Image Coding with Adaptive Wavelet Packet Trees

Hsi-Chin Hsin and Tze-Yun Sung

***Abstract*—Embedded zero-tree image coding in the wavelet domain has drawn a lot of attention. Among noteworthy algorithms is the set partitioning in hierarchical trees (SPIHT). Typically, most of images' energy is concentrated in the low frequency subbands. For images with textures, however many middle-high frequency wavelet coefficients are likely to become significant in the early passes of SPIHT; thus the coding results are often insufficient. The middle and high frequency subbands of an image may demand a further decomposition with adaptive basis functions. As wavelet packet transform offers a great diversity of basis functions, we propose a quad-tree based adaptive wavelet packet transform to construct adaptive wavelet packet trees for zero-tree image coding. Experimental results show that the coding performance can be significantly improved especially for fingerprints images.**

***Index Terms*—adaptive wavelet packet transform, embedded zero-tree coding, SPIHT.**

I. INTRODUCTION

Wavelet transform provides many desirable properties such as multiresolution representation, scalability, and progressive transmission that are beneficial to the image compression applications [1]. If a wavelet coefficient is insignificant, all the spatially related wavelet coefficients (in the successively higher frequency subbands of the same orientation) are likely to be insignificant. Shapiro first introduced the above-mentioned self similarity of wavelet coefficients and proposed the embedded zero-tree wavelet (EZW) algorithm [2]. The improved version known as set partitioning in hierarchical trees (SPIHT) has been considered a benchmark [3]. Mukherjee proposed a vector extension called VSPIHT, in which wavelet coefficients are first grouped into small vectors and then coded by SPIHT [4]. Besides the famous EZW and SPIHT algorithms based on the self-similarity of wavelet coefficients, there are alternative algorithms based on the energy clustering of wavelet coefficients [5]-[7]. Taubman proposed the EBCOT algorithm [8], which has been adopted by JPEG2000 [9]. In general, JPEG2000 outperforms JPEG; however, the computational complexity of EBCOT is highly intense [10].

For images with textures, many high frequency wavelet coefficients are noted for their significance and therefore they are likely to become significant after several code passes of SPIHT, which substantially degrades the coding performance. As discussed in [11], a great diversity of basis functions obtained by wavelet packet transform is desirable for decomposing the significant wavelet subbands of an image. Rajpoot proposed a tree structure with arbitrary subband decompositions via wavelet packet transform [12]. In which, a reorganization scheme is required to solve the so-called parenting conflict problem: the spatial resolution of a parent node is finer than that of the children nodes; and moreover, the constructed wavelet packet trees may not be quad-trees. In [13], high frequency wavelet coefficients are first decomposed into wavelet packet coefficients and then rearranged to construct hierarchical quad-trees. In this paper, a quad-tree based algorithm is proposed to construct adaptive wavelet packet trees, which can be efficiently coded by SPIHT for image compression.

The remainder of this paper proceeds as follows. In Section II, wavelet transform, wavelet packet transform, and SPIHT are reviewed. Section III describes the proposed algorithm for constructing adaptive wavelet packet trees. Experimental results are presented in Section IV. Conclusion is given in Section V.

II. REVIEW OF WAVELET TRANSFORM, WAVELET PACKET TRANSFORM, AND SPIHT

Wavelet transform (WT) offers an efficient multiresolution representation. Figure 1(a) shows 3-level WT, where HL_ℓ, LH_ℓ, and HH_ℓ denote subbands of wavelet coefficients $D_\ell^1(m,n)$, $D_\ell^2(m,n)$, and $D_\ell^3(m,n)$ representing the detail information at resolution ℓ in the horizontal, vertical, and diagonal orientations, respectively; LL_3 denotes the approximation at the coarsest resolution 3. The original image can be taken as the scaling coefficients $S_0(m,n)$ at the finest resolution 0. After 1-level WT, $S_0(m,n)$ is decomposed into $S_1(m,n)$, $D_1^1(m,n)$, $D_1^2(m,n)$, and $D_1^3(m,n)$. Moreover, $S_0(m,n)$ can be exactly reconstructed from $S_1(m,n)$, $D_1^1(m,n)$, $D_1^2(m,n)$, and $D_1^3(m,n)$ by inverse wavelet transform.

In WT, only the scaling coefficients of an image are successively decomposed to generate wavelet coefficients. However, both scaling and wavelet coefficients can be

This work was supported in part by the National Science Council of Taiwan, under Grant NSC 96-2221-E-239-036.

Hsi-Chin Hsin is with the Department of Computer Science and Information Engineering, National United University, Taiwan (e-mail: hsin@nuu.edu.tw)

Tze-Yun Sung is with the Department of Microelectronics Engineering, Chung Hua University, Taiwan

decomposed, which leads to wavelet packet transform (WPT). Figure 2(a) shows an example of 3-level WPT, where the diagonal high frequency wavelet coefficients $D_1^3(m,n)$ and $D_2^3(m,n)$ are decomposed into wavelet packets.

After WT, the spatially related wavelet coefficients taken from all the subbands of the same orientation can be grouped to form hierarchical trees. The tree hierarchy is based on the resolution level; the leaf nodes are at the finest resolution. Each non-leaf node can be a parent node with four children nodes at the next finer resolution. Figure 1(b) shows a wavelet tree in the diagonal orientation. In many cases, if a parent node is insignificant with respect to a given threshold, all the descendant nodes are likely to be insignificant with respect to the same threshold, and thus these nodes may form a zero-tree and can be efficiently coded by SPIHT. The following is the SPIHT algorithm presented in steps [3].

1) Initialization: Set the initial threshold value: $T_b = 2^N$ for $b = 1$, where $N = \lfloor \log_2(\max_{(m,n)} |c_{m,n}|) \rfloor$, $c_{m,n}$ is the wavelet coefficient at position (m,n).
2) Sorting pass: Identify wavelet coefficients with $T_b < |c_{m,n}| \le 2T_b$. Output their respective signs and positions.
3) Refinement pass: Output the b^{th} bit of the significant wavelet coefficients with $|c_{m,n}| > 2T_b$, which have been identified in the previous passes.
4) Increase b by one, divide the threshold value by 2, and go to step 2.

The scan pass, i.e. step 2 followed by step 3, is repeatedly performed until a given bit rate is reached. In the sorting pass, if an insignificant node stored in the list of insignificant pixels (LIP) becomes significant, it is removed to the list of significant pixels (LSP), and the sign bit is coded; otherwise, it remains in LIP. For a zero-tree with the root node stored in the list of insignificant sets (LIS), if none of the descendant nodes becomes significant, the root node remains in LIS; otherwise, it becomes a broken tree and a further decomposition takes place to locate the significant nodes. In the refinement pass, every significant node stored in LSP is refined by updating the magnitude information.

III. Proposed Adaptive Wavelet Packet Trees for Image Coding

As wavelet packet transform (WPT) offers more subband decompositions than wavelet transform (WT), zero-tree coding with adaptive wavelet packets is thus proposed. The strategy for adaptive WPT is based on the significance of the non-offspring descendants of the constructed quad-trees.

In the wavelet domain, the construction of hierarchical trees is straightforward due to the pyramid structure of WT. More specifically, the constructed wavelet trees of an image are based on the spatial relations between wavelet coefficients across subbands of the same orientation. For images with textures, the middle and high frequency subbands (in the wavelet domain) may demand a further decomposition. In [13], we proposed an efficient scheme to construct wavelet packet trees by rearranging wavelet packet coefficients, which are quad-trees and therefore can be efficiently coded by using the SPIHT algorithm. Figure 2 shows an example of 3-level WPT and a hierarchical wavelet packet tree in the diagonal orientation, where the arrows drawn between subbands indicate the relations between a parent node and its children nodes before rearranging wavelet packet coefficients.

In SPIHT, if a zero-tree becomes significant, each offspring node (of the tree) is coded as a single pixel, but on the other hand the non-offspring nodes are expected to be insignificant and then coded in a group manner. Thus, the desirable quad-trees with adaptive wavelet packets should be the one with as many insignificant non-offspring nodes as possible. Specifically, after L-level WT, an image is decomposed into multiresolution wavelet subbands (from the coarsest resolution L to the finest resolution 1). For each wavelet subband at resolution ℓ ($\le L-2$), the wavelet packet coefficients generated by p-level WPT are rearranged; notice that wavelet packet coefficients are actually wavelet coefficients and therefore no rearrangement takes place while $p = 0$. Together with the subbands at coarser resolutions: ℓ+1 and ℓ+2, a collection of 3-level quad-trees can be obtained, in which the rearranged wavelet packet coefficients (obtained from the wavelet subband at resolution ℓ) are the leaf nodes. If any descendant nodes of a tree become significant, it is called a broken tree. For computational simplicity, the following measure indicating the weighted number of broken (3-level) quad-trees due to the (non-offspring) leaf nodes is proposed.

$$S_p = \sum_{t \in B_p} N(t) \cdot Sig(t) \tag{1}$$

where B_p denotes the set of the constructed wavelet packet trees with leaf nodes obtained from a wavelet subband by using p-level WPT, $Sig(t) = 1$ if t is a broken tree; otherwise $Sig(t) = 0$, and $N(t)$ denotes the number of significant groups of 4 neighboring leaf nodes in a broken tree: t.

In view of zero-tree coding, it is desirable that the sum of the proposed measures over the individual wavelet subbands of an image is minimized. For the sake of simplicity, an efficient algorithm is proposed to construct hierarchical quad-trees with adaptive wavelet packets. The pseudo-code is as follows:

1) Decompose the input image using L-level WT. Set $\ell = L$-2.
2) For each wavelet subband at resolution ℓ, set $p = 1$.
   Loop:

Compute the wavelet packet coefficients using p-level WPT.

Rearrange these wavelet packet coefficients.

Together with the subbands (of the same orientation) at resolutions: $\ell+1$ and $\ell+2$, construct 3-level quad-trees, and compute S_p (defined in equation (1))

If $\frac{S_{p-1}-S_p}{S_{p-1}}\times 100\% < T_r$, exit the loop; otherwise increase p by 1.

Continue the loop until $p = L-\ell$.

3) Decrease ℓ by 1. If $\ell \geq 1$, go to step 2.

As the number of the low frequency wavelet coefficients (at resolutions L-1 and L) is quite small, no further decomposition is performed. Recall that wavelet packet coefficients with $p = 0$ are actually wavelet coefficients.

As the proposed adaptive wavelet packet tree is a quad-tree, an efficient image coder can be obtained by applying SPIHT to the adaptive wavelet packet trees of an image, which is called set partitioning in adaptive wavelet packet trees (SPIAWPT). The following is SPIAWPT presented in step:

1) Initialization: Decompose an image using the proposed quad-tree based adaptive wavelet packet transform, construct the adaptive wavelet packet trees, and determine the initial threshold: $T_b; b = 1$ such that all the tree nodes are in the range of $[-2T_1, 2T_1]$.
2) Sorting pass: Identify the significant tree nodes in comparison with T_b. If a tree node becomes significant, output its sign bit.
3) Refinement pass: Output the refinement bits of the significant tree nodes found previously.
4) Increase b by one, compute $T_b = T_{b-1}/2$, and go to step 2.

For progressive transmission, the sorting pass followed by the refinement pass is repeatedly performed. To reconstruct the image, the received code stream is first parsed and the wavelet packet coefficients are rearranged back to the conventional format [13]. Finally, the decoded image can be obtained by inverse wavelet packet transform.

IV. Experimental Results

The proposed SPIAWPT algorithm has been evaluated on grayscale images. The coding results of images namely Barbara, San Francisco, and Fingerprint are presented in this paper. Figure 3 shows the original images together with subband decompositions obtained by using the proposed quad-tree based adaptive wavelet packet transform; the threshold T_r is 2. The number of decomposition levels L is 5. Bi-orthogonal 9/7 wavelet is used. The compression rate is measured in bits per pixel (bpp). The peak signal to noise ratio (PSNR) is measured in dB. The bit rates and PSNR values are taken to form the rate distortion curves for comparisons. The coding performance of SPIAWPT is compared to SPIHT.

There is a large portion of high details in Barbara image; thus, many middle and high frequency wavelet coefficients become significant in the early passes of SPIHT. As shown in Figure 4, the proposed SPIAWPT outperforms SPIHT at the low bit rates. For aerial images with textures such as San Francisco, SPIAWPT is preferable to SPIHT as shown in Figure 5.

Compression of fingerprints images is one of the most important issues, which demands the best solution. As one can expect, further decompositions of wavelet subbands would be desirable. It is noted that a significant improvement can be obtained by using SPIAWPT in terms of the rate distortion curves shown in Figure 6. Figure 7 shows the visual quality comparison between SPIHT and SPIAWPT.

V. Conclusion

Wavelet transform has been adopted by JPEG2000 as the underlying method to decompose an image into multiresolution subbands with orientation selectivity. In the well known SPIHT algorithm, the constructed wavelet trees of an image can be efficiently coded by exploiting the self similarity across wavelet subbands. However, for texture-rich images, many middle-high frequency wavelet coefficients are likely to become significant after a few coding passes of SPIHT. Thus, it is desirable that each individual wavelet subband can be represented by using different wavelet packet basis functions. To incorporate with the framework of SPIHT, a quad-tree based adaptive wavelet packet transform has been proposed to construct adaptive wavelet packet trees, which can be efficiently coded using SPIHT at no extra cost of coding complexity. Experimental results show that the proposed set partitioning in adaptive wavelet packet trees (SPIAWPT) is superior to set partitioning in wavelet trees, especially for coding fingerprints images, which is one of the most demanding tasks.

Acknowledgment

The National Science Council of Taiwan, under Grant NSC 96-2221-E-239-036 supported this work.

References

[1] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image Coding Using Wavelet Transform," IEEE Trans. On Image Processing, vol. 1, pp. 205-220, 1992.
[2] J. M. Shapiro, "Embedded Image Coding Using Zero-Trees of Wavelet Coefficients," IEEE Trans. On Signal Processing, vol. 40, pp. 3445-3462, 1993.

[3] A. Said and W. A. Pearlman, “A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees,” IEEE Trans. On Circuits Syst. Video Tech. vol. 6, pp. 243-250, 1996.

[4] D. Mukherjee and S. K. Mitra, “Vector SPIHT for Embedded Wavelet Video and Image Coding,” IEEE Trans. On Circuits Syst. Video Tech. vol. 13, pp. 231-246, March, 2003.

[5] A. Said and W. A. Pearlman, “Low Complexity Waveform Coding via Alphabet and Sample-Set Partitioning,” Proc. SPIE Visual Communications and Image Processing, vol. 3024, pp. 25-37, Feb., 1997.

[6] J. Andrew, “A Simple and Efficient Hierarchical Image Coder,” Proc. IEEE Int. Conf. Image Processing (ICIP), vol. 3, pp. 658-661, Oct., 1997.

[7] W. A. Pearlman, A. Islam, N. Nagaraj, and A. Said, “Efficient, Low Complexity Image Coding With a Set-Partitioning Embedded Block Coder,” IEEE Trans. On Circuits Syst. Video Tech. vol. 14, pp. 1219-1235, Nov., 2004.

[8] D. Taubman, “High Performance Scalable Image Compression with EBCOT,” IEEE Trans. On Image Processing, vol. 9, pp. 1158-1170, July, 2000.

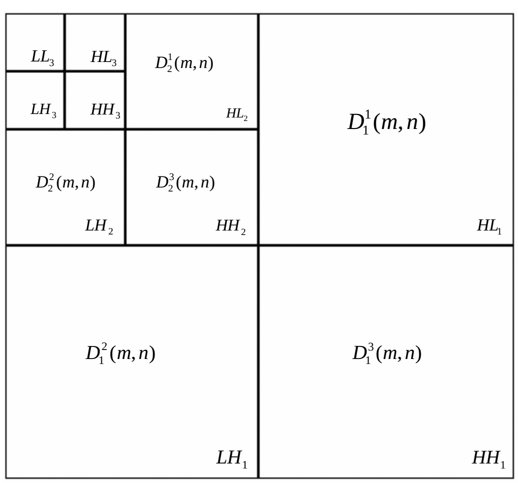
[9] A. Skodras, C. Christopoulos, and T. Ebrahimi, “The JPEG 2000 still image compression standard,” IEEE Signal Process. Mag., vol. 18, pp. 36-58, September, 2001.

[10] H.-C. Fang, Y.-W. Chang, T.-C. Wang, C.-T. Huang, and L.-G. Chen, “High-Performance JPEG 2000 Encoder with Rate-Distortion Optimization,” IEEE Trans. On Multimedia, vol. 8, no. 4, pp. 645-653, August. 2006.

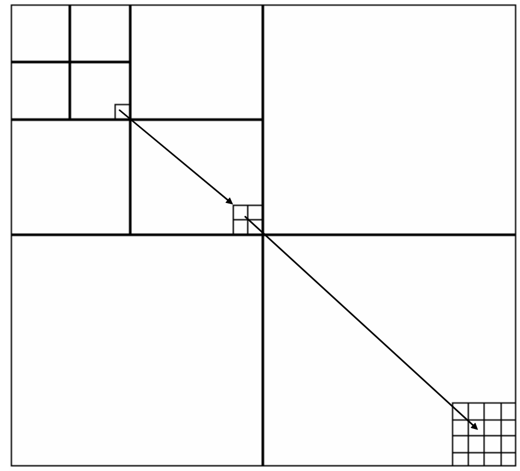
[11] D. Engle, A. Uhl, “Adaptive Object Based Image Compression Using Wavelet Packets,” VIPromCom-2002, 4th EURASIP, IEEE Region 8 International Symposium on Video/Image Processing and Multimedia Communications, pp. 183-187, 2002.

[12] N. M. Rajpoot, R. G. Wilson, F. G. Meyer and R. R. Coifman, “Adaptive Wavelet Packet Basis Selection for Zerotree Image Coding,” IEEE Trans. On Image Processing, Vol. 12, pp. 1460-1472, 2003.

[13] T.-Y. Sung and H.-C. Hsin, “An Efficient Rearrangement of Wavelet Packet Coefficients for Embedded Image Coding Based on SPIHT Algorithm,” IEICE Trans. Fundamentals, vol. E90-A, no. 9, pp. 2014-2020, 2007.

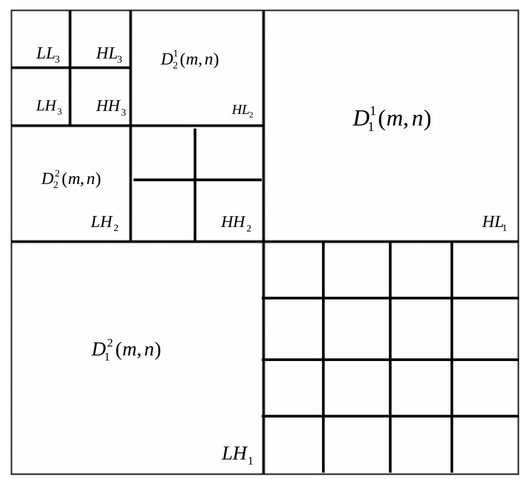


(a)

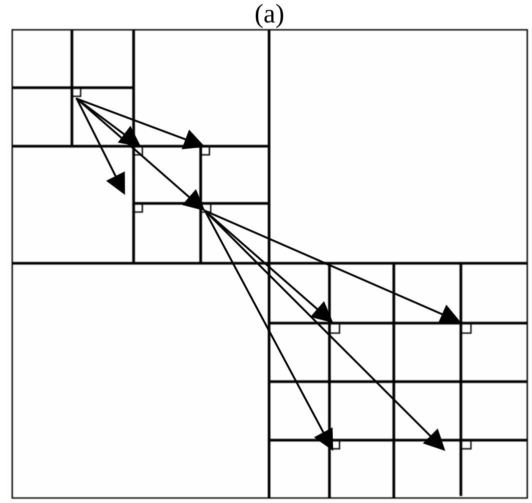


(b)

Fig. 1 (a) 3-level WT, (b) a hierarchical wavelet tree in the diagonal orientation.



(a)



(b)

Fig. 2 Example of WPT, (a) a 3-level WPT with subbands delimited by thick lines, (b) a hierarchical wavelet packet tree in the diagonal orientation.

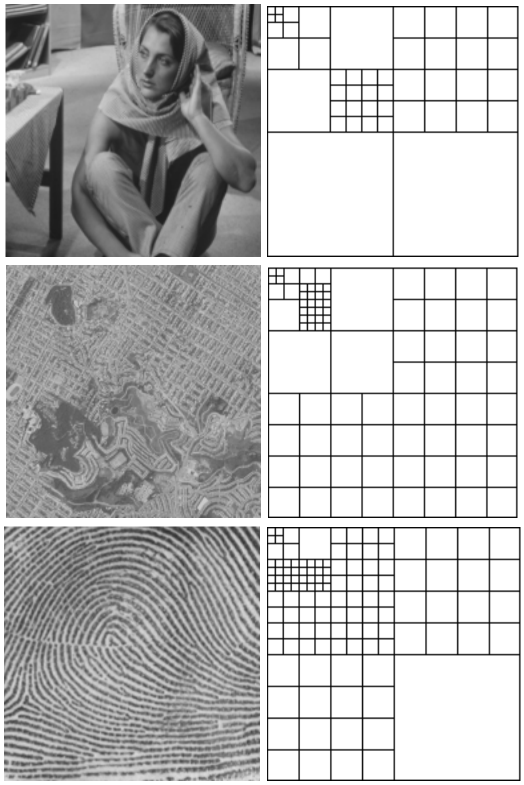


Fig. 3 Test images (left column) together with the subband decompositions obtained by using the proposed quad-tree based adaptive wavelet packet transform (right column), from top to bottom: Barbara (512 x 512), San Francisco (512 x 512), and Fingerprint (256 x 256) resized for demonstration.

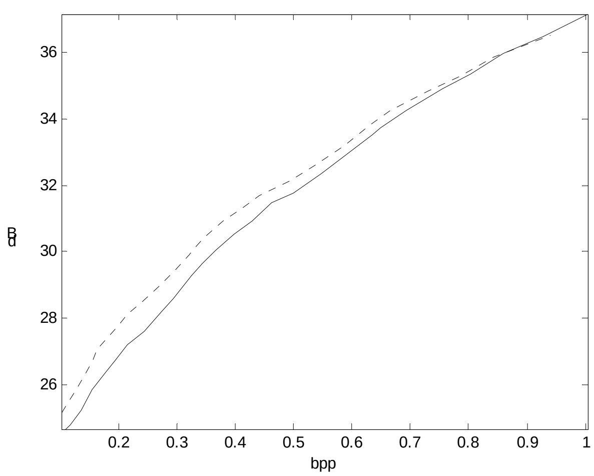


Fig. 4 Rate distortion curves of Barbara image obtained by SPIHT (solid line) and the proposed SPIAWPT (dotted line).

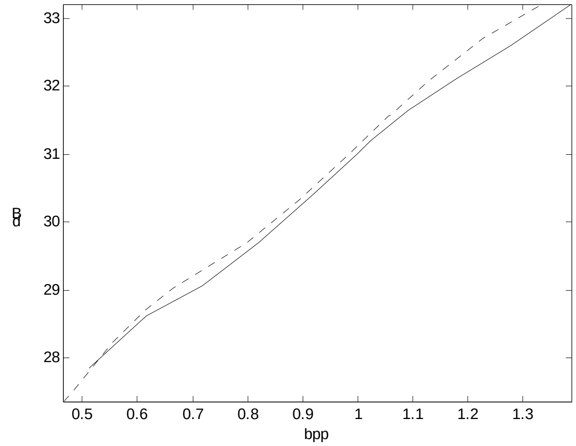


Fig. 5 Rate distortion curves of San Francisco image obtained by SPIHT (solid line) and the proposed SPIAWPT (dotted line).

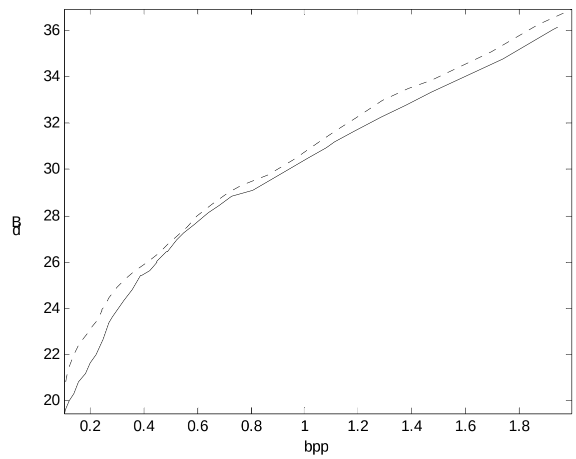
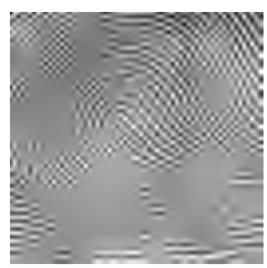
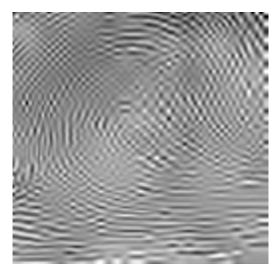
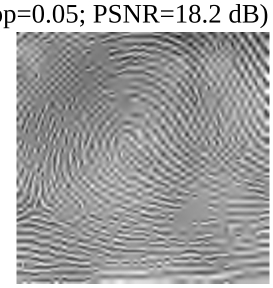
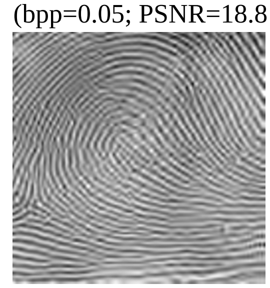


Fig. 6 Rate distortion curves of Fingerprint image obtained by SPIHT (solid line) and the proposed SPIAWPT (dotted line).

(bpp=0.05; PSNR=18.2 dB) (bpp=0.05; PSNR=18.8 dB)

(bpp=0.1; PSNR=19.7 dB) (bpp=0.1; PSNR=20.7 dB)

Fig. 7 Visual quality of the decoded Fingerprint images obtained by SPIHT (left column) and the proposed SPIAWPT (right column).