SHORT PAPER
International Journal of Recent Trends in Engineering, Vol 2, No. 3, November 2009

Digital Image Watermarking Using SPIHT

Patil Ramana Reddy[1], Munaga.V.N.K.Prasad[2] and D.Srinivasa Rao[3]
[1]Associate Professor, ECE Department, JNTUCE, Anantapur, A.P, India, Email: prrjntu@gmail.com
[2] Assistant Professor, Institute for Development & Research in Banking Technology (IDRBT), Hyderabad, A.P, India
[3] Professor, ECE Department, JNTUCE, Hyderabad, A.P, India

ABSTRACT:
In recent years access to multimedia data has become much easier due to the rapid growth of internet. The same access makes unauthorized copying and distributing of multimedia data much easier. So there is strong need for protection of these multimedia data. Digital watermarking is widely used for copyright protection of these data. Digital watermarking is process of inserting identification code or logo in multimedia data without affecting the quality of multimedia data. In this paper compression-domain watermarking technique based on the SPIHT coding is presented. In contrast of the conventional approaches that incorporate watermarks into the transformed coefficients, the implemented method impresses the binary watermark sequence directly on the bit stream generated after SPIHT coding process. The security of the system is assured since watermark is embedded in compressed domain. The performance of the system can be enhanced with minimal complexity by a joint optimization of quantization and watermarking. An experimental result demonstrates that the watermarking is robust to signal processing and geometric attacks. This watermarking technique is implemented for color images. Watermark is embedded in blue plane of image after SPHIT coding.

I. Introduction

The ever increasing amount of information that is stored and transmitted digitally raises the problem of copyright protection. The owners of multimedia data are unwilling to allow the distribution of their documents in a networked environment because the ease of reproducing digital data in their original shape makes copyright violation easy. The digital watermarking of multimedia has been proposed as a possible solution, since it allows the dissemination of data to be tracked. Digital watermarking is a technique of inserting identifiable code in the host data. Extracting and verifying the watermark embedded in the host data helps to identity the ownership of content owners. A good watermarking system for images should satisfy the following requirements [1]:

- **Transparency:** The embedded watermark should degrade the quality of images to a minimal degree.
- **Robustness**.: The embedded watermark should withstand common image processing operations such as cropping, rotation, filtering and compression
- **Security.** The watermarking scheme must be secure even when the embedding algorithm is made public. Security is usually achieved by using cryptographic techniques.
- **Appropriate complexity**. The computation and memory requirements should be small relative to the compression/decompression processes, especially for real-time applications. Apparently, these requirements may conflict with each other, and careful design is needed for tradeoff.

Image watermarks can be inserted in the spatial domain by directly modulating the pixel intensities [2]. Alternatively, one can embed the watermark in the frequency domain by changing the coefficients after performing discrete cosine transform (DCT) [3] or discrete wavelet transform (DWT) [4]. In general, frequency-domain approaches have better performance since images manifest their perceptual characteristics in the frequency domain. Moreover, the spread spectrum techniques widely used in secure communications can easily be incorporated [5].

II. About EZW and SPIHT

The concept of lossy image coding based on wavelet tree quantization was initially introduced in embedded zero-tree wavelet (EZW) coding [6] by Shapiro in 1993. In the embedded zero-tree wavelet concept, wavelet coefficient at a given scale is considered insignificant with respect to a given threshold. Most of the coefficients in these sub bands may have very small magnitudes and thus low energy, so few significant coefficients are exploited in EZW to give an efficient coding scheme. The SPIHT algorithm, developed by Said and Pearlman in 1996 [7], is a fast and efficient image compression algorithm that works by testing ordered wavelet coefficients for significance and quantizing only the significant coefficients. The high coding efficiency obtained by this algorithm is due to a group testing of the coefficients that belong to a wavelet tree. Group testing is advantageous because of the inter-band correlation that exists between the coefficients belonging to a tree. The SPIHT algorithm groups the wavelet coefficients and trees into sets based on their significance information. The encoding algorithm consists of two main stages, sorting and refinement. In sorting stage, the threshold for significance is set as ‘2^n’, where ‘n’ is the bit level, and its initial value is determined by the number of bits required to represent the wavelet coefficient with the maximum absolute value. Significance of trees is obtained by checking all the detail coefficients. Approximation coefficients are tested as individual entries. The initial listing that determines the order in which significance tests are done is predetermined for both the approximation coefficients as well as the trees. The algorithm searches each tree, and partitions the tree into one of three lists: 1) the list of significant pixels (LSP) containing the coordinates of pixels found to be significant at the current threshold; 2) The list of insignificant pixels (LIP), with pixels that are not significant at the current threshold; and 3) the list of insignificant sets (LIS), which contain information about trees that have all the constituent entries to be insignificant at the current threshold. If a coefficient or a tree is found to be insignificant, a “0” bit is sent to the output bit stream and the corresponding coordinates are moved to the LIP or LIS respectively, for subsequent testing at a lower bit level. When a coefficient is found to be significant, a “1” bit and a sign bit are sent and its coordinate is moved to the LSP. If an LIS member is found to be significant, a “1” bit is sent and the tree is partitioned into its offspring and descendants of offspring. The offspring are moved to the end of the LIP and subsequently tested for significance at the same bit level. The offspring are also moved to the LIS as the roots ,if a coefficient or a tree is found to be insignificant, a “0” bit is sent to the output bit stream and the corresponding coordinates are moved to the LIP or LIS respectively, for subsequent testing at a lower bit level. When a coefficient is found to be significant, a “1” bit and a sign bit are sent out and its coordinate is moved to the LSP. If an LIS member is found to be significant, a “1” bit is sent out and the tree is partitioned into its offspring and descendants of offspring. The offspring is moved to the end of the LIP and subsequently tested for significance at the same bit level. The offspring is also moved to the LIS as the roots of their corresponding descendant sets that will be subsequently tested for significance at the same bit level. The bit level is successively lowered and above iterations is repeated to achieve maximum efficiency.

III. Watermarking Technique

The proposed watermark-embedding technique is shown in Figure 1. A binary logo is chosen as watermark which is inserted in selective output bits after SPHIT refinement pass for watermark embedding. Here the SPHIT coding is performed on extracted blue plane of color image since variations in blue plane coefficients will not lead to severe distortions of color image. The rest of the SPIHT coding process is unchanged. Once the embedding locations are decided, they are replaced with watermark bits. These embedding locations are stored as a secure key and used for detection. The watermarked image is obtained by applying SPIHT decoding process on modified blue plane and then merged with red and green plane. Some of the important parameters determine the performance of the watermarking technique, which include

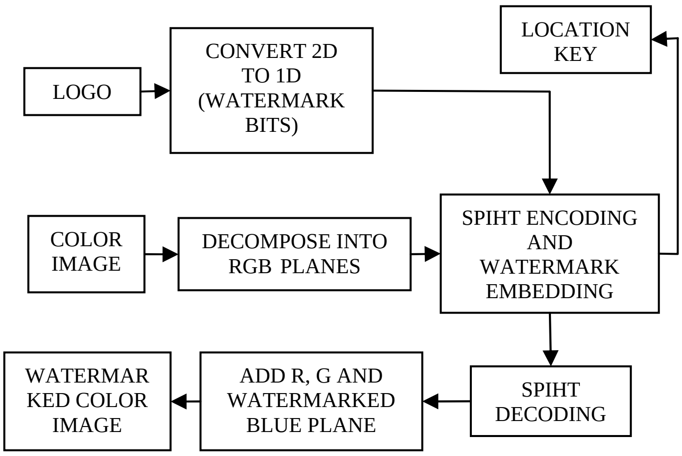

Figure 1 Block diagram of watermark embedding

• **Watermark locations**: Embed the watermark bits in wavelet coefficients that are found significant in the SPIHT coding process.

• **Watermarking strength**: The watermark can be thought of as a distortion added to the host image. It is necessary to choose appropriate watermarking energy for balancing the requirements of transparency and robustness. In this technique the watermark is added after quantization, so the effects of quantization upon watermarks can easily be controlled. One can avoid embedding of watermarks into the coefficients that are vulnerable to attacks.

• **Length of the binary watermark:** Length of the watermark is appropriately chosen to avoid the distortion of the image.

The detection process of this watermarking technique is shown in Figure 2. With the aid of the location key, the embedded watermark can be retrieved by the normal SPIHT coding process. The extracted watermark bits are then compared with the original watermark bits by a similarity test. An appropriate threshold is determined for watermark identification, which aims to optimize the watermark detection process. It has been observed [1] that geometric attacks (including rotation, cropping, scaling and others), though hardly affect the image quality, could effect many watermarking techniques. Geometric transforms usually result in the loss of synchronization between the watermark pattern and the watermarked image. Synchronization may be retrieved at the detector by performing the template matching the wavelet coefficients of a test image are sorted and arranged into the same locations as those of its corresponding watermarked image.

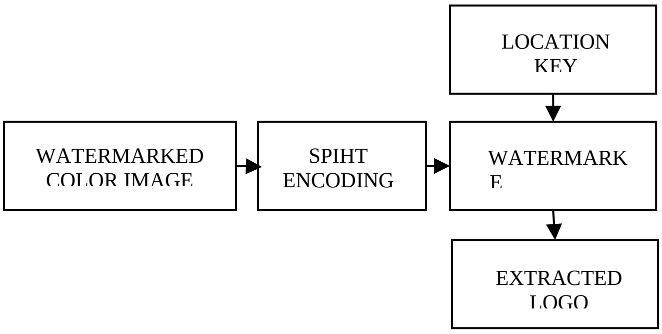

Figure 2. Block diagram of watermark detection

IV. Simulation Results

This watermarking technique is tested with the 512 X 512, 24 bit color LENA image, tests with other images yield similar results. Since the SPIHT and other zero tree-based algorithms are distinguishably superior to JPEG in the low bit-rate situations, It should be noted that this technique is especially effective at low bit rates, since the subsequent refinements may introduce distortions to the embedded watermark. The watermark is a binary logo of size (64X64) which is converted into one row vector of size 4096 X 1 as the watermarking signal, these watermark bits are embedded in the output of SPIHT refinement bits. Performance metrics of watermarking technique such as MSE, PSNR, SIM and BCR are computed with and without attacks. The PSNR of the watermarking technique is reasonably high and the artifacts introduced by watermark embedding are almost invisible. However, if an attacker ignores all the refinement bits at this threshold, the obtained PSNR will be less than 30 dB, which is not satisfactory enough in most of the cases. Simulated results without attacks i.e. original image, watermarked image, difference image, original and extracted watermarks are shown in Table 1,Figure.3 and 4.

Mean Square Error (MSE)	2.57
Peak Signal to Noise Ratio(PSNR)	44.57
Similarity Index Measurement (SIM)	0.998
Bit Correct Rate (BCR)	1
Embedding Time	12.14 Sec
Extraction Time	9.45 Sec

Table 1. Performance metrics of watermarking technique without attacks

Original Watermark Extracted Watermark

Figure 3. Original and Extracted watermarks without attacks

Original image Watermarked image Difference image

Figure 4. Simulation results without attacks

V. Results with Attacks

(A) **Addition of Noise**: Adding Salt Pepper noise and Gaussian noise to the watermarked image and by varying noise density in case of Salt Pepper noise and variance in case of Gaussian noise. The obtained results show PSNR is decreasing as noise intensity increases, SIM is reasonably well indicating quality of extracted image. In case Gaussian case PSNR is quite good, But SIM is decreasing as variance increases. In both cases BCR is '1' indicating good watermark recovery. The obtained results are tabulated in Tables 2 and 3

NOISE DENSITY	**MSE**	**PSNR**	**SIM**	**BCR**
0.002	**42.56**	**31.84**	**0.78**	**1**
0.004	**85.24**	**28.82**	**0.76**	**1**
0.008	**165.56**	**25.94**	**0.72**	**1**
0.01	**207.63**	**24.95**	**0.71**	**1**
0.03	**641.78**	**20.78**	**0.69**	**1**
0.06	**1250**	**17.16**	**0.66**	**1**
0.1	**2110**	**14.87**	**0.65**	**1**
0.2	**4270**	**11.81**	**0.64**	**1**

Table 2. Performance metrics with addition of Salt Pepper noise attack

VARIANCE	**MSE**	**PSNR**	**SIM**	**BCR**
0.04	**0.0016**	**76.099**	**0.665**	**1**
0.06	**0.0036**	**72.56**	**0.65**	**1**
0.1	**0.010**	**68.12**	**0.642**	**1**
0.2	**0.04**	**62.08**	**0.632**	**1**
0.4	**0.16**	**56.08**	**0.55**	**1**

Table 3. Performance metrics with addition of Gaussian noise attack

(B) **Cropping:** When watermarked image is cropped even for small areas the recovered watermark quality is very poor, It is because removal of any part of the image effects a lot, since watermark is embedded after SPHIT encoding. The obtained results are tabulated in Table.4

C.A	MSE	PSNR	SIM	BCR
16X16	130.79	26.96	0.75	1
32X32	316.34	23.13	0.65	1

Table 4. Performance metrics with Cropping attack

(C) Rotation: When watermarked image is rotated even for small angle, the recovered watermark quality is very poor, It is because watermarked image loses synchronization with original image. The obtained results are tabulated in Table.5

R.A(Degrees)	**MSE**	**PSNR**	**SIM**	**BCR**
0.5	**406.4**	**22.04**	**0.75**	**1**
1	**856.9**	**18.9**	**0.53**	**1**

Table 5. Performance metrics with rotation attack

VI .Conclusions

In this paper a new compression-domain image watermarking technique is presented which is based on SPIHT coding. This method impresses the watermark bits directly into the bit stream generated after SPHIT coding process. The security of the technique is assured by the location key chosen using watermark. A simple relocation strategy is adopted in the detection process. Experimental results show that this method could simultaneously achieve transparency, security, and robustness with minimal complexity.

REFERENCES

[1]. G. C. Langelaar, I. Setywan, and R. L. Lagendijk, "Watermarking digital image and video data," *IEEE Signal Process. Mag.*, vol. 17, no. 5, pp. 20-46, Sep. 2000.

[2]. I. Pitas, "A method for watermark casting on digital images," *IEEE Trans. Circuits Syst. Video Technol.*, vol.8, no. 6, pp. 775-780, Oct. 1998.

[3]. J. R. Hernandez, M. Amado, and F. Perez-Gonzalez, "DCTdomain watermarking techniques for still images: detector performance analysis and a new structure," *IEEE Trans. Image Process.*, vol. 9, no. 1, pp. 55-68, Jan. 2000.

[4]. W. Zhu, Z. Xiong, and Y.-Q. Zhang, "Multiresolution watermarking for images and video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 4, pp. 545-550, June 1999.

[5]. I. J. Cox, J. Killian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia"*IEEE Trans. Image Process.*, vol.6, no. 12, pp.1673-1687,Dec. 1997.

[6]. J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, no. 12, pp. 3445-3462, Dec. 1993.

[7]. A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in Hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 243-250, June 1996.