www.ijecs.in
International Journal Of Engineering And Computer Science ISSN: 2319-7242
Volume 4 Issue 10 Oct 2015, Page No. 14817-14823

A Novel Reversible De-Identification Approach For Lossless Image Compression Based On Reversible Watermarking Mechanism Based On Obfuscation Process

***Ghatamaneni Tejaswini*[1] *T.Venkataramana*[2]**

Department of ECE (DECS) [1], Assistant professorM.tech[2]

Siddhartha educational academy group of institutions, Tirupathi, Andhra Pradesh, INDIA

Abstract

Although tremendous progress has been made in the past years on watermarking for protecting information from incidental or accidental hacking, there still exists a number of problems. De-Identification is a process which can be used to ensure privacy by concealing the identity of individuals captured by video surveillance systems. One important challenge is to make the obfuscation process reversible so that the original image/video can be recovered by persons in possession of the right security credentials. This work presents a novel Reversible De-Identification method that can be used in conjunction with any obfuscation process. The residual information needed to reverse the obfuscation process is compressed, authenticated, encrypted and embedded within the obfuscated image using a two-level Reversible Watermarking scheme. The proposed method ensures an overall single-pass embedding capacity of 1.25 bpp, where 99.8% of the images considered required less than 0.8 bpp while none of them required more than 1.1 bpp. Experimental results further demonstrate that the proposed method managed to recover and authenticate all images considered.

KEYWORDS: De-Identification, embedding capacity, obfuscated image

1. INTRODUCTION

The goals of the reversible watermarking are to protect the copyrights and recover the original image. The robustness, imperceptibility, higher embedding capacity, effectiveness, payload capacity, visual quality and the security are the basic criterion of the reversible watermarking. The reversible watermarking is especially suitable for the applications that require high quality images such as medical and military images. Reversible watermarking is also useful in remote sensing, multimedia archive management, law enforcement etc. It is a novel category of watermarking schemes. Reversible watermarking schemes have to be robust against the intentional or the unintentional attacks, and should be imperceptible to avoid the attraction of attacks and value lost. The robustness of the watermarked images against attacks has been verified on the parameters of PSNR (Peak Signal to Noise Ratio) and MSE (Mean Square Error) which show that the resulting quality of combination watermarking method is good than other techniques. Navnath Narawade et.al [3] describes a complete review of reversible watermarking techniques based upon the embedding capacity, PSNR and processing time. Nikhil Dalshania et.al [4] describes a comparative study of reversible watermarking techniques based on different parameters. Hence in previous works, there are various techniques are available for reversible watermarking. The challenge was to find which one is the best method robust to noise. In traditional reversible watermarking techniques, our main concern is to embed and recover the watermark and also restore the original image with minimum distortion. Our attempt here is to study three basic robust techniques and compare them on the basis of PSNR and processing time. The techniques we have studied here are difference expansion, reversible contrast mapping and least significant bit technique.

The concept of reversible watermark firstly appeared in the patent owned by Eastman Kodak [1]. Honsinger et al. [1] utilised a robust spatial additive watermark combined with modulo additions to achieve reversible data embedding. Goljan et al. [2] proposed a two cycles flipping permutation to assign a watermarking bit in each pixel group. Celik et al. [3] presented a high capacity, reversible dataembedding algorithm with low distortion by compressing quantization residues. Tian [4] presented a reversible data embedding approach based on expanding the pixel value difference between neighboring pixels, which will not overflow or underflow after expansion. Thodi and Rodrguez exploited the inherent correlation among the neighbouring pixels in an image region using a predictor. Xuan et al. [5] embedded data into high-frequency coefficients of integer wavelet transforms with the companding technique, and utilised histogram modification as a preprocessing step to prevent overflow or underflow caused by the modification of wavelet coefficients.

The earliest reference to reversible data embedding we could find is the Barton patent [8], filed in 1994. In his invention, the bits to be over layed will be compressed and added to the bit string, which will be embedded into the data block. Honsinger et al. [9], reconstruct the payload from an embedded image, then subtract the payload from the embedded image to losslessly recover the original image. Macq [10] proposes an extension to the patchwork algorithm to achieve reversible data embedding. Fridrich et al. [1], develop a high capacity reversible data-embedding technique based on embedding message on bits in the status of group of pixels. They also describe two reversible data-embedding techniques for lossy image format JPEG. De Vleeschouwer et al. [11], propose a reversible data-embedding algorithm by circular interpretation of Objective transformations. Kalker et al. [12], provide some theoretical capacity limits of lossless data compression based reversible data embedding [6] and give a practical code construction. Celik et al. [13], [14], present a high capacity, low distortion reversible data embedding algorithm by compressing quantization residues. They employ the lossless image compression algorithm CALIC, with quantized values as side-information, to efficiently compress quantization residues to obtain high embedding capacity.

Reversible watermarking has found a huge surge of experimentation in its domain in past decade as the need of recovering the original work image after extracting the watermark arises in various applications such as the law enforcement, medical and military image system, it is crucial to restore the original image without any distortions [7]. In traditional watermarking techniques, our main concern is to embed and recover the watermark with minimum loss. The quality of original work image we get after extraction is highly degraded and not restorable. But in applications like law enforcement, medical and military, in which superior quality of image is needed, we cannot use these algorithms. In medical images, some prerequisite information about the patient is watermarked in it while transmitting and at reception we need to have both, the original image and that information to be recovered lossless. This type of result is achievable by making use of any reversible watermarking algorithm out of a pool of algorithms [10].

In our opinion, the interest in reversible watermarking is appropriate. However, we expect that military applications will be overshadowed by applications such as medical application and law enforcement application. These latter applications may turn out to be the most compelling. Considerable progress has been made toward enabling these applications. Further progress is needed in methods for handling geometric and temporal distortions. We expect other exciting developments to arise from research in reversible watermarking. A reversible watermark will drive the next generation era.

2. SYSTEM OVERVIEW

Fig. 1 illustrates the schematic diagram of the Forward Reversible De-Identification process which receives the original image ***I*** and conceals the face of the person using the *Face Obfuscation* process to generate an obfuscated image ***IΘ***. This work considers color images using the *YCbCr* color space. The coordinates of the top left corner and bottom right corner of the De-identified region is enclosed within the bounding box ***β***, which is passed to both *ROI Extraction* processes to extract the face image ***F*** and the obfuscated face image ***FΘ***. The face images are then subtracted to derive the difference face image ***D***.

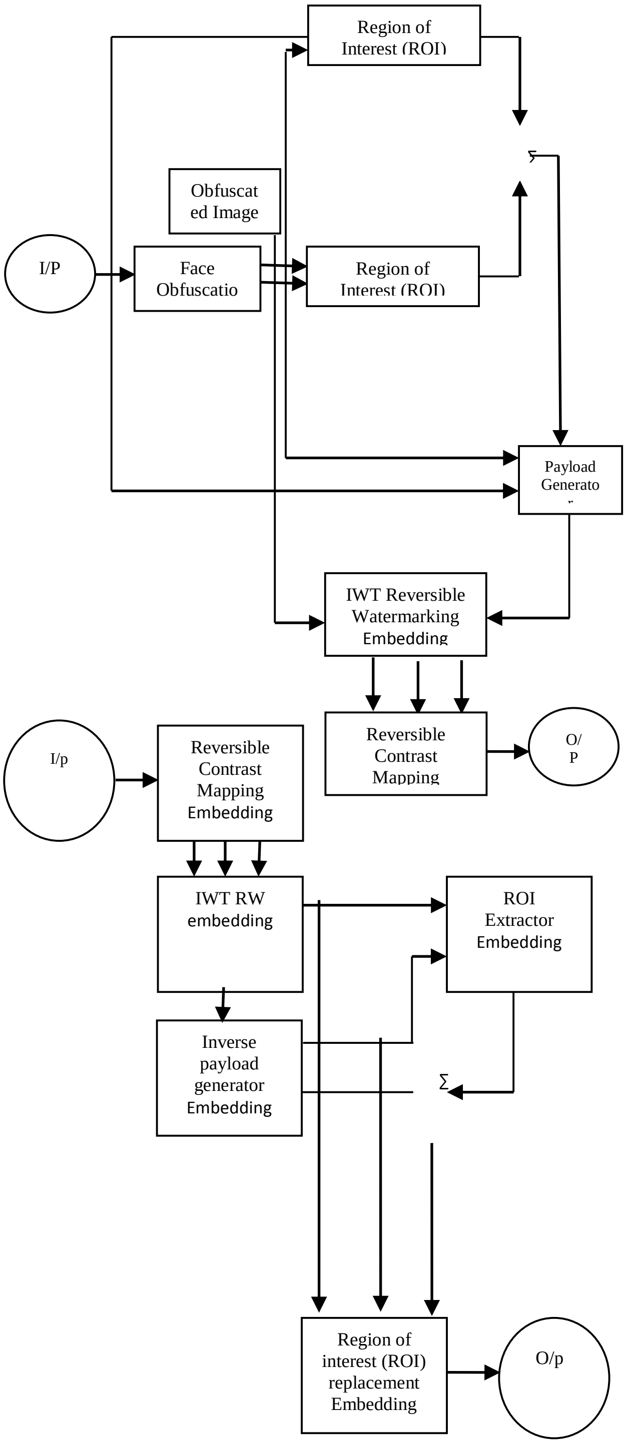
Region of Interest (ROI)
Obfuscated Image
I/P
Face Obfuscatio
Region of Interest (ROI)
Payload Generator
IWT Reversible Watermarking Embedding
Reversible Contrast Mapping
O/P
I/p
Reversible Contrast Mapping Embedding
IWT RW embedding
ROI Extractor Embedding
Inverse payload generator Embedding
Σ
Region of interest (ROI) replacement Embedding
O/p

Fig.2 Block diagram of Inverse Reversible De Identification Process

Figure 1: Block diagram of Forward Reversible De Identification Process

Fig. 2 depicts the schematic diagram of the Inverse Reversible De-Identification process. The second order embedded obfuscated image $\hat{I}_{\theta}^{W}$ is inputted to the Reversible Contrast Mapping Extraction process which extracts the first level embedded obfuscated image I_{θ}^{W} together with the auxiliary information **A** and the residual bit stream **e**. The IWT Reversible Watermark Extraction process is then used to extract the original payload $\mathbf{p}_{\mathbf{a}}^{\mathbf{e}}$ and original obfuscated image $\mathbf{I}_{\boldsymbol{\theta}}$. The Inverse Payload Generator reverses the process of the Payload Generator and recovers the difference image **D** and the bounding box **β**, which is used by the ROI Extractor process to extract the obfuscated face $\mathbf{F}_{\boldsymbol{\theta}}$.

3. FORWARD REVERSIBLE DE-IDENTIFICATION

A. Face Obfuscation

The Face Obfuscation process receives the original image ***I*** and detects the face region and eye locations using the ground truth information available in the color FERET dataset. This can be automated using the face detector in [12] and the eye detector in [13] which ensure high accuracies. However, the main contribution of this work is to present a Reversible De-Identification method which is independent from the obfuscation process.

B. ROI Extraction

The ROI Extraction process is a simple algorithm which employs the bounding box coordinates ***β*** to identify the region to be cropped from the input image ***I*** (or $\mathbf{I}_{\boldsymbol{\theta}}$). The cropped sub-image is then stored in the face image ***F*** (or $\mathbf{F}_{\boldsymbol{\theta}}$ obfuscated face image).

C. Payload Generator

The Payload Generator Process receives the difference image ***D*** which is compressed using

the predictive coding method presented in [15] followed by the Deflate algorithm.

β	Payload	Hash

Figure 3: The authenticated packet *pa*

D. IWT Reversible Watermarking Embedding

The IWT Reversible Watermarking Embedding process first derives the number of decompositions $\mathbf{N_{dec}}$ needed to embed the packet $\mathbf{p_a^e}$ within $\mathbf{I_\theta}$ using

$$\mathbf{N_{dec}} = \begin{cases} \left\lceil -\frac{1}{2}\log_2(1-\mathrm{C}) \right\rceil & \text{if } \mathrm{C} < 1 \\ \mathrm{M} & \text{otherwise} \end{cases} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

where **M** indicates the maximum number of decomposition allowed and **C** represents the capacity needed to embed $\mathbf{p_a^e}$ bits and is computed using

$$\boldsymbol{C} = \frac{|\mathbf{p_a^e}|}{\boldsymbol{Ch \times W \times H}} \quad \ldots\ldots\ldots\ldots\ldots\mathbf{(2)}$$

where l l represents the cardinality of the set, **W** and **H** represent the number of columns and rows in the image and **Ch** represents the number of color channels (in our case 3). This process then adopts the CDF(2,2) integer wavelet transform specified in [17] to decompose the image.

1) Threshold Selection

The proposed Threshold Selection method is based on the observation that different sub-bands provide different levels of distortions [18]. However, in order to reduce the complexity of the optimization function, the following assumptions were made

- The chrominance sub-bands have similar properties and thus share the same threshold.
- The **HL** and **LH** sub-bands within the same color channel (luminance or chrominance) are assumed to have similar characteristics and therefore have the same threshold.

The number of thresholds to be considered by the Threshold Selection process is given by

$$\boldsymbol{N_T} = \mathbf{2(1 + N_{dec})} \ldots\ldots\mathbf{(3)}$$

The population of thresholds evolves over a number of generations using mutation and cross-over. The mutation process generates a mutant vector for every threshold vector contained within the population of **NP** vectors using

$$\mathbf{V_i} = \mathbf{\Delta_{r1}} + \boldsymbol{\alpha} \times (\mathbf{\Delta_{r2}} - \mathbf{\Delta_{r3}}) \ldots\ldots\ldots\mathbf{(4)}$$

where {r1,r2,r3} ∈[0,NP-1] are mutually different integers which are different from index *i*, while α is a mutation factor that controls the magnitude of the differential variation. The cross-over process is then used to increase the diversity of the mutated vectors, and are included in another list of potential thresholds $\bar{\Delta}_\mathbf{i}$ according to

$$\bar{\Delta}_\mathbf{i} = \begin{cases} \boldsymbol{V_{i,j}} & \boldsymbol{if\ \vartheta \le \Gamma} \\ \boldsymbol{\Delta_{i,j}} & \boldsymbol{otherwise} \end{cases} \ldots\ldots\ldots(5)$$

where $\boldsymbol{\vartheta}$ is a uniformly distributed random number [0,1] and $\boldsymbol{\Gamma}$ represents the cross-over probability which has a constant value. The potential thresholds vectors $\bar{\Delta}_\mathbf{i}$which do not provide enough capacity are pruned from the list.

2) Forward Integer Wavelet Expansion

The Forward Integer Wavelet Expansion process receives the set of thresholds **T** which are derived by the Threshold Selection process and encapsulates the packet $\mathbf{p_a^e}$ shown in Fig. 3 to generate the packet **s** to be Embedded in fig 4

N_d	T	L	$\boldsymbol{p_a^e}$

Figure 4: The actual bit stream to be embedded *s*

A wavelet coefficient $\mathbf{w_{\mu,j}}$ is considered for embedding if its magnitude is smaller than the threshold responsible of sub-band μ. A bit *b* is embedded using

$$\mathbf{\hat{w}_{\mu,j}} = \mathbf{2w_{\mu,j}} + \boldsymbol{b} \ldots\ldots\ldots\ldots\mathbf{(6)}$$

The other wavelet coefficients are not considered for embedding. However, in order to prevent ambiguities at the receiver, they are expanded using

$$\mathbf{\hat{w}_{j,\mu}} = \begin{cases} \mathbf{w_{j,\mu}} + \boldsymbol{T_\mu} & \boldsymbol{if}\ \mathbf{w_{j,\mu} \ge 0} \\ \mathbf{w_{j,\mu}} - \boldsymbol{T_\mu} + \mathbf{1} & \boldsymbol{otherwise} \end{cases} \quad \ldots\ldots\mathbf{(7)}$$

where **Tμ** is the threshold responsible for sub-band **μ**. This process is terminated once all the bits in ***s*** are embedded.

DOI: 10.18535/ijecs/v4i10.33

In the rare event where this process fails to embed all bits, these bits are stored in a bit-sequence ***e***.

E. Reversible Contrast Mapping

The only problem with the proposed Forward Integer Wavelet Expansion process is that sometimes **A** and **e** are not The main advantage of using RCM is that it embeds all information within the image without any ambiguities and provides an additional capacity of 0.5 bpp. However, the main limitation of the RCM is its limited capacity and that the distortion can become significant when embedding large payloads. However, the packet size ***r*** is expected to be very low (generally 2-bits).

The forward RCM transforms two neighboring pixel pairs $(\boldsymbol{x}, \boldsymbol{y})$ into $(\boldsymbol{x'}, \boldsymbol{y'})$ using

$$\mathbf{x'} = \mathbf{2x} - \mathbf{y}, \mathbf{y'} = \mathbf{2y} - \mathbf{x} \quad \textbf{.......(8)}$$

To prevent overflow and underflow, the transform is restricted to a sub-domain defined by the pixels which satisfy the conditions

$$\mathbf{0 \leq 2x - y \leq 255, 0 \leq 2y - x \leq 255} \ \textbf{......(9)}$$

The RCM scheme replaces the least significant bits (LSBs) of the transformed pairs $(\boldsymbol{x'}, \boldsymbol{y'})$**.**
The LSB of **x'** is used to indicate whether information is embedded within **y'** or not.

4. INVERSE REVERSIBLE DE-IDENTIFICATION

A. Reversible Contrast Mapping Extraction

The Reversible Contrast Mapping Extraction process receives the image $\hat{I}_\theta^W$ and recovers I_θ^W and ***r***. The information bit can be extracted from the LSB of x' when the LSB of $\acute{y}$ is '1'. However, in the event when the LSB of x' is '0', both LSBs of x' and y' are forced to be odd and condition (9) is checked. If the condition is satisfied it then represents an odd pixel pair while if it does not it indicates that $y = \acute{y}$ and the original LSB value of *x* is extracted from the bitstream. More information about this is available in [21]. The auxiliary information ***A*** and residual bitstream ***e*** are then extracted

empty. This work adopts the syntax shown in Fig. 5 to represent this information **r** to be embedded. The Flag is a 2-bit field which indicates whether **A** and **e** are empty or not.

from the packet ***r***. The original pixel values are recovered using

$$\boldsymbol{x} = \left[\frac{2}{3}\,\mathbf{x}' + \frac{1}{3}\mathbf{y}'\right], \boldsymbol{y} = [\frac{1}{3}\mathbf{x}' + \frac{2}{3}\mathbf{y}'] \ \textbf{.....}(10)$$

B. IWT Reversible Watermarking Extraction

The IWT Reversible Watermarking Extraction reverses the IWT Reversible Watermarking Embedding process and extracts the payload information $\mathbf{p_a^e}$ and the original obfuscated image I_θ. It must be noted here that initially, the decoder has no knowledge about the number of decompositions employed $\mathbf{N_{dec}}$and the threshold values **T**. The decoder thus assumes that a single decomposition is employed and that the threshold values are set to zero. These values are then updated once they are extracted from the header information of ***s***. It is important that the
encoder does the same thing during embedding in order to ensure synchronization between the encoder and decoder.

The watermarked obfuscated image I_θ^W is then decomposed into N_{dec} levels using the CDF(2,2) integer wavelet transform and the wavelet coefficients of the high frequency sub-bands are scanned using the random permutation index generated using the encryption key. The bits of the packet are extracted from coefficients which satisfy the condition $-2T_{\mu,j} + 1 < \widehat{w}_{\mu,j} < 2T_{\mu,j}$ and are extracted from the LSB of $\widehat{w}_{\mu,j}$ while the original wavelet coefficient is recovered using

$$\mathbf{w_{\mu,j}} = \left[\frac{\mathbf{\widehat{w}_{\mu,j}}}{\mathbf{2}}\right] \ \textbf{..............(11)}$$

The remaining coefficients are not used for embedding and are recovered using

$$\mathbf{w_{\mu,j}} = \begin{cases} \widehat{w}_{\mu,j} - T_\mu & if \ \mathbf{w_{\mu,j}} \geq 0 \\ \overrightarrow{\mathbf{w}}_{\mathbf{\mu,j}} + T_\mu - 1 & otherwise \end{cases} \textbf{.......(12)}$$

This method terminates once all the *L* bytes are extracted. The resulting image is then inverse CDF(2,2) transformed to recover the original obfuscated image I_θ . The auxiliary information (if any) is then used to recover ambiguous pixel values while the residual bit stream ***e*** (if any) is appended to the recovered payload.

C. ROI Replacement

The ROI Replacement process replaces the region marked by the bounding box $\boldsymbol{\beta}$ with the recovered face image ***F***. The image $\mathbf{I_{rec}}$ can be authenticated by comparing the hash derived by computing the *SHA-1* on $\mathbf{I_{rec}}$ to the *Hash* value present in the tail of the packet p_a.

5. RESULTS

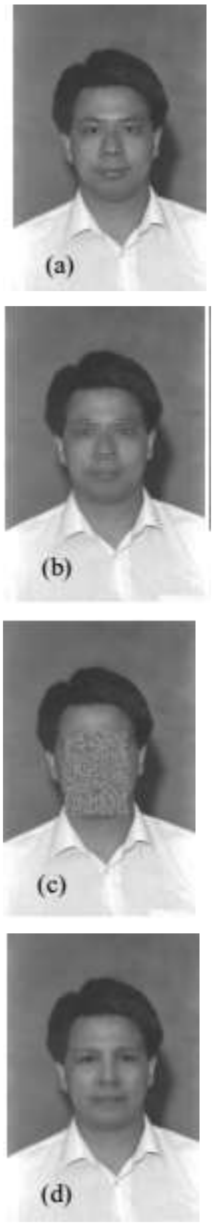

Figure 6: Comparing the resulting reversible de-identified images (a) Original Image, (b) Scrambling of DCT coefficients, (c) Encryption of pixel values and (d) proposed method

6. CONCLUSION

This work presents a novel Reversible De-Identification method for lossless compressed images. The proposed scheme is generic and can be employed with other obfuscation strategies other than k-Same. A two level Reversible-Watermarking scheme was adopted which uses Differential Evolution to find the optimal set of thresholds and provides a single-pass embedding capacity close to 1.25 bpp. Simulation results have shown that this method is able to recover the original image if the correct encryption key is employed. It further shows that 0.8 bpp were sufficient to cater for 99.8% of the frontal images considered and none of the image needed more than 1.1 bpp. Future work will focus on the extension of this algorithm for lossy image and video compression standards.

REFERENCES

[1] A. Senior, S. Pankanti, A. Hampapur, L. Brown, Y. Li Tian and A. Ekin, "Blinkering Surveillance: Enabling video privacy through Computer Vision," *IBM Research Report*, vol. 22886, 2003.

[2] E.M. Newton, L. Sweeney and B. Malin, "Preserving privacy by de-identifying face images," *IEEE Trans. on Knowl. and Data Eng.*, vol. 17, no. 2, pp. 232-243, Feb. 2005.

[3] W. Zhang, S.S. Cheung and M. Chen, "Hiding privacy information in video surveillance systems," in *IEEE Int. Conf. on Image Processing*, Genoa, Italy, Sep. 2005.

[4] I. Martinez-Ponte, X. Desumont, J. Meessen and J.F. Delaigle, "Robust Human Face Hiding ensuring Privacy," in *Proc. of Int. Workshop on Image Analysis for Multimedia Services*, Montreux, Switzerland, Apr. 2005.

[5] T.E. Boult, "Pico: Privacy throough invertible cryptographic obscuration," in *IEEE Proc. of the Computer Vision for Interactive Intelligent Environment*, Whashington DC, USA, Nov. 2005.

[6] K. Martin and K.N. Plataniotis, "Privacy protected surveillance using secure visual object coding," *IEEE Trans.*

Circuits and System for Video Technol., vol. 18, no. 8, pp. 1152-1162, Aug. 2008.

[7] F. Dufaux, M. Ouaret, Y. Abdeljaoued, A. Navarro, F. Bergnenegre and T. Ebrahimi, "Privacy Enabling Technology for Video Surveillance," in *SPIE Mobile Multimedia/Image Processing for Military and Security Applications*, Orlando, Florida, May 2006.

[8] F. Dufaux and T. Ebrahimi Scrambling for privacy protection in video surveillance systems, "Scrambling for privacy protection in video surveillance systems," in *IEEE Trans on Circuits and Systems for Video Technol.*, vol. 18, no. 8, pp. 1168-1178, Aug. 2008.

[9] H. Sohn, W. De Neve and Y-M. Ro, "Privacy protection in video surveillance systems: Analysis of subband-adaptive scrambling in JPEG XR," in *IEEE Trans. Circuits and Systems for Video Technol.*, vol. 21, no. 2, pp. 170-177, Feb. 2011.

[10] J. Meuel, M. Chaumont and W. Puech, "Data hiding in H.264 video for lossless reconstruction of region of interest," in *European Signal Processing Conf.*, Poznan, Poland, Sep. 2007.