

CERIAS Tech Report 2004-85

LOT-Based Adaptive Image Watermarking

by Yuxin Liu and Bin Ni and Xiaojun Feng and Edward J. Delp

Center for Education and Research in
Information Assurance and Security,
Purdue University, West Lafayette, IN 47907-2086

LOT-Based Adaptive Image Watermarking

Yuxin Liu, Bin Ni, Xiaojun Feng, and Edward J. Delp

Video and Image Processing Laboratory
School of Electrical and Computer Engineering
Purdue University, West Lafayette, Indiana

ABSTRACT

A robust, invisible watermarking scheme is proposed for digital images, where the watermark is embedded using the block-based Lapped Orthogonal Transform (LOT). The embedding process follows a spread spectrum watermarking approach. In contrast to the use of transforms such as DCT, our LOT watermarking scheme allows larger watermark embedding energy while maintaining the same level of subjective invisibility. In particular, the use of LOT reduces block artifacts caused by the insertion of the watermark in a block-by-block manner, hence obtaining a better balance between invisibility and robustness. Moreover, we use a human visual system (HVS) model to adaptively adjust the energy of the watermark during embedding. In our HVS model, each block is categorized into one of four classes (*texture*, *fine-texture*, *edge*, and *plain-area*) by using a feature known as the Texture Masking Energy (TME). Blocks with edges are also classified according to the edge direction. The block classification is used to adjust the watermark embedding parameters for each block.

Keywords: digital watermark, Lapped Orthogonal Transform (LOT), human visual system (HVS), perceptual analysis

1. INTRODUCTION

In this paper we describe a robust, invisible watermark embedded in a host image. Such a watermark must be resistant to attacks and at the same time the visual quality of the image must be preserved. In order to improve the robustness of the watermark, one may increase the watermark embedding energy, which may increase the visibility of the watermark. Therefore, our goal is to achieve a balance between robustness and transparency in the embedding.

For invisible watermarking, the watermark must be inserted in a way that the presence of the watermark is not noticeable under typical viewing conditions. Thus modelling the human vision system (HVS) plays a critical role in watermarking [1, 2]. Algorithms for general watermark embedding/detection are described in [3–9].

The development of spread spectrum watermarking is a milestone in invisible robust watermarking [10, 11]. The watermark is embedded using spread spectrum techniques in the most significant components in a host signal, thus guaranteeing its robustness.

It is worth mentioning that with respect to image quality watermarking is closely related to image compression. Human visual system (HVS) research has been actively exploited in lossy image compression to improve the coding efficiency while maintaining a good subjective quality. Theoretically, every aspect of HVS research can be used in watermarking as well particularly in trying to determine how the watermark will manifest itself in the watermarked image.

It has been recognized that watermark embedding in the frequency domain has many advantages in terms of robustness and transparency. Many watermarking methods have been based on this concept. Some of the methods obtain transforms of the entire image [10] while others use block-based transforms such as the DCT [12]. The method we propose in this paper is based on the block-based Lapped Orthogonal Transform (LOT) [13]. The advantage of using LOT as opposed to conventional block-based transform such as DCT is that it

Address all correspondence to E. J. Delp at ace@ecn.purdue.edu, telephone: +1 765 494 1740. This work was partially supported by the Air Force Research Laboratory, Information Directorate, Rome Research Site, under research grant number F30602-02-2-0199.

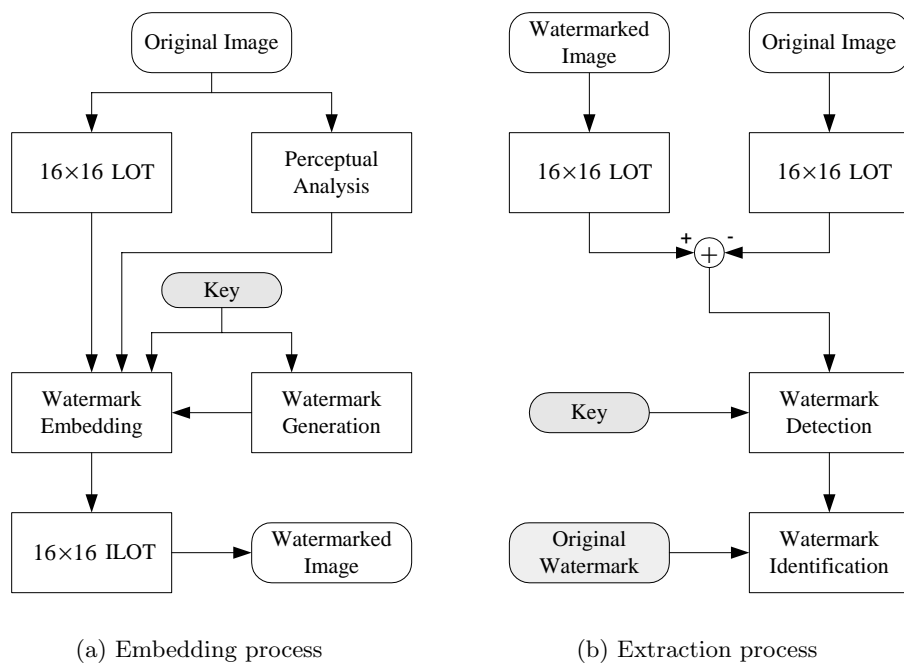


Figure 1. Block Diagram of LOT Watermarking

reduces the block artifacts caused by the watermark embedding. LOT was used in secure watermarking in [14] with the watermark embedded using a template with log-polar-mapping to combat the geometrical attacks. Our new method differs in the way the watermark is embedded using an HVS model. We use two types of characteristics in HVS to embed the watermark in an image-content-adaptive manner, thus improving our watermarking performance. Our technique also does not currently use a synchronization template.

2. GENERAL FRAMEWORK

As shown in Figure 1, our watermarking scheme follows the spread spectrum watermarking approach [10, 15]. In contrast to the use of transforms such as DCT, we use block-based LOT. LOT divides an image into overlapping 16×16 blocks and maps each block into an 8×8 block in the frequency domain, hence obtaining an image in the transform domain with the same size as the original. The use of LOT helps reduce the block artifact caused by block-based embedding. Details of LOT are discussed in Section 3.1. To obtain a balance between robustness and invisibility of the embedded watermark, we introduce a perceptual analysis module that uses HVS properties to adaptively embed the watermark in each block. The perceptual analysis module first extracts a feature from each block, namely the Texture Masking Energy (TME), described below. The blocks are then classified into four categories according to the TME. Blocks classified as *edge* by the use of the TME are further classified into four *edge* classes using an edge detector. The effect of this is that each block is classified into one of seven classes. Details of the perceptual analysis are presented in Section 3.2. The watermark embedding energy in each block is adjusted accordingly to adapt to the sensitivities of the HVS.

3. WATERMARKING USING LOT AND HVS PERCEPTUAL ANALYSIS

3.1. The Lapped Orthogonal Transform

LOT is described in-depth in [13], which presents a DCT-based fast implementation for LOT and its inverse transform (ILOT). In the one-dimensional (1D) case, for a discrete 1D signal x_0 with length MN , LOT divides

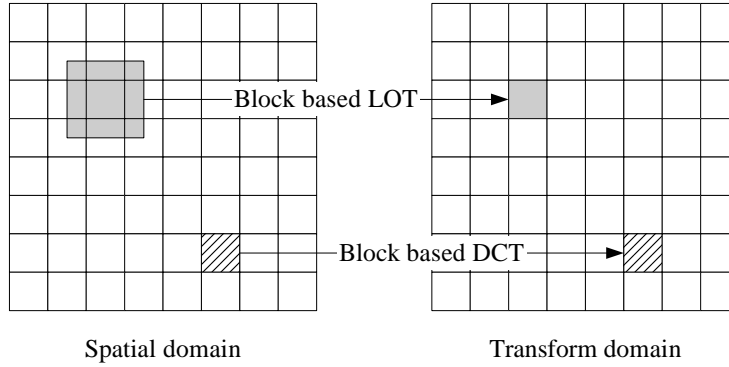


Figure 2. Block-based LOT as opposed to block-based DCT

the signal into M segments where each segment has length L that satisfies $N \leq L \leq 2N$. This means that every two adjacent segments have an overlapped portion of length $(L - N)/2$. LOT transforms each signal fraction into an N -entry coefficient vector, and thus the transformed signal has the same total length as the original signal. Like some other traditional transforms, it is straightforward to derive the two-dimensional (2D) separable LOT from the 1D version. As shown in Figure 2, the 2D LOT maps every $L \times L$ block in the spatial domain to an $N \times N$ block in the frequency domain. We chose $N = 8$ and $L = 2N$ in our implementation. Since every block in the transform domain contains information of the adjacent blocks in the spatial domain, block artifacts caused by any processing in a block-by-block manner in the transform domain will not be as noticeable in the spatial domain. LOT has been used in image and video compression [16].

3.2. Block Categorization Using HVS Perceptual Analysis

Characteristics of the HVS have been studied and exploited in the area of image and video compression [17]. The limits of the HVS in the intensity, spatial, and temple domain allows images to be represented with fewer bits but adequate subjective visual quality. For example, the HVS has different sensitivities to different image details, resulting in that changes in a “flat” region are more perceptible than the same amount of changes in a complex texture region. The HVS perceptual analysis also provides methods in digital watermarking to increase the energy of the embedded watermark while keep it imperceptible [1, 2].

In our perceptual model, two types of HVS properties are used: the sensitivity to the complexity of texture and the sensitivity to the direction of edges. We classify the image blocks into different categories based on these HVS properties. The block-by-block watermark embedding is then implemented accordingly. This is done to maximize the capacity of each host block to keep the watermark invisible.

We first extract an HVS feature, known as the Texture Masking Energy (TME), from each block as follows [18]

$$T_E(k) = \left[\sum_{\substack{i=0 \\ (i,j) \neq (0,0)}}^{N-1} \sum_{j=0}^{N-1} \hat{H}^{-1}(i,j)^2 X_k(i,j)^2 \right]^{1/2}. \quad (1)$$

In the above equation, $X_k(i,j)$ denotes the LOT transform coefficient located in position (i,j) of the k th block. $\hat{H}(f)$ is the HVS relative sensitivity function with respect to the spatial frequency f , where f is related with the spatial position (i,j) as follows

$$f(\text{cycles/degree}) = \frac{\sqrt{i^2 + j^2}}{2N} (\text{cycles/pixel}) \times f_s (\text{pixels/degree}), \quad (2)$$

and we chose $f_s = 32$. $\hat{H}(f)$ can be obtained as [18]

$$\hat{H}(f) = |A(f)|H(f)$$

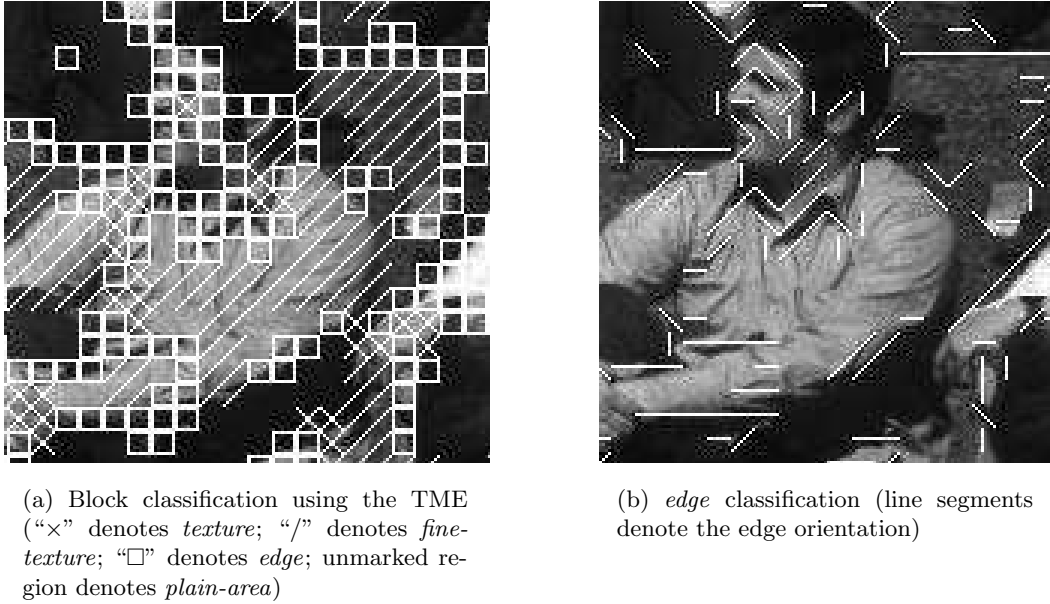


Figure 3. An enlarged block classification using the HVS perceptual analysis of *crowd* (the image given in Figure 4)

$$H(f) = (0.31 + 0.69f) \exp(-0.29f)$$

$$A(f) = \left\{ \frac{1}{4} + \frac{1}{\pi^2} \left[\ln \left(\frac{2\pi f}{\delta} + \sqrt{\frac{4\pi^2 f^2}{\delta^2} + 1} \right) \right]^2 \right\}^{1/2},$$

where $\delta = 11.636 \text{ degree}^{-1}$.

Note that $T_E^2(k)$ is a sum of the energy in the k th block weighted by the reciprocal of the square of the sensitivity function $\hat{H}^2(f)$. This implies that the TME provides a metric measuring the insensitivity, or equivalently, the capability of the block being resistant to noise. We also divide each $N \times N$ block into four $N/2 \times N/2$ sub-blocks, and obtain the TME for each sub-block, namely $T_E(k, l)$, $l = 1, 2, 3, 4$, in a similar way.

Since the TME represents the insensitivities of a block and its sub-portions to noise, we use both $T_E(k)$ and $T_E(k, l)$ to classify the k th $N \times N$ block into four major categories: *texture*, *fine-texture*, *edge*, and *plain-area*. This is done using a similar approach as in [18].

Studies of the HVS have shown that different edge orientations have different masking capabilities. The HVS is less sensitive to changes along the direction of an edge than changes perpendicular to the edge. Thus we further classify each *edge* block into four types according to the edge direction: *vertical*, *horizontal*, *diagonal*, and *anti-diagonal*, using the Sobel operator [16].

An example of the result of our perceptual analysis is shown in Figure 3.

3.3. Watermark Generation and Embedding

In this subsection, we use the spread spectrum approach to generate and embed the image watermark [10]. We use the results of the perceptual analysis described in Section 3.2 to adaptively embed the watermark in a block-by-block manner in the LOT transform domain.

The watermark is formed using an independent, identically Gaussian-distributed pseudo random sequence, with zero-mean and unit-variance. In our implementation, we fixed the number of LOT coefficients that bear

the watermark in each block to be 5. Thus for a 512×512 image, the total length of the embedded watermark is 20480.

We obtain a quantization matrix for the k th block as follows

$$Q_L(k) = Q_S \times M_{block}(k) \times M_{edge}(k), \quad (3)$$

where “ \times ” denotes the element-wise multiplication. Q_S is the standard quantization matrix used in JPEG [19], which reflects the HVS sensitivity to spatial frequencies [12]. Any “noise” added to the transform coefficients shall be barely perceptible, as long as the magnitude of the noise is within the quantization steps designated by Q_S . We design two additional matrices, $M_{block}(k)$ and $M_{edge}(k)$, to adjust the quantization steps in Q_S for the k th block, based on our perceptual analysis,

$$M_{block}(k) = \begin{cases} 0.8 & \textit{plain-area} \\ 1.0 + \min(T_E(k) \times 0.25 / \omega_{edge}, 0.25) & \textit{edge} \\ 1.0 + \min(T_E(k) \times 0.25 / \omega_{fine-texture}, 0.25) & \textit{fine-texture} \\ 1.25 + \min(T_E(k) \times 0.25 / \omega_{texture}, 0.25) & \textit{texture} \end{cases}, \quad (4)$$

$$M_{edge} = \begin{cases} M_1 & \textit{horizontal edge} \\ M_1^T & \textit{vertical edge} \\ M_2 & \textit{diagonal/anti-diagonal edge} \end{cases}. \quad (5)$$

In (4), we chose $\omega_{edge} = 600$, $\omega_{fine-texture} = 300$, and $\omega_{texture} = 400$. In (5), M_1 and M_2 are designed as

$$M_1 = \begin{bmatrix} 1.0 & 1.0 & 1.0 & 1.0 & 1.0 & 1.1 & 1.3 & 1.3 \\ 1.1 & 1.0 & 1.0 & 1.0 & 1.0 & 1.1 & 1.3 & 1.3 \\ 1.2 & 1.2 & 1.0 & 1.0 & 1.1 & 1.1 & 1.3 & 1.4 \\ 1.2 & 1.2 & 1.2 & 1.0 & 1.1 & 1.3 & 1.3 & 1.4 \\ 1.2 & 1.2 & 1.2 & 1.2 & 1.3 & 1.3 & 1.4 & 1.5 \\ 1.2 & 1.2 & 1.2 & 1.3 & 1.3 & 1.4 & 1.5 & 1.5 \\ 1.2 & 1.2 & 1.3 & 1.3 & 1.4 & 1.5 & 1.5 & 1.5 \\ 1.2 & 1.3 & 1.4 & 1.4 & 1.5 & 1.5 & 1.5 & 1.5 \end{bmatrix}, \quad M_2 = \begin{bmatrix} 1.0 & 1.0 & 1.0 & 1.1 & 1.2 & 1.2 & 1.3 & 1.3 \\ 1.0 & 1.0 & 1.0 & 1.0 & 1.2 & 1.2 & 1.3 & 1.3 \\ 1.1 & 1.0 & 1.0 & 1.0 & 1.2 & 1.2 & 1.3 & 1.4 \\ 1.1 & 1.0 & 1.2 & 1.0 & 1.2 & 1.2 & 1.3 & 1.4 \\ 1.2 & 1.2 & 1.2 & 1.2 & 1.2 & 1.2 & 1.4 & 1.5 \\ 1.2 & 1.2 & 1.2 & 1.2 & 1.2 & 1.2 & 1.4 & 1.5 \\ 1.3 & 1.3 & 1.3 & 1.3 & 1.4 & 1.5 & 1.5 & 1.5 \\ 1.3 & 1.3 & 1.4 & 1.4 & 1.5 & 1.5 & 1.5 & 1.5 \end{bmatrix}, \quad (6)$$

and M_1^T is the transpose of M_1 .

The entries in Q_L are an indication of the insensitivities of the LOT coefficients subject to quantization noise. That is, the smaller the entry in Q_L , the more important the corresponding LOT coefficient is in maintaining the subjective imperceptibility when the image block is distorted by quantization noise. Therefore, motivated by the concept of spread spectrum watermarking [10], we choose the five most visually important AC coefficients in each block, whose corresponding entries in Q_L have the least values (except the one located in the (0,0) position that is excluded from watermark embedding), to bear the watermark. The watermark is embedded in the following way

$$X'_k(i_{n,k}, j_{n,k}) = X_k(i_{n,k}, j_{n,k}) + \alpha Q_L(i_{n,k}, j_{n,k}) w(n), \quad (7)$$

where $w(n)$ is the watermark element, and $(i_{n,k}, j_{n,k})$ denotes the specified position to bear the watermark in the k th block. X denotes the LOT coefficient of the original image, and X' is the corresponding watermarked coefficient. α determines the overall intensity of the watermark embedding energy, which usually takes a value between 1/6 to 1.

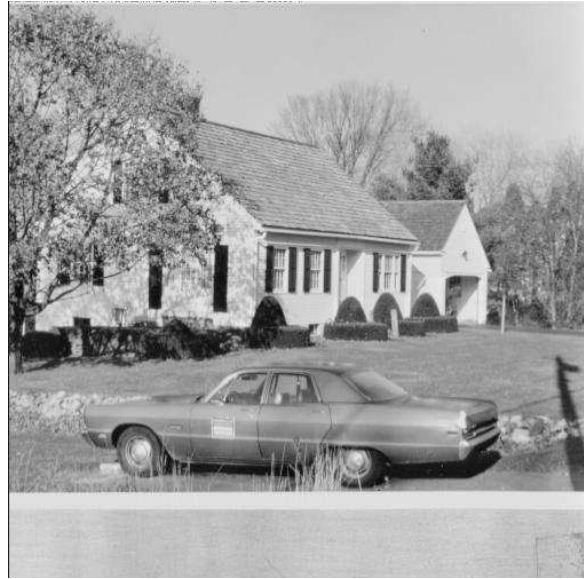
3.4. Watermark Detection and Identification

We need the original unwatermarked image to implement the watermark detection. We take LOT for both the original image and the possibly corrupted watermarked image, as well as the location and intensity information. For each watermarked frequency component, we extract the watermark using

$$\hat{w}(n) = (\hat{X}'_k(i_{n,k}, j_{n,k}) - X_k(i_{n,k}, j_{n,k})) / (\alpha Q_L(i_{n,k}, j_{n,k})). \quad (8)$$



(a) *crowd*



(b) *umas*

Figure 4. Original images

Let \hat{W} denote the extracted watermark vector, and W a candidate watermark. We identify the watermark by obtaining the similarity (correlation) as follows

$$\text{sim}(W, \hat{W}) = \frac{W \cdot \hat{W}}{\sqrt{\hat{W} \cdot \hat{W}}}, \quad (9)$$

where “ \cdot ” denotes the inner production. It is easy to show that if W is a sequence of identically independently distributed (i.i.d.) random variables with zero-mean unit-variance, and also independent of \hat{W} , $\text{sim}(W, \hat{W})$ is a zero-mean unit-variance Gaussian random variable. If $\text{sim}(W, \hat{W})$ is greater than a threshold, we claim that \hat{W} comes from the original watermark W , hence identifying the embedded watermark. In our implementation, we chose 6 as the threshold, which corresponds to a false alarm probability of $10^{-9} (= 1 - \Phi(6))$, where $\Phi(x)$ denotes the cumulative distribution function, i.e. c.d.f., of a Gaussian random variable with zero-mean unit-variance). Note that when $\hat{W} = W$, $\text{sim}(W, \hat{W})$ achieves its maximum value $143.11 (= \sqrt{5 \times 512 \times 512 / (8 \times 8)})$.

4. EXPERIMENTAL RESULTS

We used our LOT watermarking scheme to watermark several 512×512 gray scale images shown in Figure 4. We also implemented a block-based DCT watermarking scheme for comparison. The only difference between the two schemes is the transform. The Stirmark 3.1 benchmark [20] is used to attack the watermarked images and test the robustness of the embedded watermark. Some major results are given in Table 1, Table 3, and Figure 5.

It can be observed from Table 1 that under the same embedding intensity α , our LOT scheme is slightly more resistant to blurring attacks, but slightly more vulnerable to sharpening attacks.

Figure 5 suggests that using the same α , the DCT watermarking is slightly more resistant to lossy JPEG compression. We have argued that one distinguished advantage of the block-based LOT, as opposed to the block-based DCT, is its capability of reducing block artifacts. Thus we can choose a slightly larger α for our LOT method to increase its robustness without compromising the transparency. For example, if we chose $\alpha = 0.60$, the LOT watermarking scheme will outperform the DCT scheme with $\alpha = 0.50$ as described in Table 2. The

Table 1. Watermark identification after attacks using common filtering operations ($sim(W, \hat{W})$)

Image	Scheme	α	Median		Gaussian	Average	Sharpen
			2×2	3×3	3×3	3×3	
crowd	LOT	0.25	15.8	32.3	48.4	36.7	30.2
	DCT		15.7	28.9	44.1	34.5	33.2
	LOT	0.50	31.9	61.6	83.5	67.2	56.5
	DCT		30.8	57.5	76.9	62.9	62.2
	LOT	2.00	94.2	127.0	133.8	127.7	120.0
	DCT		90.0	124.7	130.1	122.7	120.5
umas	LOT	0.25	19.3	39.3	54.7	39.4	33.8
	DCT		15.7	35.7	42.1	33.2	40.5
	LOT	0.50	36.3	71.2	90.8	70.8	62.6
	DCT		31.1	66.2	75.7	62.7	71.5
	LOT	2.00	98.3	130.4	135.8	129.4	124.8
	DCT		91.2	129.0	130.7	123.8	125.8

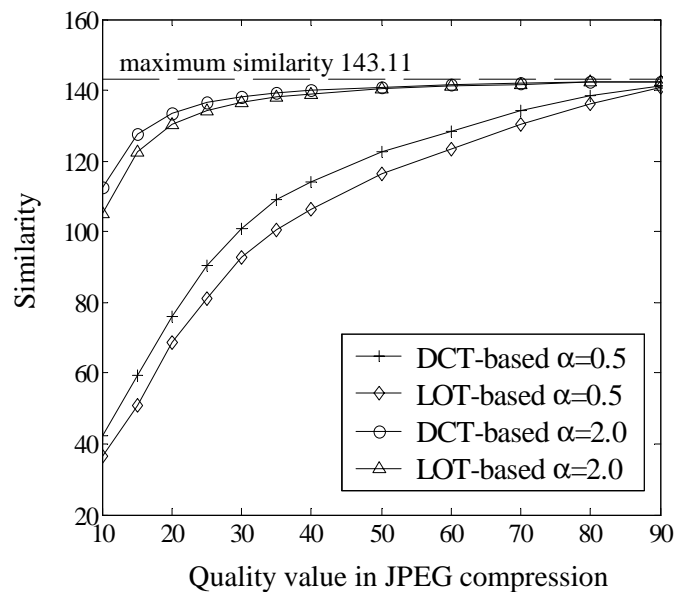


Figure 5. LOT vs. DCT watermarking using a lossy JPEG compression attack to *crowd*

Table 2. JPEG attack performance of LOT watermarking with increased α for *crowd* ($sim(W, \hat{W})$)

Quality	10	20	30	40	50	70	90
LOT ($\alpha = 0.60$)	42.8	78.0	101.6	114.8	123.0	133.8	141.4
DCT ($\alpha = 0.50$)	42.1	76.1	100.7	114.0	122.8	134.0	141.4

Table 3. LOT watermark identification using a cropping attack to *crowd* ($\alpha = 0.6$)

Cropping (upper-left position (x, y))	Area size: 171×170			Area size: 51×51		
	(193, 169)	(292, 45)	(57, 290)	(150, 125)	(315, 53)	(15, 252)
$sim(W, \hat{W})$	48.1	47.5	49.3	14.9	14.1	14.7

LOT watermarked image has the same or even better subjective quality compared with the DCT watermark, as shown in Figure 6. The difference images between the original image and the watermarked images are also given, showing that LOT effectively reduces the block artifacts.

Our LOT watermark survives the cropping attack, as shown in Table 3. Intuitively, if we lose 90% of the image, we lose 90% of the watermark sequence. However, considering our goal is to verify the similarity between the extracted sequence and the original sequence, losing 90% of the extracted data only makes the similarity function to be $1/\sqrt{10}$ of its original value. Usually the correlation is still well above 6, so we can still make a correct decision after a severe cropping. We chose two cropping sizes, 171×170 and 51×51 , which correspond to $1/9$ and $1/100$ of the original image size respectively. The theoretical values for the maximum similarity between the extracted watermark and a candidate watermark (i.e. $sim(W, W)$) are $47.70(= \sqrt{5} \times 512 \times 512/9)$ and $14.31(= \sqrt{5} \times 512 \times 512/100)$ for the above two size choices. As discussed in [10], since we are using a non-blind approach in our watermarking, i.e., we have the original image available upon watermark detection, the watermark detector can easily obtain the position of the watermarked cropped area relative to the original image (such as using image correlation). For each cropping size, we tried three different cropping locations, as given in Table 3 and Figure 7.

In addition, we used the Stirmark 3.1 JPEG attack on the cropped area, and our LOT watermarking can still survive as long as the cropping size is large enough, as shown in Table 4.

One approach to improve our LOT watermarking scheme is to adaptively adjust the number of watermark elements embedded in each block rather than fix it at 5. A better transparency-robustness balance should be expected.

5. CONCLUSION

In this paper, we presented a LOT-based adaptive image watermarking scheme. LOT inherits all the best properties of the DCT, such as decorrelation and energy compaction. Specifically, LOT greatly reduced block artifacts compared with DCT, making the embedded watermark less perceptible with same embedding energy. In addition, we used the HVS perceptual analysis to embed the watermark in an adaptive manner. Our experiments showed that our scheme obtains a good balance between robustness and transparency of the watermark.

Table 4. LOT watermark identification under cropping and JPEG lossy compression attacks to *crowd* (cropping size of 171×170 with upper-left corner (292, 45); $\alpha = 0.6$)

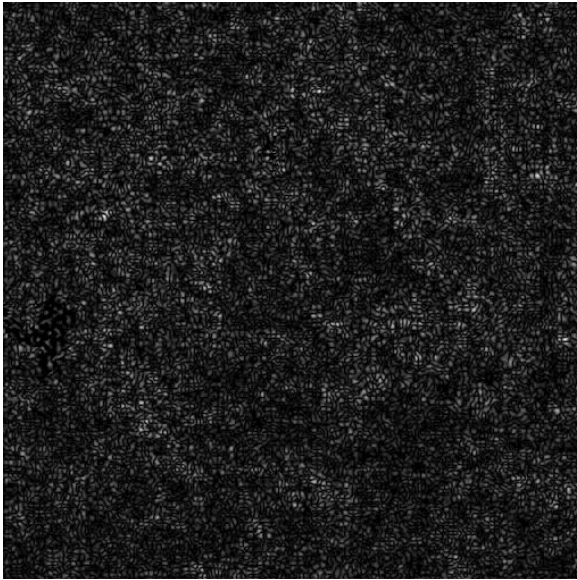
Quality	10	15	20	30	40	50	70	90
$sim(W, \hat{W})$	6.6	8.2	11.3	16.6	20.8	24.0	33.4	43.5



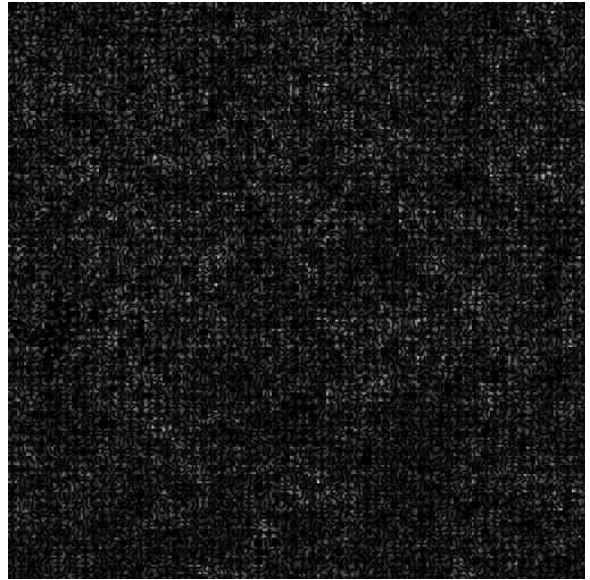
(a) Watermarked image using LOT ($\alpha = 0.6$)



(b) Watermarked image using DCT ($\alpha = 0.5$)



(c) Difference image using LOT ($\alpha = 0.6$)



(d) Difference image using DCT ($\alpha = 0.5$)

Figure 6. LOT vs. DCT image watermarking for *crowd*



(a) Original image with the cropping area marked

(292, 45)



(462,214)

(b) Enlarged cropped area (171 × 170)

Figure 7. Image cropping attack

REFERENCES

1. R. B. Wolfgang, C. I. Podilchuk, and E. J. Delp, "Perceptual watermarks for digital images and video," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1108–1126, July 1999.
2. C. D. Vleeschouwer, J.-F. Delaigle, and B. Macq, "Invisibility and application functionalities in perceptual watermarking - an overview," *Proceedings of the IEEE*, vol. 90, no. 1, pp. 64–77, January 2002.
3. C. I. Podilchuk and E. J. Delp, "Digital watermarking: algorithms and applications," *IEEE Signal Processing Magazine*, vol. 18, no. 4, pp. 33–46, July 2001.
4. M. D. Swanson, M. Kobayashi, and A. H. Tewfik, "Multimedia data-embedding and watermarking technologies," *Proceedings of the IEEE*, vol. 86, no. 6, pp. 1064–1087, June 1998.
5. A. M. Eskicioglu and E. J. Delp, "An overview of multimedia content protection in consumer electronics devices," *Signal Processing: Image Communication*, vol. 16, no. 7, pp. 681–699, April 2001.
6. F. Hartung and F. Ramme, "Digital rights management and watermarking of multimedia content for m-commerce applications," *IEEE Communications Magazine*, vol. 38, no. 11, pp. 78–84, November 2000.
7. F. Hartung and M. Kutter, "Multimedia watermarking techniques," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1079–1107, July 1999.
8. J. A. Bloom, I. J. Cox, T. Kalker, J.-P. M. G. Linnartz, M. L. Miller, and C. B. S. Traw, "Copy protection for DVD video," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1267–1276, July 1999.
9. G. Voyatzis and I. Pitas, "The use of watermarks in the protection of digital multimedia products," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1197–1207, July 1999.
10. I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1673–1687, December 1997.
11. I. J. Cox, M. L. Miller, and A. L. McKellips, "Watermarking as communications with side information," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1127–1141, July 1999.
12. B. Tao and B. Dickinson, "Adaptive watermarking in the DCT domain," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, Munich, Germany, April 21–24, 1997, pp. 2985–2988.

13. H. S. Malvar and D. H. Staelin, "The LOT: transform coding without blocking effects," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 4, pp. 553–559, April 1989.
14. S. Pereira, J. J. K. O. Ruanaidh, and T. Pun, "Secure robust digital watermarking using the Lapped Orthogonal Transform," in *Proceedings of the SPIE International Conference on Security and Watermarking of Multimedia Contents*, vol. 3657, San Jose, CA, January 25–27, 1999, pp. 21–30.
15. S. Pateux and G. L. Guelvouit, "Practical watermarking scheme based on wide spread spectrum and game theory," *Signal Processing: Image Communication*, vol. 18, no. 4, pp. 283–296, April 2003.
16. Y.-C. Li, T.-H. Wu, and Y.-C. Chen, "A scene adaptive hybrid video coding scheme based on the LOT," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 1, pp. 92–103, February 1998.
17. N. Jayant, J. Johnston, and R. Safranek, "Signal compression based on models of human perception," *Proceedings of the IEEE*, vol. 81, no. 10, pp. 1385–1422, October 1993.
18. S. H. Tan, K. K. Pang, and K. N. Ngan, "Classified perceptual coding with adaptive quantization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 4, pp. 375–388, August 1996.
19. G. K. Wallace, "The JPEG still picture compression standard," *IEEE Transactions on Consumer Electronics*, vol. 38, no. 1, pp. xviii–xxxiv, February 1992.
20. Stirmark 3.1. [Online]. Available: <http://www.petitcolas.net/fabien/watermarking/stirmark31/>