Available online at www.sciencedirect.com

Journal of Applied Research and Technology

Journal of Applied Research and Technology 15 (2018) 524–532

www.jart.ccadet.unam.mx

Original

A novel Diamond-Mean predictor for reversible watermarking of images

Muhammad Ishtiaq[a], Arfan Jaffar[b,*]

[a] *Department of Computer Science, FAST-National University of Computer & Emerging Sciences, Islamabad, Pakistan*
[b] *Al Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia*

Received 14 December 2015; accepted 3 June 2017
Available online 13 December 2017

Abstract

Reversible watermarking (RW) is the art of embedding secret information in the host image such that after extraction of hidden information, original image is also restored from the watermarked image. Prediction error expansion (PEE) is state of the art technique for RW. Performance of PEE methods depends on the predictor's ability to accurately estimate image pixels. In this paper, a novel Diamond-Mean (D-Mean) prediction mechanism is presented. The D-Mean predictor uses only D-4 neighbors of a pixel, i.e. pixels located at {east, west, north, south}. In the estimation process, apart from edge presence, its orientation and sensitivity is also taken into account. In experimental evaluations, the D-Mean predictor outperforms currently in use MED (median edge detector) and GAP (gradient adjusted predictor) predictors. For, standard test images of Lena, Airplane, Barbara and Baboon, an average improvement of 51.79 for mean squared PE and an average improvement of 0.4 for error-entropy than MED/GAP are observed. Payload vs imperceptibility comparison of the method shows promising results.

Keywords: Prediction; Error-expansion; Reversible watermarking; MED; GAP; D-Mean

1. Introduction

Internet is a highly convenient medium for transmitting and sharing of multimedia data. It rejuvenated businesses by attracting large number of customers. With the ease of access and increased business benefits, arises a greater challenge of ownership and authenticity of the media content. Watermarking of multimedia content is a very popular technique of coupling secret information with images and videos.

Image watermarking is the art of hiding 'secret message' in the image in such a way that it should not cause any visible distortion. It is one of the recent image protection methods, which provides security against illegal production and re-sharing of the copyright content. The process of hiding secret information is called as 'watermark embedding'. When an image, also known as 'cover-image', undergoes embedding process it is called as 'watermarked-image'. The main objective of watermarking is to make the watermarked-image alterations imperceptible to the image viewer, i.e. the quality of cover-image and watermarked-image should be visually identical. The amount of data that can be successfully embedded and retrieved in the cover-image is called as 'payload' of the embedding method. Increase in payload affects the imperceptibility negatively and vice versa.

Methods that can extract the watermark (hidden message) as well as completely retrieve the cover image from its watermarked-image are called as reversible watermarking (RW) methods (Caldelli, Filippini, & Becarelli, 2010). Recovery of the cover image is very important in medical imaging or in military applications where even a minute alteration/distortion is unacceptable. An MRI or CT scan image of a patient if not fully recovered might endanger the life of a patient by misleading the physician. Other applications of RW include remote sensing (Barni, Bartolini, Cappellini, Magli, & Olmo, 2001) and multimedia archive management (Park, 2014). Many reversible watermarking techniques has been proposed in the most recent literature (Kotvicha, Sanguansat, & Kasemsa, 2012; Shi & Xiao, 2013; Song, Li, Zhao, Hu, & Tu, 2015; Zhang, Qian, Feng, & Ren, 2014; Zhao & Feng, 2016).

[*] Corresponding author.
E-mail address: arfan.jaffar@ccis.imamu.edu.sa (A. Jaffar).
Peer Review under the responsibility of Universidad Nacional Autónoma de México.

https://doi.org/10.1016/j.jart.2017.06.001
1665-6423/

Tian (2003) introduced difference expansion transform for RW which was followed by prediction-error expansion (PEE) method by Thodi and Rodríguez (2004, 2007) they introduced histogram-shifting (HS) which significantly reduced location map (LM) size. Since then, many variations of PEE based methods (Kamstra & Heijmans, 2005; Luo, Chen, Chen, Zeng, & Xiong, 2010; Peng, Li, & Yang, 2012; Sachnev, Kim, Nam, Suresh, & Shi, 2009; Tai, Yeh, & Chang, 2009; Wang, Li, & Yang, 2010) were developed. Performance of these methods heavily rely on the predictor's ability to accurately predict image pixels. In most of the methods researchers investigated different embedding mechanisms while not much work is being done in developing new predictors. MED (Weinberger, Seroussi, & Sapiro, 2000) and GAP (Wu & Memon, 1997) are mostly used predictors by these methods.

In this paper a novel predictor for reversible watermarking is proposed. The predictor accurately models the flat and edge regions in an image hence better prediction of pixels which leads to less distortion of watermarked image. Improved embedding of watermark in conjunction with histogram shifting and D-Mean predictor led to gain higher imperceptibility levels for a given payload.

The paper organization is as follows: In the next section relevant reversible image hiding methods are discussed. In Section 3, MED and GAP are reviewed and novel D-Mean predictor is presented. Reversible watermarking method based on the proposed predictor is discussed in Section 4. Experimental setup and results are provided in Section 5 and conclusions are drawn in Section 6.

2. Related work

First expansion based method was presented by Tian (2003) and it achieved moderate payload with good image quality. In this method, image is divided into pairs of pixels and each pair based on its mean and difference value undergoes 1 bit of data embedding. Let (x_0, x_1) be the values of a pair of pixels, then integer mean and difference are denoted as $l = \lfloor (x_0 + x_1)/2 \rfloor$ and $h = x_1 - x_0$. To embed 1 bit watermark $b \in \{0, 1\}$ in h, it is expanded to $h' = 2h + b$ and watermarked values (x'_0, x'_1) are obtained using Eq. (1). For an easy reference, notations used in the paper are summarized in Table 1:

$$\begin{aligned} x'_0 &= l - \lfloor h'/2 \rfloor \\ x'_1 &= l + \lfloor (h' + 1)/2 \rfloor \end{aligned} \tag{1}$$

Table 1
Notations.

Symbol	Description
h	Difference of pixels
h'	Expanded difference after embedding of data
e	Prediction error
E	Expanded prediction error
x	Original pixel
$\hat{x}$	Estimate/prediction of a pixel
x'	Watermarked pixel
T	Embedding capacity threshold
T_S	Edge sensitivity threshold used in D-Mean predictor
S_{PE}	Entropy measured over PE of a predictor
I	Original Image
Θ_1 and Θ_2	Two disjoint sets of an image
Ψ	Pixels which can be modified twice without producing overflow
Φ	Pixels which can be modified once and will lead to overflow upon 2nd modification
Υ	Pixels which cannot be modified
b_h	hard bit used for testing of overflow
$\mathcal{D}_u$	Watermark data to be embedded in the image
$\mathcal{D}$	Payload which includes auxiliary information, LM and watermark data
A_i	Auxiliary information necessary for watermark extraction
T_v	Pixel selection threshold based on variance of the context
$v(\cdot)$	Variance of a set of pixels
N_{LM}	Length of location map
$\odot$	Concatenation operator

Pairs which are not suitable for data embedding are listed in location map (LM). For successful extraction of hidden data and restoration of cover image pixels, LM is also stored in the watermarked-image alongside embedded bits as overhead information. Originally, image pixel values range from 0 to 255 for an 8 bits per pixel representation. Expanded pixels which lie outside the permissible range are marked as unexpandable pairs in LM. Size of LM hampers the embedding capacity (EC); therefore reduction in its size is very important, hence researchers are using lossless compression methods to reduce its size. Using Tian method, 1 bit can be embedded in each pair of pixels while 1 bit for each pair is also required in LM hence space for data embedding is created by lossless compression of LM.

Alattar (2004) extended the idea of a pair to k pixels cell. Each cell has been used to hide $k - 1$ bits. For each cell one bit is required in LM, this helps decrease the size of LM to $(1/k)$th the image size. Unexpandable cells cannot be used for embedding due to problems of underflow or overflow (noted as overflow), so payload of the method is always less than $(k - 1)/k$ bpp (bits per pixel).

Location map is the bottleneck of reversible image hiding methods. Even compressed LM takes significant part of the payload. Thus, LM size determines the performance of a method. Later, authors investigated methods that generate small-size location map or no map at all. Lee, Yoo, and Kalker (2007) used block-based approach with integer-to-integer wavelet transform to hide data. An image of size $X \times Y$ is divided into blocks of size $N \times M$. The method produces relatively small LM and better exploits redundancy present in the sub-bands of wavelet transformed coefficients, hence, outperforms Tian (2003) and Alattar (2004).

Image pixels are highly correlated with neighboring pixels. Thodi and Rodríguez (2004) used MED predictor to predict image pixels from their neighborhood pixels. The prediction error (PE) is used to hide watermark bits instead of difference between pairs of pixels. In prediction, more than one pixel of the neighborhood is used which results in smaller PE. PE denoted as e of a pixel x having prediction value $\hat{x}$, is computed using Eq. (2). In Thodi and Rodríguez (2007), HS was incorporated in error expansion. In this method e is expanded for data embedding and the expanded error E is computed using Eq. (3):

$$e = x - \hat{x} \tag{2}$$

$$E = \begin{cases} 2e + b, & \text{if } \; e \in [-T, T] \\ e + T + 1, & \text{if } \; e > T \\ e - T, & \text{if } \; e < -T \end{cases} \tag{3}$$

Watermarked pixel x' is calculated using Eq. (4)

$$x' = E + \hat{x} \tag{4}$$

In the decoder, E and e are computed using Eqs. (5) and (6) respectively:

$$E = x' - \hat{x} \tag{5}$$

$$e = \begin{cases} \lfloor E/2 \rfloor & \text{if } \; E \in [-2T, 2T + 1] \\ E - T - 1, & \text{if } \; E > 2T + 1 \\ E + T, & \text{if } \; E < -2T \end{cases} \tag{6}$$

Embedded data bit b can be extracted using Eq. (7) and original pixel value is restored using Eq. (8):

$$b = E - 2\lfloor E/2 \rfloor \quad \text{if } \; E \in [-2T, 2T + 1] \tag{7}$$

$$x = \hat{x} + e \tag{8}$$

Kim, Sachnev, Shi, Nam, and Choo (2008) further used a simplified approach to reduce the size of LM and need for lossless compression was evaded. van der Veen, Bruekers, van Leest, and Cavin (2003) used companding technique for reversible watermarking of audio streams while van Leest, van der Veen, and Bruekers (2004) extended the same method for images. For data hiding Ni, Shi, Ansari, and Su (2006) used shifting of bins in histogram of image pixels. Yang, Schmucker, Funk, Busch, and Sun (2004) presented a generalized RW method for coefficients of integer discrete cosine transform. In another research Yang, Schmucker, Busch, Niu, and Sun (2005), they also used high frequency wavelet coefficients with histogram expansion. Luo et al. (2010) introduced an interpolation based RW method which has high fidelity but with relatively low capacity. Hong, Chen, Chang, and Shiu (2010) presented a high performance error expansion based RW method. Tsai, Hu, and Yeh (2009) compute the residual image from basic pixels and reference pixels in nonoverlapping blocks, high payload is achieved by multilevel embedding in residual image. Luo et al. (2010) used full context for interpolation of a pixel, interpolation error is expanded to embed data. Due to the fact that they have used 8 neighbors of a pixel context their results are significantly improved from methods that uses MED and GAP based prediction mechanism which use 3 and 7 pixels respectively.

Wang, Li, Yang, and Guo (2010) presented generalized version of the Tian's difference expansion algorithm. The difference is converted to extended integer transform and rather than using difference, mean of the block is computed and difference of the mean and pixel is used for embedding. It uses multiple embedding passes to achieve high EC. It had better imperceptibility performance than its predecessors. Hu, Lee, and Li (2009) proposed a major improvement in DE by incorporating MED and dual expansion. Dual expansion is carried out in two stages. In 1st stage the histogram bin is shifted to the right and in the 2nd stage the bin is slightly shifted back to the left, hence reducing/reversing the distortion caused by 1st stage embedding. The performance was further improved by introducing capacity control mechanism.

Peng et al. (2012) proposed a block based method. The image is first divided into non-overlapping blocks. Embedding capacity of each block is computed by using variance as the capacity control parameter. Variance is inversely proportional to the EC of a block. Location map for each block is computed and compressed using a lossless method. Watermark and LM are embedded into each block to generate watermarked image. This approach has better performance at higher payloads.

In most of the expansion based methods MED and GAP predictors are used. As these predictors were designed for lossless compression rather than data hiding, they cannot fully exploit existing correlations among image pixels for the purpose of watermarking.

3. Diamond-Mean predictor

In reversible watermarking methods PE histogram is modeled by Laplacian distribution. This is because of the spatial redundancy in image pixels. To get higher peaks in the center of the PE histogram, high performance predictors are being used in the prediction process. MED (Weinberger et al., 2000) and GAP (Wu & Memon, 1997) are the advanced predictors used in JPEG-LS and Context-based Adaptive Lossless Image Coding (CALIC). In the proposed reversible watermarking method D-Mean (Diamond-Mean) predictor is used. Before presenting the D-Mean predictor MED and GAP are reviewed.

3.1. Median edge detector

MED is one of the mostly used predictor in lossless compression and reversible watermarking. To calculate the prediction it uses forward context pixels. Pixel context is given in Fig. 1. Using MED, the estimate of a pixel $\hat{x}$ can be calculated using Eq. (9):

$$\hat{x} = \begin{cases} \min(x_s, x_e), & \text{if } \; x_{se} \geq \max(x_s, x_e) \\ \max(x_s, x_e), & \text{if } \; x_{se} \leq \min(x_s, x_e) \\ x_s + x_e - x_{se}, & \text{otherwise.} \end{cases} \tag{9}$$

The predictor selects x_s incase a vertical edge is detected, x_e incase of a horizontal edge and $x_s + x_e - x_{se}$ when no edge is detected. In Martucci (1990) they have denoted it as the median

	x_n		
x_w	x	x_e	x_{fe}
x_{sw}	x_s	x_{se}	
x_{fsw}	x_{fs}		

Fig. 1. Context of a pixel x for MED, GAP and D-Mean predictors.

of the set $\{x_s, x_e, x_s + x_e - x_{se}\}$. MED is being used by Thodi and Rodríguez (2007) and many other PE techniques. It efficiently detects the presence of an edge but major limitation is its inability to detect intensity of the edge.

3.2. Gradient adjusted predictor

GAP is more complex than MED. The prediction context is extended to 7 pixels. It not only detects the existence of an edge but also the intensity (weak, normal, strong). The direction of edge is detected by comparing local gradients with empirical thresholds. It performs better than MED at the expense of mathematical complexity. Estimate $\hat{x}$ of a pixel is calculated using Eq. (10):

$$\hat{x} = \begin{cases} x_e, & \text{if} \quad \Delta > 80 \\ (x_e + a)/2, & \text{if} \quad \Delta \in (32, 80] \\ (x_e + 3a)/4, & \text{if} \quad \Delta \in (8, 32] \\ a, & \text{if} \quad \Delta \in [-8, 8] \\ (x_s + 3a)/4, & \text{if} \quad \Delta \in [-32, 8) \\ (x_s + a)/2 & \text{if} \quad \Delta \in [-80, 32) \\ x_s & \text{if} \quad \Delta < -80 \end{cases} \tag{10}$$

where

$$\begin{aligned} \Delta &= \Delta_V - \Delta_H \\ \Delta_V &= |x_e - x_{se}| + |x_{sw} - x_{fsw}| + |x_s - x_{fs}| \\ \Delta_H &= |x_e - x_{fe}| + |x_{sw} - x_s| + |x_s - x_{se}| \\ a &= \frac{x_e + x_s}{2} + \frac{x_{sw} - x_{se}}{4} \end{aligned}$$

Both MED and GAP were originally designed for predictive coding in image/video compression. Due to limitations in coding for compression in the respective standards these predictors only use one side of a pixel context hence are unable to fully exploit the correlation between neighboring pixels.

3.3. Proposed Diamond-Mean predictor

A predictor that would better exploit correlation of pixels is presented. The proposed D-Mean (Diamond-Mean predictor) calculates the estimate $\hat{x} = \lfloor \alpha \rfloor$, while α is calculated using Eq. (11):

$$\alpha = \begin{cases} \dfrac{x_n + x_s}{2} & \text{if} \quad (|x_n - x_s| < T_S \quad \text{AND} \quad \{(\{x_n, x_s\} < Min_H) \quad \text{OR} \quad (\{x_n, x_s\} > Max_H)\}) \\ \dfrac{x_e + x_w}{2} & \text{elseif} \quad (|x_e - x_w| < T_S \quad \text{AND} \quad \{(\{x_e, x_w\} < Min_V) \quad \text{OR} \quad (\{x_e, x_w\} > Max_V)\}) \\ M\left(A'\right) & \text{otherwise} \end{cases} \tag{11}$$

where

$$\begin{aligned} Min_H &= \min(x_e, x_w), & Max_H &= \max(x_e, x_w) \\ Min_V &= \min(x_n, x_s), & Max_V &= \max(x_n, x_s) \\ A &= \{x_e, x_w, x_n, x_s\}, & M(B) &= \frac{b_1 + b_2 + \cdots + b_n}{n}, \quad b_i \in B \end{aligned}$$

$$A' = \{(a_2, a_3) \mid \{a_1 \leq a_2 \leq a_3 \leq a_4\} \wedge a_i \in A\}$$

Set A consists of four neighbors of a pixel, i.e. $\{x_e, x_w, x_n, x_s\}$ as defined in Fig. 1. A' contains the 2nd and 3rd largest elements of A. M function computes the mean of a set while $\lfloor \alpha \rfloor$ returns floor of α, i.e. the largest integer less than α.

For each pixel, the predictor first checks if neighboring pixels (x_n, x_s) or (x_e, x_w) are close enough to be on the same edge, an edge sensitivity threshold T_S is used for this purpose. For vertical edges, if both x_n and x_s are less than Min_H (a dark edge on a bright background) or both are greater than Max_H (bright edge on a dark background) then there exist an edge which passes through x_n and x_s (a vertical edge). Horizontal edges are detected in the same manner and current pixel prediction is calculated. If there exist no definite edge, mean of 2nd and 3rd largest pixels from $\{x_e, x_w, x_n, x_s\}$ is taken as prediction. Using this method the predicted value may not be an integer, which must be converted to an integer if it is to be used in RW, for that purpose floor of the predicted value is calculated. The PE histogram of MED, GAP and D-Mean predictors is compared in Fig. 2. Significant improvement is observed for all four standard images, i.e. Lena, Airplane, Barbara and Baboon. Surge in the histogram peaks at 0 and short tail of PE for D-Mean confirms the superior performance of the proposed D-Mean predictor over MED and GAP methods.

Quantitative measures of predictor's performance are mean squared prediction error (MSPE) and entropy of PE (S_{PE}). Watermarking method's performance is inversely proportional to both the measures, i.e. smaller MSPE and entropy of PE (S_{PE}) leads to better imperceptibility results. MSPE is computed using Eq. (12) and S_{PE} using Eq. (13). Here, e is PE computed using Eq. (2), N_e is the length of error vector, and $pr(e)$ is the probability of error, e. In Table 2, predictors are compared on the basis of MSPE. For all the test images D-Mean yields the least MSPE than MED and GAP. Entropy comparison of PE is provided in Table 3 and again D-Mean has outperformed other predictors. Overall, average performance of D-Mean is also better for both MSPE and S_{PE}:

$$\text{MSPE} = \frac{\sum_{e \in PE} (e)^2}{N_e} \tag{12}$$

$$S_{\text{PE}} = -\sum_{e \in PE} pr(e) \times \log(pr(e)) \tag{13}$$

With D-Mean predictor two-stage image traversal mechanism is used. Pixels classification into 2 sets is provided in Fig. 3. Pixels of an image I having positions (i, j) are assigned to disjoint sets Θ_1 and Θ_2. As both sets are disjoint, pixels in Θ_1 can be used in estimation of pixels in Θ_2 and vice versa.

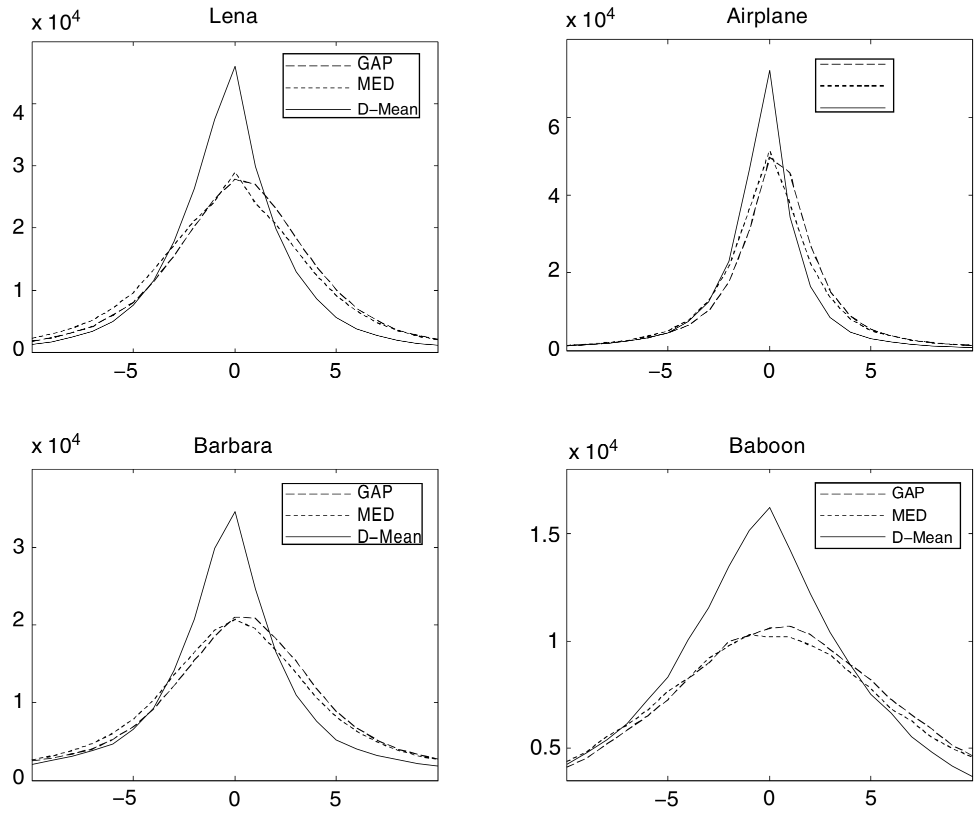


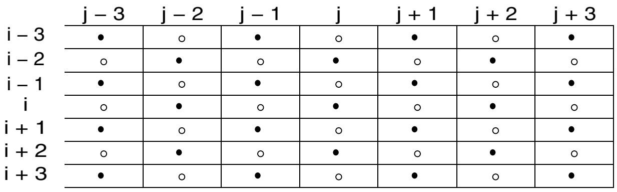
Fig. 2. Prediction-error histograms on standard images. *X*-axis and *y*-axis represent PE and its occurrence respectively.

Table 2
Comparison of predictors based on MSPE for standard images.

Test image	Predictor		
	GAP	MED	D-Mean
Lena	47.27	51.14	23.90
Airplane	33.70	33.98	19.68
Barbara	234.84	269.90	176.33
Baboon	306.82	319.37	195.57
Average	155.66	168.60	103.87

Table 3
Comparison based on entropy of the PE, S_{PE}.

Test image	Predictor		
	GAP	MED	D-Mean
Lena	4.55	4.48	3.99
Airplane	3.99	3.95	3.57
Barbara	5.48	5.42	5.02
Baboon	6.06	6.01	5.65
Average	5.02	4.96	4.56



	j – 3	j – 2	j – 1	j	j + 1	j + 2	j + 3
i – 3	●	○	●	○	●	○	●
i – 2	○	●	○	●	○	●	○
i – 1	●	○	●	○	●	○	●
i	○	●	○	●	○	●	○
i + 1	●	○	●	○	●	○	●
i + 2	○	●	○	●	○	●	○
i + 3	●	○	●	○	●	○	●

Fig. 3. Notation of pixels in two disjoint sets where $\bullet \in \Theta_1$ and $\circ \in \Theta_2$.

4. Proposed method

Proposed watermarking method uses high performance D-Mean prediction mechanism. Pixel's PE is expanded to embed watermark bits. Some of the pixels that may result in overflow due to embedding are left unmodified in the watermarked image. Intelligent embedding is being used to only select pixels that have small PE which further improved imperceptibility of the proposed method. Illustration of the proposed method is provided in Fig. 4.

Given an image I of size $M \times N$ having pixels $\{(i, j)|1 \leq i \leq M, 1 \leq j \leq N\}$ is processed in two stages. Θ_1 pixels are processed first. Based on pixel's modifiability 3 sets of pixels are defined for Θ_1, unambiguously modifiable (noted as Ψ), ambiguously modifiable (noted as Φ) and non-modifiable (noted as Υ) pixels. Pixel's PE is calculated using Eq. (2) and tested by modification with hard bit (b_h) using Eq. (3), modified pixel x^{t_1} is calculated using Eq. (4). If x^{t_1} satisfies $(x^{t_1} < 0 \vee 255 < x^{t_1})$ it is assigned to Υ otherwise x^{t_1} is further checked for modification with b_h in the same manner and modified pixel x^{t_2} is calculated. If x^{t_2} satisfies $(x^{t_2} < 0 \vee 255 < x^{t_2})$ it is assigned to Φ, otherwise it is assigned to Ψ. In Eq. (3), b_h is being used for overflow checking in place of b and is defined as:

$$b_h = \begin{cases} 0 & \text{if} \quad e < 0 \\ 1 & \text{if} \quad e \geq 0 \end{cases} \tag{14}$$

LM is maintained for pixels that may result in overflow due to embedding. Here an approach similar to Kim et al. (2008),

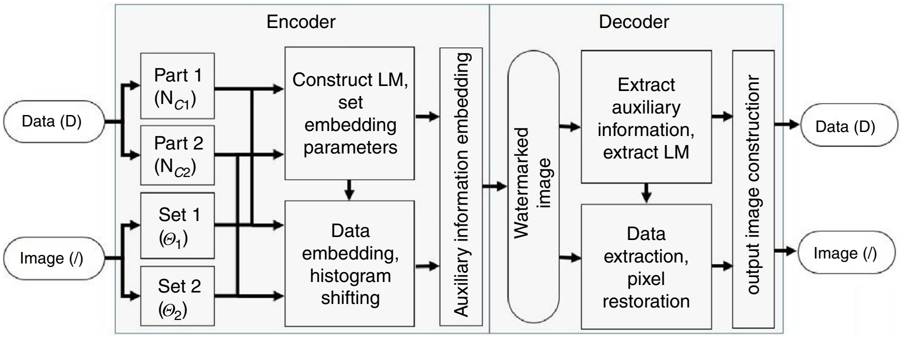

Encoder
Decoder
Data (D)
Image (I)
Part 1 (N$_{C1}$)
Part 2 (N$_{C2}$)
Set 1 (Θ_1)
Set 2 (Θ_2)
Construct LM, set embedding parameters
Data embedding, histogram shifting
Auxiliary information embedding
Watermarked image
Extract auxiliary information, extract LM
Data extraction, pixel restoration
output image constructionr
Data (D)
Image (I)

Fig. 4. System block diagram.

is being used in recording of LM. Pixels in Ψ can be unambiguously interpreted in the decoder, hence these pixels do not require the use of LM. Pixels of the set Φ are only modified with hard bit and pixels of the set Υ are not modified at all. In LM Φ and Υ pixels are noted 0 and 1 respectively. In our case LM is a 1D binary string and its length is defined as $|\Phi| + |\Upsilon|$, here $|\cdot|$ is a cardinal number of the set.

Payload of the proposed method is controlled by EC parameter T as used in Eq. (3). Due to iterative nature of existing RW algorithms, selection of T is a computationally expensive task because they had to compress LM for each value of T. We are using a simple methodology to record LM which significantly reduces its size hence compression is not required. A simple procedure to compute T is followed.

Let N_c be the length of data $\mathcal{D}_u$ to be embedded in the image. In the proposed two stage processing, N_{c1} and N_{c2} are the sizes of data embedded in stage 1 and 2 respectively. N_{c1} and N_{c2} are computed as follows:

$$N_{c1} = \left\lfloor \frac{N_c}{2} + 0.5 \right\rfloor \tag{15}$$

$$N_{c2} = \left\lfloor \frac{N_c}{2} \right\rfloor \tag{15}$$

Auxiliary information A_i is required for successful extraction of watermark in the decoder. A_i is also embedded as part of the payload in the image. $|A_i|$ is 68 bits, 8 for EC threshold T, 24 for pixel selection threshold T_υ and 18 for LM length and 18 for last modified pixel position according to $\lceil \log_2(M \times N) \rceil$ for 512×512 size image. For each stage Θ_1 and Θ_2 three sets $\Upsilon(T)$, $\Phi(T)$ and $\Psi(T)$ can be defined for $T \in \{1, 2, \ldots, 255\}$. Data embedding in each stage is possible subject to Eq.(16). If T which satisfies Eq.(16) then it is not possible to hide data of size N_{ci} in each stage.

$$\begin{aligned} &\text{If } \exists \;\; T \in [1, 255] : \{|\Psi(T)| \geq \mathcal{D}\} \\ &\mathcal{D} = |A_i| + |\Phi(T)| + |\Upsilon(T)| + N_{ci} \end{aligned} \tag{16}$$

Smallest T for which, $|\Psi(T)| \geq \mathcal{D}$ is selected as EC threshold.

Maximum data that can be embedded in a single stage is $|\Psi|$. Let N_{ci} be the length of data desired to be embedded in each stage. If $\mathcal{D} < |\Psi|$ then a subset Ψ_s of pixels can be selected from Ψ for embedding. Error due to embedding of PEE based methods is directly proportional to the magnitude of PE hence Ψ_s must contain pixels having small PE. Here a pixel selection mechanism is defined based on variance of the neighboring pixels. Let Ψ_s be defined as:

$$\Psi_s = \{x | x \in \Psi \wedge \upsilon(x_N) < T_\upsilon\} \tag{17}$$

Here x_N for a pixel x at location (i, j) is defined as a set, i.e. $\{(i-1, j), (i, j-1), (i, j+1), (i+1, j)\}$ also mentioned as x_n, x_w, x_e and x_s of a pixel context in Fig. 1, while $\upsilon(x_N)$ is defined as:

$$\begin{aligned} \upsilon(x_N) &= \frac{(x_1 - x_\mu)^2 + (x_2 - x_\mu)^2 + \cdots + (x_n - x_\mu)^2}{n} \\ x_\mu &= \frac{x_1 + x_2 + \cdots + x_n}{n} \end{aligned} \tag{18}$$

T_υ is a threshold for selection of pixels having small PE. In this way, proposed method tends to select the pixels having small error and hence better visual quality of the watermarked image is obtained.

4.1. Watermark embedding

In this sub-section watermark embedding procedure is listed. Step by step description of the proposed method is followed. Embedding data $\mathcal{D}_u$ is divided into 2 parts, $\mathcal{D}_{u1}$ and $\mathcal{D}_{u2}$ using Eq. (15). Data streams $\mathcal{D}_{u1}$, $\mathcal{D}_{u2}$ contains $\mathcal{D}_u\{1 \cdots N_{c1}\}$ and $\mathcal{D}_u\{N_{c1} + 1 \cdots N_c\}$ respectively. $\mathcal{D}_{u1}$ is embedded in pixels from Θ_1 while Θ_2 will be embedded with $\mathcal{D}_{u2}$. For each set, Θ_1 and Θ_2, repeat the following steps:

Step 1: Select the EC threshold T, which satisfies Eq. (16).
Step 2: Compute υ for all pixels using Eq. (18) and arrange pixels in increasing magnitude with respect to υ.
Step 3: Skip the initial 68 pixels. Among the remaining, using Eq. (17) select a subset Ψ_s from Ψ.
Step 4: All pixels of the set, Φ will be modified with hard-bit using Eq. (14). Pixels of the set, Υ will not be modified but noted in LM. In LM Φ and Υ pixels are noted as '0' and '1' respectively. Size of LM, N_{LM} will be $|\Phi(T)| + |\Upsilon(T)|$
Step 5: Collect LSB of initial 68 pixels and construct the payload as $\mathcal{D}_{up} = LSB \odot LM \odot \mathcal{D}_{ui}$. Where $\odot$ is a concatenation operator.

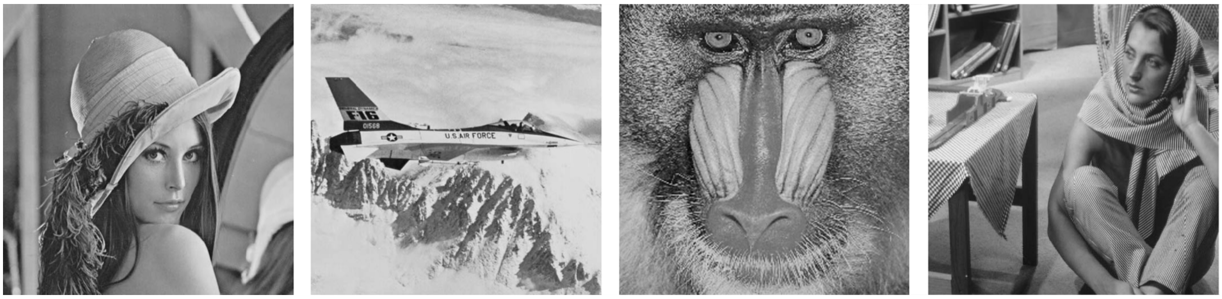


Fig. 5. 512 × 512 sized, grayscale test images. From left to right are, Lena, Airplane, Baboon and Barbara.

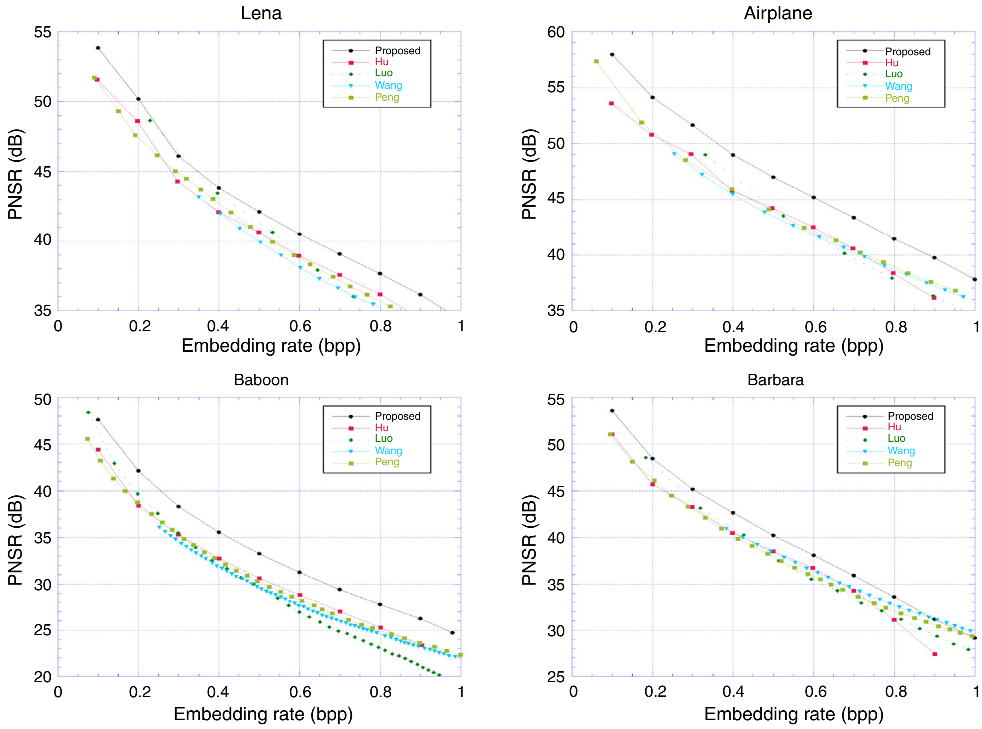


Fig. 6. Performance comparison between proposed method and that of Hu et al. (2009), Luo et al. (2010), Wang, Li, Yang, and Guo (2010) and Peng et al. (2012) over standard images.

Step 6: Each pixel of the set Ψ_s is used to embed 1 data bit from $\mathcal{D}_{up}$ using Eq. (4). Estimate of the pixel, $\hat{x}$ is computed using proposed D-Mean predictor.

Step 7: Auxiliary information A_i, consists of 68 bits and contains, T, T_υ, N_{LM} and position of the last modified pixel. Replace LSBs of the initial 68 pixels with A_i.

The above steps 1–7 are repeated for each, Θ_1 and Θ_2. In the second stage for the prediction of pixels modified pixels of the set Θ_1 are used.

4.2. Watermark extraction

In the decoder watermarked image is processed in the reverse order. First, pixels of stage-II, Θ_2 are decoded and restored. Then, Θ_1 pixels are processed. Step-by-step procedure of decoding is described as:

Step 1: Extract LSBs of the initial 68 pixels and compute the parameters of T, T_υ, and N_{LM} and location of the last modified pixel.

Step 2: As done in the encoder, compute υ for all pixels using Eq. (18) and arrange pixels in the increasing magnitude with respect to υ.

Step 3: Make a set Ψ_e of pixels having $\upsilon < T_\upsilon$.

Step 4: Check each pixel of the set Ψ_e for expansion with hard bit, using Eq. (3) and Eq. (4). This step is performed until we reach the last modified pixel of the set.

a. If the pixel results in overflow, check LM, it belongs to one of the sets, Φ or Υ. If corresponding bit in the LM is 0, it belongs to the set, Φ. The pixel is restored using the Eqs. (5) and (8). The bit stored in the current pixel is discarded and is not recorded in the data.

b. If the pixel did not resulted in overflow then stored data bit is extracted and the pixel is restored. The

Table 4
Comparison of PSNR (in dB) at an embedding rate of 0.5 bpp.

	Proposed	Hu (2009)	Luo (2010)	Wang (2010)	Peng (2012)
Lena	42.13	40.65	41.05	39.9	40.6
Airplane	46.98	44.26	44.21	43.07	43.52
Barbara	40.23	38.51	37.93	38.44	37.91
Baboon	33.26	30.63	29.46	29.48	30.29
Average	40.65	38.51	38.16	37.72	38.08

data bit from the pixel is extracted using Eq. (5) and (7) and recorded in the output data. While pixel restoration is carried out using Eq. (8).

Step 5: Restore the LSBs of initial 68 pixels from the extracted data.

After restoration of pixels of Θ_2. The set Θ_1 is processed in the same manner. Data extracted from both the stages is combined to make the extracted message. Block diagram of the embedding and extraction process is provided in the Fig. 4.

5. Experimental results

The method is assessed by comparing with Hu et al. (2009), Luo et al. (2010), Wang, Li, Yang, and Guo (2010) and one of the recent works by Peng et al. (2012). Results are compiled over the standard grayscale images of Lena, Airplane, Barbara and Baboon of size 512 × 512. The images are shown in the Fig. 5. The imperceptibility results are given in Fig. 6. The superior performance of the proposed method can be observed over all the test images. Improvement in results is primarily due to the use of high performance D-Mean predictor in the prediction mechanism. Maximum payload that can be embedded in a single pass of the proposed method is at most 1 bpp. Hence, for large payload size, the method should be applied iteratively.

Selection of appropriate value for Edge sensitivity threshold T_S, is important for predictor performance. In this work the T_S is empirically determined and a value of 5 is used. It may be observed that the EC threshold controls the dual behavior of the predictor. For smaller values of T_S the predictor will act like median of four pixels, while large value of the threshold will make the predictor similar to mean predictor. An image, I after watermark embedding phase is noted as, I'. Imperceptibility test of the methods is carried out based on PSNR for a specific bpp (bits per pixel). Higher PSNR is always desired. It can be computed using Eq. (19). Comparison based on PSNR measure of watermarking methods is provided in Fig. 6. Imperceptibility test for a specific bpp of 0.5 is provided in Table 4. An average of 2.14 (dB) improvement in PSNR can be observed in comparison to Peng et al. (2012):

$$\text{PSNR} = 20\log_{10}\left(\frac{MAX_f}{\sqrt{MSE}}\right)$$
$$\text{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left[I(i,j) - I'(i,j)\right] \tag{19}$$

MAX_f for an 8 bit grayscale image is 255.

6. Conclusion

In this paper, a novel D-Mean predictor is proposed for PEE based reversible watermarking methods. D-Mean is state-of-art predictor which outperformed MED and GAP. D-Mean better models the presence/absence of an edge and uses 4 pixels around a pixel context which led to reduction in prediction-error. The method is simple and can be easily incorporated in the existing systems. Due to reduced overflow situation, location map shrinks to smaller sizes. The advantage of using D-Mean proved useful and results are improved for PEE based method. Future researchers may like to devise a scheme to auto-tune the parameters used in the D-Mean predictor and watermark embedding routine.

Conflict of interest

The authors have no conflicts of interest to declare.

Acknowledgements

First author would like to thank Higher Education Commission, Pakistan for funding a PhD fellowship under indigenous scheme through the grant no. 074-0941-PS4-070.

References

Alattar, A. M. (2004). Reversible watermark using the difference expansion of a generalized integer transform. *IEEE Transactions on Image Processing*, *13*(8), 1147–1156.

Barni, M., Bartolini, F., Cappellini, V., Magli, E., & Olmo, G. (2001). Watermarking-based protection of remote sensing images: Requirements and possible solutions. *Mathematics of data/image coding, compression, and encryption IV, with applications* (Vol. 4475) (pp. 191–202).

Caldelli, R., Filippini, F., & Becarelli, R. (2010). Reversible watermarking techniques: An overview and a classification. *EURASIP Journal on Information Security*, *2010*, 1–19.

Hong, W., Chen, T.-S., Chang, Y.-P., & Shiu, C.-W. (2010). A high capacity reversible data hiding scheme using orthogonal projection and prediction error modification. *Signal Processing*, *90*(11), 2911–2922.

Hu, Y. J., Lee, H. K., & Li, J. W. (2009). DE-based reversible data hiding with improved overflow location map. *IEEE Transactions on Circuits and Systems for Video Technology*, *19*(2), 250–260.

Kamstra, L., & Heijmans, H. J. (2005). Reversible data embedding into images using wavelet techniques and sorting. *IEEE Transactions on Image Processing*, *14*(12), 2082–2090.

Kim, H.-J., Sachnev, V., Shi, Y.-Q., Nam, J., & Choo, H.-G. (2008). A novel difference expansion transform for reversible data embedding. *IEEE Transactions on Information Forensics and Security*, *3*(3), 456–465.

Kotvicha, A., Sanguansat, P., & Kasemsa, M. L. K. (2012). Expand variance mean sorting for reversible watermarking. *International Journal of Computer and Communication Engineering*, *1*(3).

Lee, S., Yoo, C. D., & Kalker, T. (2007). Reversible image watermarking based on integer-to-integer wavelet transform. *IEEE Transactions on Information Forensics and Security*, 321–330.

Luo, L., Chen, Z., Chen, M., Zeng, X., & Xiong, Z. (2010). Reversible image watermarking using interpolation technique. *IEEE Transactions on Information Forensics and Security*, *5*(1), 187–193.

Martucci, S. A. (1990). Reversible compression of HDTV images using median adaptive prediction and arithmetic coding. In *IEEE international symposium on circuits and systems* (pp. 1310–1313).

Ni, Z., Shi, Y.-Q., Ansari, N., & Su, W. (2006). Reversible data hiding. *IEEE Transactions on Circuits and Systems for Video Technology*, *16*(3), 354–362.

Park, S. B. (2014). Security requirements for multimedia archives. *Advances in Multimedia*, *2014*.

Peng, F., Li, X., & Yang, B. (2012). Adaptive reversible data hiding scheme based on integer transform. *Signal Processing*, *92*(1), 54–62.

Sachnev, V., Kim, H. J., Nam, J., Suresh, S., & Shi, Y. Q. (2009). Reversible watermarking algorithm using sorting and prediction. *IEEE Transactions on Circuits and Systems for Video Technology*, *19*(7), 989–999.

Shi, X, & Xiao, D. (2013). A reversible watermarking authentication scheme for wireless sensor networks. *Information Sciences*, *1*(240), 173–183.

Song, G., Li, L., Zhao, J., Hu, J., & Tu, H. (2015). A reversible video steganography algorithm for MVC based on motion vector. *Multimedia Tools Application*, *74*(11), 3559–3782.

Tai, W. L., Yeh, C. M., & Chang, C. C. (2009). Reversible data hiding based on histogram modification of pixel differences. *IEEE Transactions on Circuits and Systems for Video Technology*, *19*(6), 906–910.

Thodi, D. M., & Rodríguez, J. J. (2004). Prediction-error based reversible watermarking. In *International conference on image processing (ICIP)* (pp. 1549–1552).

Thodi, D. M., & Rodríguez, J. J. (2007). Expansion embedding techniques for reversible watermarking. *IEEE Transactions on Image Processing*, *16*(3), 721–730.

Tian, J. (2003). Reversible data embedding using a difference expansion. *IEEE Transactions on Circuits and Systems for Video Technology*, *13*(8), 890–896.

Tsai, P., Hu, Y.-C., & Yeh, H.-L. (2009). Reversible image hiding scheme using predictive coding and histogram shifting. *Signal Processing*, *89*(6), 1129–1143.

van der Veen, M., Bruekers, F., van Leest, A., & Cavin, S. (2003). High capacity reversible watermarking for audio. In E. J. JIII Delp, & P. W. Wong (Eds.), *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* (Vol. 5020) (pp. 1–11).

van Leest, A. J., van der Veen, M., & Bruekers, F. (2004). Reversible watermarking for images. In E. J. Delp, & P. W. Wong (Eds.), *Security, steganography, and watermarking of multimedia contents, SPIE, Proceedings of SPIE* (Vol. 5306) (pp. 374–385).

Wang, C., Li, X., & Yang, B. (2010). High capacity reversible image watermarking based on integer transform. In *International conference on image processing (ICIP)* (pp. 217–220). IEEE.

Wang, X., Li, X., Yang, B., & Guo, Z. (2010). Efficient generalized integer transform for reversible watermarking. *IEEE Signal Processing Letters*, *17*(6), 567–570.

Weinberger, M. J., Seroussi, G., & Sapiro, G. (2000). The loco-i lossless image compression algorithm: Principles and standardization into JPEG-LS. *IEEE Transactions on Image Processing*, *9*(8), 1309–1324.

Wu, X., & Memon, N. D. (1997). Context-based, adaptive, lossless image coding. *IEEE Transactions on Communications*, *45*(4), 437–444.

Yang, B., Schmucker, M., Funk, W., Busch, C., & Sun, S.-H. (2004). Integer DCT-based reversible watermarking for images using companding technique. In *Security, steganography, and watermarking of multimedia contents* (pp. 405–415).

Yang, B., Schmucker, M., Busch, C., Niu, X., & Sun, S. (2005). Approaching optimal value expansion for reversible watermarking. In *Proceedings of the 7th workshop on multimedia and security* (pp. 95–102). New York, NY, USA: ACM.

Zhang, X., Qian, Z., Feng, G., & Ren, Y. (2014). Efficient reversible data hiding in encrypted images. *Journal of Visual Communication and Image Representation*, *25*(2), 322–328.

Zhao, J. Z.-T. L., & Feng, B. (2016). A novel two-dimensional histogram modification for reversible data embedding into stereo h.264 video. *Multimedia Tools Application*, *75*(10), 5959–5980.