# EFFICIENT ADAPTIVE PREDICTION BASED REVERSIBLE IMAGE WATERMARKING

*Sunil Prasad Jaiswal, *Oscar C. Au, *Vinit Jakhetiya, *Yuanfang Guo, *†Anil K. Tiwari, *Kong Yue

* The Hong Kong University of Science and Technology, Hong Kong
*† Indian Institute of Technology, Rajasthan, India
Email: *{ spjaiswal,eeau,vjakhetiya,eeandyguo,ykongaa}@ust.hk,*†akt@iitj.ac.in,

## ABSTRACT

In this paper, we propose a new reversible watermarking algorithm based on additive prediction-error expansion which can recover original image after extracting the hidden data. Embedding capacity of such algorithms depend on the prediction accuracy of the predictor. We observed that the performance of a predictor based on full context prediction is preciser as compared to that of partial context prediction. In view of this observation, we propose an efficient adaptive prediction (EAP) method based on full context, that exploits local characteristics of neighboring pixels much effectively than other prediction methods reported in literature. Experimental results demonstrate that the proposed algorithm has a better embedding capacity and also gives better Peak Signal to Noise Ratio (PSNR) as compared to state-of-the-art reversible watermarking schemes.

***Index Terms*—** Reversible image watermarking, adaptive prediction, embedding capacity.

## 1. INTRODUCTION

Reversible watermarking received much popularity in recent years because of copyright protection of data. Reversible watermarking means embedding a specific information into the cover media in such a way that it can recover the original cover media at the decoder. Reversible watermarking is useful for important media, such as medical and military image, because it need to recover completely without any loss. Reversible watermarking are also useful in other applications such as image and video coding [1].

Several reversible watermarking algorithms have been proposed in literature in recent years. Reversible Image watermarking (RIW) algorithms can be classified into three categories [2]: data compression [3] based RIW, histogram based shifting methods [4]-[6] and difference expansion (DE) [7]-[10] based RIW. Algorithms based on DE method have achieved better embedding capacity with low computation cost as compared to other techniques. Algorithms in [11]-[13] have extended DE based methods by embedding watermarking data into prediction error instead of difference between paired pixels as done in DE scheme and leads to achieve even high embedding capacity.

All the above mentioned algorithms based on DE are constrained to operate on an unsymmetrical pixel space i.e, they use only causal pixels for prediction as decoder do not have information of non-causal pixels. However, Ming $et.al$ [11] proposed an algorithm based on full context prediction (FCP) that makes better prediction as compared to previous algorithms. But their method fails to exploit neighborhood pixels for effective prediction as their algorithm is non-adaptive and can work only on low detailed images. Another reason for poor efficiency of FCP is that it uses a lower order predictor model that fails to exploit effective relationship between the unknown pixel and its neighboring pixels.

In view of above mentioned problems, we propose an efficient adaptive prediction (EAP) algorithm based on full context prediction. We propose a predictor of $8^{th}$ order as against $4^{th}$ order proposed in [11]. Besides of increasing the predictor order, the predictor is also adaptive as against the fixed one in [11]. The rest of the paper is organized as follows. In Section 2, additive prediction error expansion is explained. The Proposed algorithm is discussed in section 3. Section 4 includes results analysis and concluding remarks in section 5.

## 2. ADDITIVE PREDICTION ERROR EXPANSION

Watermarked information bit (0 or 1) is embedded into original image using additive predictor error expansion (PEE) method [11]-[13] as discussed below.

First predicted value $P(n)$ of original pixel $I(n)$ is estimated by a prediction technique and then prediction error $e(n)$ is obtained as, $e(n) = I(n) - P(n)$. The watermark is then embedded into the residual/error pixel ($e(n)$) by traditional additive PEE method as given in (1).

$$e_w(n) = \begin{cases} e(n) + b \times sign(e(n)) \times Q & -Q \le e(n) < \text{Q} \\ e(n) + b \times Q & \text{if } e(n) = 0 \\ e(n) + sign(e(n)) \times Q & \text{otherwise} \end{cases} \quad (1)$$

Here $b$ is to-be-embedded bit (0 or 1), $sign(e(n))$ implies $+1$ if $e(n)$ is positive and $(-1)$ if negative. $Q$ is order of embedding i.e, $Q = 1$ implies single-layer embedding and multilayer embedding means for $Q > 1$. Thus after PEE, watermarked pixels can be given by, $I_w(n) = e_w(n) + P(n)$.

At decoder, we can restore original image ($\mathbf{I}$) and watermark information from watermarked image ($\mathbf{I_w}$). First, prediction value is estimated (as done at encoder) and thus $e_w(n)$ can be calculate via, $e_w(n) = I_w(n) - P(n)$. Once we get $e_w$, the residual image ($e$) and hidden bit ($b$) can be obtained using (2) given below.

$$e(n) = \begin{cases} e_w(n) \text{ and b = 0} \\ \quad \text{if } -Q \leq e_w(n) < \text{Q} \\ \\ e_w(n) - sign(e_w(n)) \times Q \text{ and b=1} \\ \quad \text{if } -2 \times Q \leq e_w(n) < -Q \\ \quad \text{or } Q \leq e_w(n) < 2 \times Q \\ \\ e_w(n) - sign(e_w(n)) \times Q \\ \quad \text{otherwise} \end{cases} \tag{2}$$

Then original pixel is estimated by, $I(n) = P(n) + e(n)$. Thus, original image and watermark information can be recovered completely using additive PEE.

At encoder, relationship between $e(n)$ and $e_w(n)$ is given by (1) and distortion caused by watermarked pixel is $I(n) - I_w(n) = (P(n)+e(n))-(P(n)+e_w(n)) = e(n)-e_w(n) \leq Q$. Thus, distortion is tiny and changes in the watermarked image is imperceptible for smaller values of $Q$. From (1), we observe that bits ($b$) are embedded into residue samples ($e(n)$) having values in the range of [-$Q$,$Q$]. Thus, better the accuracy of the predictor, more will be the embedding capacity.

## 3. PROPOSED ALGORITHM

Aim of the proposed EAP algorithm is to increase the prediction accuracy so as to achieve higher embedding capacity. For this, we propose a symmetrical predictor structure which is adopted from [11], briefly describes as below.

Based on the coordinate position of the pixels of given image ($\mathbf{I}$), label its pixels by four symbols. Let $A(n)$, $B(n)$, $C(n)$ and $D(n)$ are symbols representing pixels of $\mathbf{I}$ at even-even positions, even-odd positions, odd-even positions and odd-odd positions respectively with the first pixel of the image considered to be at (1,1) i.e. odd-odd position. By using these symbols, a part of given image is shown in Fig. 1(a). Motivation behind classifying the pixels into four parts as per the coordinate positions, is to make use of a symmetrical predictor structure. After the classification, rest of the prediction process involved is described in four phases as given below.

### 3.1. Prediction and watermarking of type 'A' pixels

In first phase, only pixels denoted by $\mathbf{A}$ shown in Fig. 1(a) are predicted (i.e, pixels of $\mathbf{I}$ at even-even positions) and then watermarked by PEE method. Prediction method proposed by $FCP$ [11] is $4^{th}$ order predictor and is given as

$$\hat{A}(n) = (C(n-3)+C(n+3)+B(n-1)+B(n+1))/4. \tag{3}$$

Here, $\hat{A}(n)$ is the predicted value of $A(n)$ shown in Fig. 1(b). Though the predictor utilizes both causal and non-causal pixels for prediction, but it is ineffective for pixels lying on edges

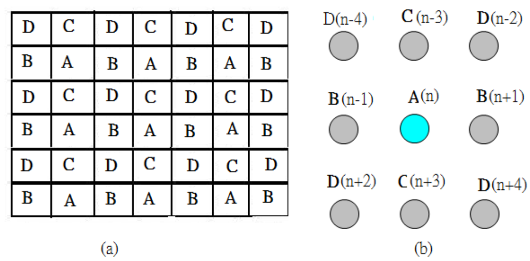


**Fig. 1**. (a) Labeling of original image ($\mathbf{I}$) based on spatial co-ordinates, (b) Prediction analysis of $A(n)$ (First Phase)

or for images having edge/rough texture regions, as averaging operation in (3) assigns equal weights to all the neighboring pixels used in prediction. In such case, amplitude of prediction error samples will be large and hence embedding capacity decreases. Also, FCP does not exploit other 4 neighboring pixels which are at diagonal to $A(n)$ as shown in Fig. 1(b).

In an image, generally a pixel has a value close to its neighbors and it depends on the local structure (edge direction etc.) of the image. Thus, we propose an adaptive prediction method of $8^{th}$ order predictor and it works as follows:

Initially, neighboring pixels of all $\mathbf{A}$ are divided into 4 sets, denoted by $set_0$, $set_{45}$, $set_{90}$, $set_{135}$. These sets, with subscript notations self explanatory, consists of three elements, respective to all $A(n)$, as described below.

$$\begin{cases} set_0(n) = \{B(n-1), B(n+1), m_0\}, \\ set_{45}(n) = \{D(n-2), D(n+2), m_{45}\} \\ set_{90}(n) = \{C(n-3), C(n+3), m_{90}\}, \\ set_{135}(n) = \{D(n-4), D(n+4), m_{135}\} \end{cases} \tag{4}$$

Here, $m_0$= $(B(n-1) + B(n+1))/2$, $m_{45}$= $(D(n-2) + D(n+2))/2$, $m_{90}$= $(C(n-3) + C(n+3))/2$ and $m_{135}$= $(D(n-4) + D(n+4))/2$.

In order to estimate strength of edges at $A(n)$, we estimate four parameters that gives intensity variation of pixels in four direction. These parameters are proposed to be estimated as

$$\begin{cases} \sigma^2_{set_0} = (1/3) \times \sum_{n=1}^{3}(set_0(n) - m)^2, \\ \sigma^2_{set_{45}} = (1/3) \times \sum_{n=1}^{3}(set_{45}(n) - m)^2 \\ \sigma^2_{set_{90}} = (1/3) \times \sum_{n=1}^{3}(set_{90}(n) - m)^2, \\ \sigma^2_{set_{135}} = (1/3) \times \sum_{n=1}^{3}(set_{135}(n) - m)^2 \end{cases} \tag{5}$$

where, $m = (m_0+m_{45}+m_{90}+m_{135})/4$. Here, $\sigma^2_{set_0}$, $\sigma^2_{set_{90}}$, $\sigma^2_{set_{45}}$ and $\sigma^2_{set_{135}}$ are the estimate of edges of $A(n)$ at horizontal (at $0^0$), vertical (at $90^0$), in $45^0$ and in $135^0$ directions respectively. Then predicted value of $A(n)$ is considered to be the weighted sum of $m_0$, $m_{45}$, $m_{90}$ and $m_{135}$,

$$\hat{A}(n) = w_0 m_0 + w_{45} m_{45} + w_{90} m_{90} + w_{135} m_{135}. \tag{6}$$

where $w_0$, $w_{45}$, $w_{90}$ and $w_{135}$ are the prediction coefficients used with $m_0$, $m_{45}$, $m_{90}$ and $m_{135}$ respectively.

The parameter with smallest value in (5) indicate that the intensity value variation of pixels in that direction is minimum. For example, suppose

$$\sigma^2_{set_{135}} < \sigma^2_{set_{45}} < \sigma^2_{set_{90}} < \sigma^2_{set_0}, \tag{7}$$

It implies that intensity variation of pixels along $135^0$ direction is minimum. Then intuitively, for better prediction of $A(n)$, the element of $set_{135}$ should be given more weight as compared to element of other sets. Thus in (7), prediction coefficients should follow

$$w_0 < w_{90} < w_{45} < w_{135} \tag{8}$$

With the arguments given above in (7) and (8), we propose to find the adaptive prediction coefficients as follows:

We take reciprocal of various estimated parameters in (5) and normalized it to find prediction coefficients as follows:

$$\begin{cases} w_0 = (\sigma^2_{set_0})^{-1}/N, w_{45} = (\sigma^2_{set_{45}})^{-1}/N) \\ w_{90} = (\sigma^2_{set_{90}})^{-1}/N), w_{135} = (\sigma^2_{set_{135}})^{-1}/N) \end{cases} \tag{9}$$

where $N$ is the normalization factor and is given by $N = (\sigma^2_{set_0})^{-1} + (\sigma^2_{set_{45}})^{-1} + (\sigma^2_{set_{90}})^{-1} + (\sigma^2_{set_{135}})^{-1})$.

Here, neighboring pixels are exploited to find the similarity of pixels in different directions and thus prediction coefficient are estimated adaptively. In case, if any of the parameter is zero, that means there is no intensity variation of pixels in that direction, then corresponding prediction coefficient is made 1 and the rest are made 0. If more than one parameter is zero, the weights are equally divided among the corresponding coefficients.

Thus combining (6) and (9), $A(n)$ can be predicted efficiently. Hence, the proposed EAP method is adaptive to edge characteristics of neighboring pixels. After predicting each pixel of $\mathbf{A}$ and finding corresponding $e(n)$, we can apply additive PEE method (discussed in section 2) to get watermarked pixels of $\mathbf{A}$ and is denoted by $\mathbf{A_w}$.

### 3.2. Prediction and watermarking of rest of the pixels

In lossless coding, decoder do not have the information of non-causal pixels whereas in reversible image watermarking, decoder has the information of non-causal watermarked pixels and, being less distorted, they can be used for prediction.

In the second phase, pixels denoted by $\mathbf{B}$, shown in Fig. 1(a), are predicted and then undergoes watermarking process by PEE method. For pixels of type $\mathbf{B}$ (shown in Fig 2(a)), FCP defined the non-adaptive predictor to be,

$$\hat{B}(n) = (D(n-3) + D(n+3) + A_w(n-1) + A_w(n+1))/4. \tag{10}$$

Thus, it uses 2 watermarked pixels obtained from first phase and 2 original pixels for prediction. Thus, it again uses an ineffective prediction method. But, we follow the same process for prediction as explained in first phase. However,

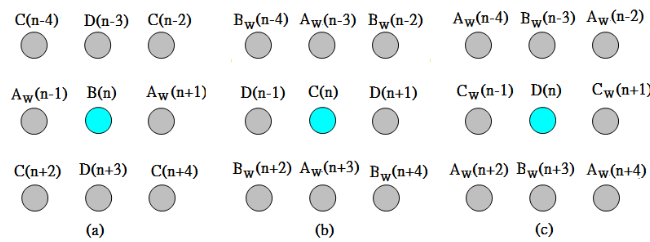


**Fig. 2**. (a) Second Phase, (b) Third Phase (c) Fourth Phase

pixels available for proposed method are: 6 original and 2 watermarked pixels shown in Fig. 2(a). Once, we get the predicted value of type $\mathbf{B}$, additive PEE method is used to get watermarked pixels denoted by $\mathbf{B_w}$.

In the third phase, prediction of $\mathbf{C}$ typed pixels is exactly the same as it is done for previous two phases. The only difference is that pixels available for prediction are: 2 original and 6 watermarked pixels which is shown in Fig. 2(b). In last phase, a similar procedure can be applied for pixels of type $\mathbf{D}$. But, pixels available for proposed EAP method consist of 8 watermarked pixel and is shown in Fig. 2(c).

Thus, proposed EAP method exploits neighborhood pixels efficiently and estimates the prediction coefficients adaptively based on the similarity of pixels in different directions.

### 3.3. Watermark Embedding and Extracting Process

Embedding process is done by traditional PEE method which may results into overflow/underflow problem. To overcome such problem, we adopt method in [13], and is briefly discussed as follows:

The input image, which is typically of 8 bit resolution, is modified in such a way that the pixel values of the modified image will be in the range [$Q$,255 - $Q$], where $Q$ is the order of embedding. The modification is given in (11) and the truncation information is send as an overhead to decoder. A detailed discussion on this issue is given in [13].

$$I_m(i,j) = \begin{cases} Q & \text{if } I(i,j) \leq Q \\ I(i,j) & \text{if } Q < I(i,j) < 255 - Q \\ 255 - Q & \text{otherwise} \end{cases} \tag{11}$$

Once we get the preprocessed image ($\mathbf{I_m}$), classify its pixels into four groups shown in Fig.1(a) based on their coordinate positions. Then the proposed algorithm, as discussed in previous sub-sections, can be used to get watermarked image ($\mathbf{I_w}$).

At the decoder side, we have watermarked image ($\mathbf{I_w}$). In a reverse order of watermark embedding, we can recover the preprocessed image ($\mathbf{I_m}$) and watermark data easily. Recovery process at decoder is as follows:

- Apply proposed prediction algorithm for pixels of type $\mathbf{D}$ using 8 neighborhood watermarked pixels shown in Fig. 2(c) as it is done at encoder. Then apply inverse PEE using (2) to get secret bits and original pixels.

**Table 1**. Phase wise performance comparison of Proposed Algorithm (EAP) with FCP [11]- Capacity (number of bits)

| Phase | Method | Pepper | Couple | Plane | Sailboat |
|---|---|---|---|---|---|
| Phase 1 | FCP | 10,675 | 11,079 | 19,658 | 9,740 |
| | EAP | 13,142 | 12,069 | 22,813 | 10,941 |
| Phase 2 | FCP | 9,978 | 10,888 | 19,658 | 9,474 |
| | EAP | 10,632 | 11,663 | 22,756 | 10,421 |
| Phase 3 | FCP | 9,871 | 10,716 | 19,577 | 9,822 |
| | EAP | 10,473 | 11,925 | 22,414 | 10,600 |
| Phase 4 | FCP | 9,353 | 10,794 | 19,453 | 9,440 |
| | EAP | 9,822 | 11,344 | 22,045 | 10,428 |
| Final | FCP | 39,877 | 43,485 | 78,154 | 38,316 |
| | EAP | **44,069** | **47,501** | **91,890** | **42,241** |

- After extracting original pixels of type $\mathbf{D}$, apply proposed prediction algorithm for pixels of type $\mathbf{C}$ using 2 original pixels (obtained from previous phase) and 6 watermarked pixels shown in Fig. 2(b). Thus, recover the original pixels of type $\mathbf{C}$ and hidden bits.

- Repeat the same process for pixels of type $\mathbf{B}$ and extract the original pixel and hidden bits. Following same process, pixels of type $\mathbf{A}$ can also be recovered.

Thus, original preprocessed image ($\mathbf{I_m}$) and full watermark information can be recovered completely. To recover original image ($\mathbf{I}$) from preprocessed image ($\mathbf{I_m}$), under/over flow recovery process can be done using method given in [13].

## 4. SIMULATION RESULT

We implemented the proposed reversible watermarking algorithm and compared its performance with existing methods. For this purpose we used several standard test images of 8 bit resolution and of $512 \times 512$ dimension. Results with single layer embedding ($Q = 1$) is given in Table 1 and Table 2 while results with higher value of $Q$ are given in Fig. 3.

In Table 1, the embedding capacities of four phases have been given and it can be found that the proposed algorithm has more embedding capacity in every phase as compared to FCP [11]. Also, final capacity is shown which is less than sum of the capacities of the four phases due to overhead factor required for sending truncation information to overcome overflow/underflow problem given in (11). For better evaluation, we compared our results with other reversible algorithms and is shown in Table 2. It can be concluded that proposed EAP algorithm outperforms existing algorithms for $Q = 1$.

To show effectiveness of our method for multi-layer embedding, we compared our results with existing algorithms. From Fig.3, it can be observed that proposed EAP algorithm works better than existing algorithms. However, as compared with FCP [11], improvement is large for image with high texture/edge (Sailboat in Fig. 3(b)) as compared to smooth images such as Plane. It is obvious as proposed algorithm is

**Table 2**. Performance comparison of Proposed Algorithm (EAP) with various other methods - Capacity (number of bits)

| Schemes | Lena | Baboon | Sailboat | Plane |
|---|---|---|---|---|
| Ni [4] | 5,460 | 5,421 | 7,301 | 16,171 |
| Lin [5] | 59,900 | 19,130 | 37,644 | 80,006 |
| Hu [10] | 60,241 | 21,411 | 28,259 | 77,254 |
| EAP | **66,512** | **22,685** | **42,241** | **91,890** |

adaptive to edges and thus works efficiently as compared to FCP in images having texture/edges.

The propose algorithm requires more computational power as compared to FCP [11]. It is because, propose algorithm estimates four parameters in (5) which involves multiplication as well as addition operations, but still computational cost is small.

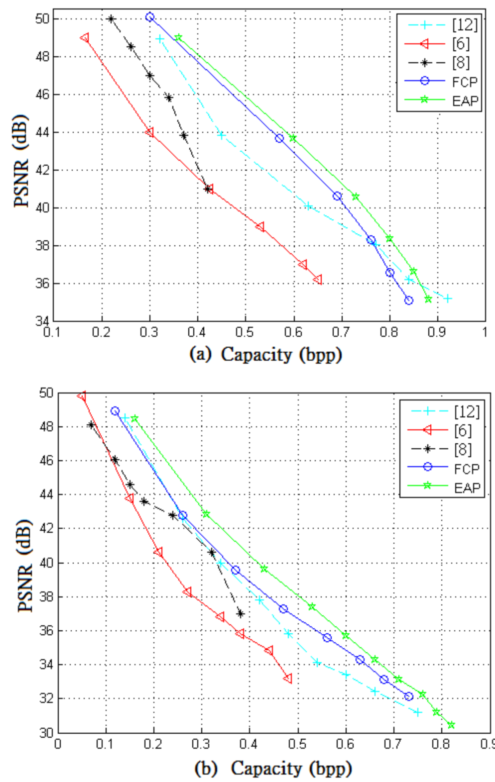


**Fig. 3**. Multilayer embedding (a) Plane (b) Sailboat

## 5. CONCLUSION

In this paper, we proposed a novel reversible image watermarking algorithm based on additive PEE. This work is based on an observation that full context prediction provides better prediction accuracy as compared to non-symmetrical prediction and thus achieves better embedding capacity. In view of this, we proposed an efficient adaptive prediction (EAP) method that exploits characteristics of neighboring pixels much effectively than other prediction methods reported in literature. Experimental results demonstrate that proposed algorithm outperforms the existing methods.

## 6. REFERENCES


[1] R. Li, O. C. Au, C. K. M. Yuk, S. Yip, and T. Chan, "Enhanced image trans-coding using reversible data hiding," in IEEE Int. Symp. Circuits and Systems ISCAS 2007, pp. 1273-1276.

[2] J.B. Feng, I.C. Lin, C.S. Tsai, Y.P. Chu, "Reversible watermarking: current status and key issues," in International Journal of Network Security, Vol.2, No.3, PP.161-11, May 2006

[3] M.U. Celik, G. Sharma, A.M. Tekalp, and E. Saber, "Lossless generalized lsb data embedding," in IEEE Trans. on Image Processing, vol. 14, no. 2, pp. 253266, 2005.

[4] Z. Ni, Y. Q. Shi, N. Ansari, and S.Wei, 'Reversible data hiding," in IEEE Trans. Circuits Syst. Video Technol." vol. 16, no. 3, pp. 354-362, 2006.

[5] C. C. Lin and N. L. Hsueh "lossless data hiding scheme based on three-pixel block differences," in Pattern Recogni., vol. 41, no. 4, pp.1415-1425, Apr. 2008.

[6] K.-S. Kim, M.-J. Leea, H.-Y. Leeb, H.-K. Leea "Reversible data hiding exploiting spatial correlation between sub-sampled images," in Pattern Recognit., 2009, DOI: 10.1016/j.patcog.2009.04.004.

[7] J. Tian, "Reversible data embedding using a difference expansion," in IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 8, pp. 890-896, Aug. 2003.

[8] H.-J. Kim, V. Sachnev, Y. Q. Shi, J. Nam, and H.-G. Choo, "A novel difference expansion transform for reversible data embedding," in IEEE Trans. Inf. Forensic Security, vol. 3, no. 3, pp. 456-465, Sep. 2008.

[9] D. M. Thodi and J. J. Rodriguez, "Expansion embedding techniques for reversible watermarking," in IEEE Trans. Image Processing, vol. 16, no. 3, pp. 721-730, 2007.

[10] Y. Hu, H.-K. Lee, and J. Li, "DE-based reversible data hiding with improved overflow location map," in IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 2, pp. 250-260, Feb. 2009.

[11] M. Chen, Z. Chen, X. Zeng, and Z. Xiong " Reversible image watermarking based on full context prediction," in International conference on Image Processing (ICIP), page 4253-4256. IEEE, (2009).

[12] L. Luo, Z. Chen, M. Chenm, X. Zeng, X. Zhang, "Reversible Image watermarking using interpolation technique," in IEEE Trans. Inf. Forensic Security, vol. 5, no. 1,pp. 187-193,march 2010.

[13] V. Conotter, G. Boato, M. Carli, K. Egiazarian, "High capacity reversible data hiding based on histogram shifting and non-local means," in Local and Non-Local Approximation in Image Processing, 2009.