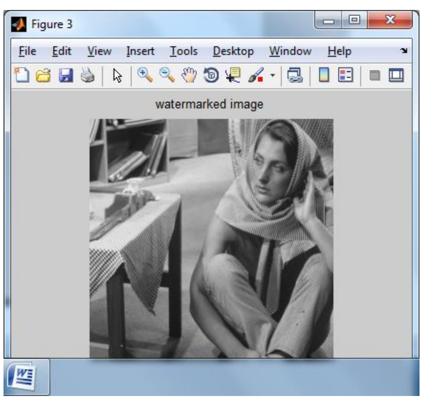

Figure 3: Watermarked image

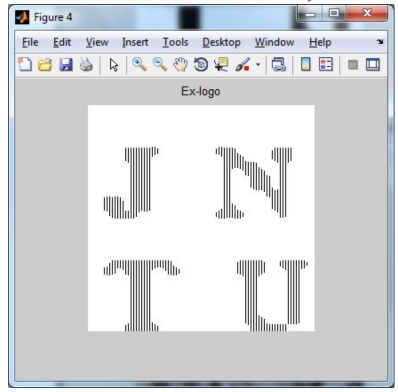

Figure 4: Ex-logo image

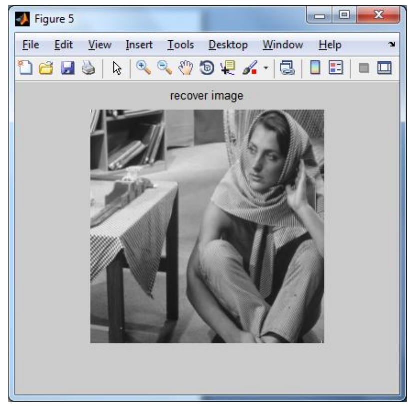

Figure 5: Input image

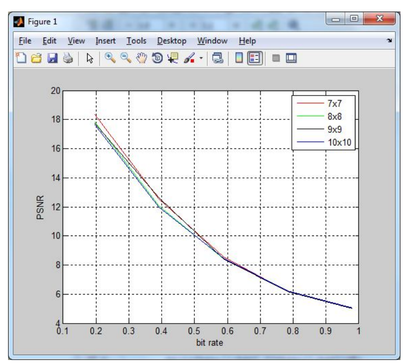

Figure 6: PSNR vs bit rate

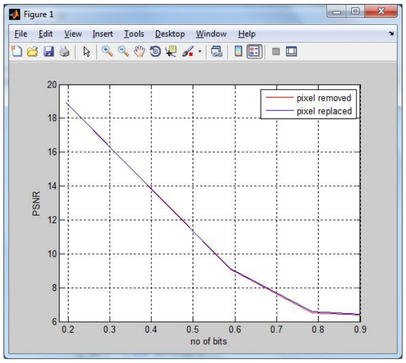

Figure 7: PSNR vs Number of bits

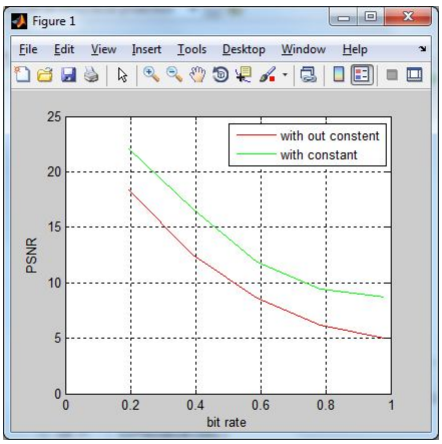

Figure 8: PSNR vs BER

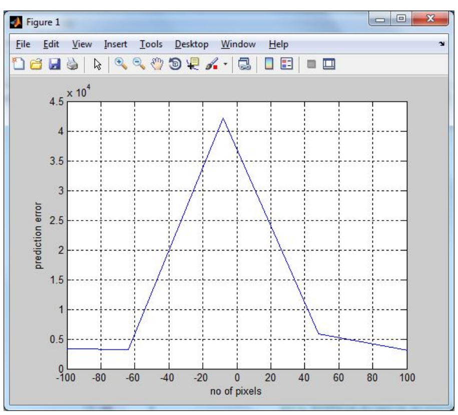

Figure 9: Prediction error vs number of pixels

CONCLUSION

The use of local prediction based reversible watermarking has been proposed. For each pixel, the least square predictor in a square block centered on the pixel is computed. The scheme is designed to allow the recovery of the same predictor at detection, without any additional information. The local prediction based reversible watermarking was analyzed for the case of four prediction contexts, namely the rhombus context and the ones of MED, GAP and SGAP predictors. The appropriate block sizes have been determined for each context. They are 12 × 12 (rhombus), 8 × 8 (MED), 10 × 10 (SGAP), 13 × 13 (GAP). The gain obtained by further optimization of the block size according to the image is negligible. The results obtained so far show that the local prediction based schemes clearly outperform their global least square and fixed prediction based counterparts. Among the four local prediction schemes analyzed, the one based on the rhombus context provides the best results. The results have been obtained by using the local prediction with a basic difference expansion scheme with simple threshold control, histogram shifting and flag bits.

REFERENCES

[1] M. Goljan, J. Fridrich, and R. Du, “Distortion-free data embedding for images,” Inf. Hiding, vol. 2137, pp. 27–41, Apr. 2001.

[2] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, “Lossless generalized LSB data embedding,” IEEE Trans. Imag. Process., vol. 14, no. 2, pp. 253–266, Feb. 2005.

[3] 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, Mar. 2006.

[4] C. C. Lin, W. L. Tai, and C. C. Chang, “Multilevel reversible data hiding based on histogram modification of difference images,” Pattern Recognit., vol. 41, no. 12, pp. 3582–3591, 2008.

[5] 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. 1129–1143, 2009.

[6] L. Luo, Z. Chen, M. Chen, X. Zeng, and Z. Xiong, “Reversible image watermarking using interpolation technique,” IEEE Trans. Inf., Forensics Security, vol. 5, no. 1, pp. 187–193, Mar. 2010.

[7] W. Hong, T.-S. Chen, and C.-W. Shiu, “Reversible data hiding for high quality images using modification of prediction errors, J. Syst. Softw., vol. 82, no. 11, pp. 1833–1842, 2009.

[8] C. Wang, X. Li, and B. Yang, “Efficient reversible image watermarking by using dynamical prediction-error expansion,” in Proc. 17th ICIP, 2010, pp. 3673–3676.

[9] J. Tian, “Reversible data embedding using a difference expansion,” IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 8, pp. 890–896, Aug. 2003.

[10] A. M. Alattar, “Reversible watermark using the difference expansion of a generalized integer transform,” IEEE Trans. Image Process., vol. 13, no. 8, pp. 1147–1156, Aug. 2004.