14th European Signal Processing Conference (EUSIPCO 2006), Florence, Italy, September 4-8, 2006, copyright by EURASIP

DCT-BASED IMAGE COMPRESSION USING WAVELET-BASED ALGORITHM WITH EFFICIENT DEBLOCKING FILTER

Wen-Chien Yan and Yen-Yu Chen
Department of Information Management, Chung Chou Institution of Technology
6, Line 2, Sec 3, Shan-Chiao Rd.,Yuanlin Changhwa. 51003, Taiwan R.O.C
Telephone: 886922638077, Fax: 88648369574, Email: miscyy@tcts1.seed.net.tw

ABSTRACT

This work adopts DCT and modifies the SPIHT algorithm to encode DCT coefficients. The algorithm represents the DCT coefficients to concentrate signal energy and proposes combination and dictator to eliminate the correlation in the same level subband for encoding the DCT-based images. The proposed algorithm also provides the deblocking function in low bit rate in order to improve the perceptual quality. This work contribution is that the coding complexity of the proposed algorithm for DCT coefficients is just close to JPEG but the performance is higher than JPEG2000. Experimental results indicate that the proposed technique improves the quality of the reconstructed image in terms of both PSNR and the perceptual results close to JPEG2000 at the same bit rate.

1. INTRODUCTION

Transform coding decomposes signal from spatial domain to other space using a well-known transform and encode these coefficients in new domain. Transform coding has higher compression performance than predictive coding in general, but requires more computation. The transform coding is an efficient compression method and the transforms include Karhunen-Loeve transform (KLT)[1], Discrete Cosine Transform (DCT)[2], Discrete Wavelet Transform (DWT)[3], Complex Wavelet Transform (CWT)[4] etc. KLT is the most optimal block based transform for data compression in a statistical sense because it optimally decorrelates an image signal in the transform domain by packing the most information in a few coefficients and minimizes the mean square error between the reconstructed and original image compared to any other transform.

The performance of DCT is very much near to the statistically optimal KLT because of its nice decorrelation and energy compaction properties. Moreover, as compared to KLT, DCT is data independent and many fast algorithms exist for its fast calculation so it is extensively used in multimedia compression standards.

In image compression, DWT based schemes have outperformed other coding schemes like the ones based on DCT. Since there is no need to divide the input image into non-overlapping 2-D blocks and its basis functions have variable length, wavelet-coding schemes at higher compression ratios avoid blocking artifacts. Because of their inherent multi-resolution nature, wavelet-coding schemes are especially suitable for applications where scalability and tolerable degradation are important. New image coding standard, JPEG-2000[5], which has been based upon DWT. And the DWT is the popular one and many excellent algorithms, such as EZW [6], SPIHT [7], EBCOT [8], GTW [9], embedded image compression based on wavelet pixel classification and sorting [10] etc, are proposed to encode the transform coefficients. But the disadvantages of DWT include the cost computing of DWT as compared to DCT are higher, the use of larger DWT basis filters produces blurring and ringing effect near edge regions in images, and compression time with DWT based schedule decomposed time longer than DCT based one.

2. PROPOSED DEBLOCKING METHOD

These algorithms for DWT coefficients, such as EZW, SPIHT, GTW, etc, are excellent in image compression, but the computing complexity of DWT is a common defect in these algorithms. Therefore, this algorithm adopts DCT and MSPIHT algorithm that designed for encoding the DWT coefficients [11]. The algorithm that was called DCT-based-MSPIHT was modified to suit it to encode DCT coefficients. Finally, the proposed algorithm reduces the blocking effects in low bit rate. Part 2.1 is DCT-based-MSPIHT and Part 2.2 describes the deblocking function.

2.1 DCT-based-MSPIHT

The compression performance of MSPIHT is more exceptional than of the original SPIHT, but it was proposed to encode the DWT-based image. Therefore, we modified the MSPIHT algorithm that use the combined function and dictator function to reduce the relationship between the same level subbands, the MSPIHT algorithm flowchart are shown in Figure 1. First, we decompose the image into ten subbands. Second, the significant DWT coefficients are sorted and this pass is suited for progressive transmission. Third, the combined function and dictator function reduce the redundancy between the same level subbands. The refinement pass evaluates the reconstructive value. The classification of the significant quadtree is good to get exceptional entropy coding. Finally, we transmit the bitstream.

The MSPIHT compression performance is exceptional than the original SPIHT's, but that encodes DWT coefficients not DCT coefficients. Therefore, we represent the DCT coefficients into subbands and use the MSPIHT to encode the represented DCT coefficients.

For block-based DCT coding, an input image is first partitioned into n x n blocks, where $n = 2^L$, $L > 2$. Each block is then transformed into the DCT domain and can be taken as an L-scale tree of coefficients with 3 x L + 1 subbands decomposition. After that, the same subbands for all DCT blocks are grouped and put onto their corresponding positions. We represent this reorganization of DCT coefficients into a single DCT clustering entity. Figure 2 demonstrates an 8 x 8 DCT block taken as three-scale tree structure with ten-subband decomposition. The reorganization of 8x8 DCT with ten-subband decomposition is illustrated in Figure 3. In Figure 3, Go0 denotes Group of subband 0 and Go*N* denotes Group of subband *N*.

Following features can be seen. Signal energy is compacted mostly into dc coefficients and small numbers of ac coefficients are related to the edges in spatial domain. Similarity between cross subbands and magnitude decay across subbands can be observed. The significant coefficients within subbands tend to be more clustered.

The above DCT characteristics after reorganization can be further utilized to DCT-based coders in order to obtain better compression performance as JPEG2000 or SPIHT did in the wavelet domain.

2.2 Deblocking

A low computational deblocking filter with four modes is proposed, including three frequency-related modes (smooth modes, intermediate mode, and complex mode for low-frequency, mid-frequency, and high-frequency regions, respectively) and one special mode (steep mode for a large offset between two blocks). A mode decision procedure is also needed to decide which mode is given by observing pixel behavior around the block boundary. To take the masking effect of HVS into consideration, the filter for smooth mode is designed much stronger than that for complex mode, because the human eyes are more sensitive to smooth regions.

Because most of the blocking artifacts occur on 8 x 8 block boundaries, the filtering should make its efforts on pixels around the block boundaries. Figure 4 shows how the pixel vectors are extracted form the block boundaries horizontally and vertically. The pixel vector is filtered one by one. The updated pixels are retained for next filtering.

The flowchart of the proposed algorithm is shown in Figure 5. The blocking artifact is caused by the discontinuity between V_3 and V_4. For simplicity, we use word *offset* to represent the difference of V_3 and V_4. If *offset* is larger than a threshold *'edge_Thre'*, this pixel vector is skipped because it may contain a real edge. Otherwise, the pixel vector is passed to the mode decision stage to decide which mode it belongs to. After that, a suitable filter for each mode is used to remove the blockiness. The proposed algorithm is composed of mode decision stage part and filtering stage.

The purpose of the mode decision is to determine the mode of filtering required to alleviate the blocking effect without excessively blurring the local feature. It is achieved by calculating the variation in the pixel vector and the calculating formulation is listed as follow:

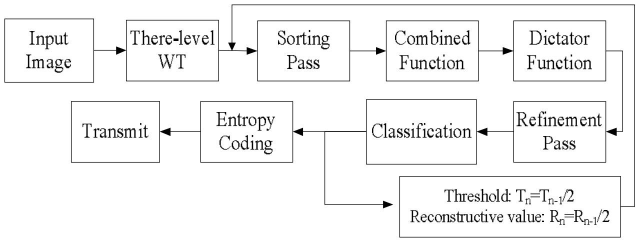

Figure 1 - MSPIHT algorithm.

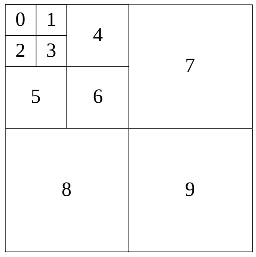

Figure 2 -8 x 8 DCT block taken as three-scale tree with ten-subband decomposition.

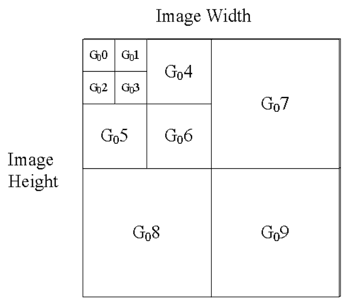

Figure 3 - Reorganization of 8 x 8DCT blocks into a single DCT clustering entity.

$$'activity' = \sum_{i=1}^{5} \Phi(v_i - v_{i+1}) \qquad (1)$$

where, $$\Phi(\Delta) = \begin{cases} 0, & if\ |\Delta| \le S \\ 1, & otherwise \end{cases} \qquad (2)$$

The *activity* value represents the variation of the pixel vector. That is, a small *'activity'* indicates a smooth region, whereas a high *'activity'* indicates a region with edge detail.

The *'activity'* value is compared with two thresholds, T_{low} and T_{high}, to determine the appropriate filtering mode. T_{low} is set to suitable small value to ensure that the region is sufficiently smooth to apply smooth mode filtering when *'activity'* < T_{low}. Similarly, the use of complex mode filtering is specified if *'activity'* > T_{high} in areas of high spatial complexity. If *'activity'* is between T_{low} and T_{high}, intermediate filtering is adopted to improve visual quality. For the above three cases, if the absolute value of *offset* is too large, for example, when the pixel vector belongs to smooth mode and the *offset* is lager than *2T*, steep mode is designed for this situation.

There are total four filter modes in the filtering stage: smooth mode, intermediate mode, complex mode and steep mode. Excepting steep mode, the other modes are corresponding to three different frequency bands that are low-frequency, mid-frequency and high-frequency. Implementation of these filters is discussed below.

2.2.1 Filter for Smooth Mode

The filter for smooth mode is shown as following step:

1. Calculate the difference between two blocks:

offset = V4 – V3;

2. Update V1, V2, V3, V4, V5, V6:

V1' = V1 + offset / 8;
V2' = V2 + offset / 4;
V3'= V3 + offset / 2;
V4' = V4 - offset / 2;
V5' = V5 - offset / 4;
V6' = V6 - offset / 8;

3. Adjust all of these updated pixels value within 0 to 255

*Clip(0, 255, **V1'**) Clip(0, 255, **V2'**) Clip(0, 255, **V3'**)*
*Clip(0, 255, **V4'**) Clip(0, 255, **V5'**) Clip(0, 255, **V6'**)*

Where Clip(a, b, c) is a function that clips *c* into the range from *a* to *b*.

It can be seen that the updated pixels not only contain two pixels across block boundary (V_3 and V_4) but also adjacent pixels (V_1, V_2, V_5 and V_6). The filtered pixel vector no longer has sharp discontinuity around block boundary.

2.2.2 Filter for Complex Mode

In complex mode, gray values of pixel vector are oscillated and human eyes are not sensitive to this region. If the blocking effects occur in a high activity region, strong filtering is not appropriate because it over-blurs the true edge also imposes an unnecessary computational burden. Following the concept, the total length of the line of filtered pixels is limited to two. The filter for complex mode is discussed below:

1. Calculate the difference between two blocks:

offset = V4 – V3;

2. Update V3 to V3':

if | V2 – V3 | < T then
V3' = (V2 + 2 × V3 + V4) / 4;
else
V3' = V3 + offset / 4;
end

3. Update V4 to V4':

if | V4 – V5 | < T then
V4' = (V3 + 2 × V4 + V5) / 4;
else
V4' = V4 - offset / 4;
End

4. Adjust all of these updated pixels value within 0 to 255

*Clip(0, 255, **V3'**) Clip(0, 255, **V4'**)*

Although the updated pixels include only two pixels across block boundary (V_3 and V_4), the adjacent pixels (V_2 and V_5) can be used as reference for more accurate reconstruction of image. If adjacent pixels are referable (difference of grayscale value not exceeding a threshold, T), a 3-tap low-pass filter can be used to smooth this region. Otherwise,the filter will alleviate the blockiness by shortening the distance between V_3 and V_4.

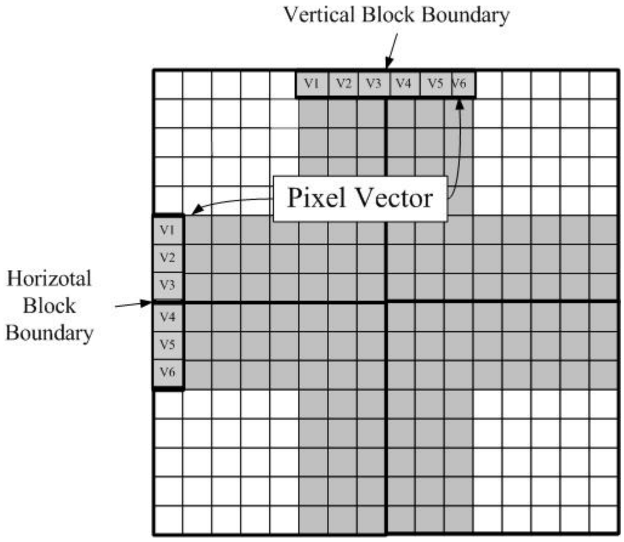

Figure 4 - Pixel vectors across block boundaries.

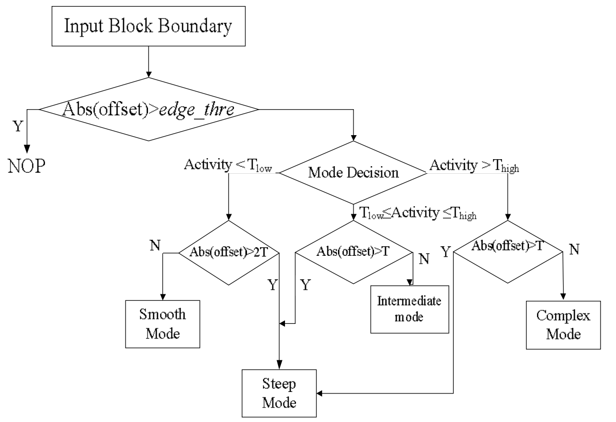

Figure 5 - Flowchart of the proposed deblocking algorithm.

2.2.3 Filter for Intermediate Mode

Filtering for intermediate mode is shown as follow:

1. Calculate the difference between two blocks:

offset = V4 – V3;

2. Update V3 to V3'

if | V2 – V3 | < T then
V3' = (V2 + 2 × V3 + V4) / 4;
else
V3' = V3 + offset / 2;
end

3. Update V4 to V4'

if | V4 – V5 | < T then
V4' = (V3 + 2 × V4 + V5) / 4;
else
V4' = V4 - offset / 2;
End

4. Adjust all of these updated pixels value within 0 to 255

*Clip(0, 255, **V3'**)   Clip(0, 255, **V4'**)*

Both complex mode and intermediate mode use the same 3-tap low-pass filter when adjacent pixels are referable. If adjacent pixels are not referable, intermediate mode update pixels by *offset/2* while complex mode update pixels by *offset/4*. This is because that in complex mode, it is reasonable to preserve high-frequency characteristic to prevent over-smoothing.

2.2.4 *Filter for Steep Mode*

The steep mode is a special filter that is used to not only improve visual quality but also remove blocking artifact. In many cases, an edge happened on block boundaries may be enhanced because of blocking artifact. The enhanced edges sometimes make human eyes uncomfortable. Filtering for steep mode is shown as follow:

1.Calculate the difference between two blocks:

offset = V4 – V3;

2.Update V3 and V4 by means of the offset:

V3' = V3 + offset / 4;

V4' = V4 - offset / 4;

3. Adjust all of these updated pixels value within 0 to 255

*Clip(0, 255, **V3'**)   Clip(0, 255, **V4'**)*

Figure 6(a) presents the complete block diagram of the encoder for compressing still images. First, the test image is transformed by 8 x 8 block-DCT and represented into subband distribution image. Then the proposed algorithm deals with subband LL_3 in a sorting pass that is the same as that of the MSPIHT algorithm. The results of sorting pass include the bitmap and the sign information of significant coefficients. LH_3, HL_3, and HH_3 subbands are handled by a combined function that reduces the interband redundancy, and the results include information indicates which subband has significant coefficients. The proposed algorithm deals with the other subbands using a dictator function and the results decide which should be sent. The proposed algorithm also uses the refinement pass and the results include the bits to correct the reconstructed value. Finally, entropy coding is used to improve performance and transmit the bit stream. The receiver decodes the bit stream and reverses the entropy coding. And then the receiver reconstructs the bit plane and adjusts the reconstructive DCT values by exact pass. When all of the bit streams are decoded, the reconstructive DCT values are represented and the decoder reverses the reconstructive DCT values to reconstructive spatial pixel values. Figure 6(b) presents the complete block diagram of the decoder with deblocking algorithm.

The deblocking function based on four filtering modes is proposed. These four filtering modes include three frequency-related filters (low-frequency, mid-frequency, and high-frequency) and one special filter. By observing the characteristic across block boundaries, suitable filter is used to remove the blocking artifact. After applying filters on the block boundaries, the block artifact is removed without over-smoothing the image details. The deblcoking function is better to contain no multiplication and division instruction. All the instructions can be used are addition, subtraction, comparison and shift operations. Finally, the computation must be low enough for real time applications.

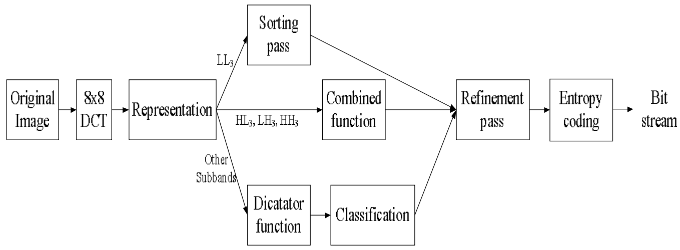

Figure 6(a) - Flowchart of the DCT-based-MSPIHT encoder

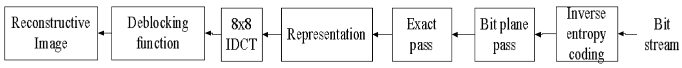

Figure 6(b) - Flowchart of the DCT-based-MSPIHT decoder with the deblocking function

3. SIMULATION RESULTS

The performance is evaluated by PSNR (peak signal to noise ratio) value. PSNR value is a mathematics evaluation expression that can be calculated as

$$PSNR = 10\log_{10}\frac{255^2}{\frac{1}{T}\sum_{i=0}^{n-1}\sum_{j=0}^{n-1}\left(x_{i,j}-x'_{i,j}\right)^2} \quad (3)$$

The DCT-based-MSPIHT is compared with JPEG2000 and JPEG. Figure 7(a) shows a part of size 500x480 in woman original image. The complexity of the test image context is low. Therefore the quality of test image compressed in the middle bit rate is acceptable by visual observation. Figure Figure 7(b) shows the test image decoded by DCT-based-MSPIHT algorithm without deblocking function at a bit rate of 0.28 bpp with a PSNR of 30.96 dB and there is few blocking effect in the image because 8 x 8 DCT transform is used. And Figure 7(c) shows that the test image decoded by DCT-based-MSPIHT algorithm with deblocking function at a bit rate of 0.28 bpp with a PSNR of 30.34 dB. In Figures 7 (b) and (c), it shows that the PSNR value is not absolutely objective to determine the visual quality. Table 1 compares the PSNR values at various bit rates obtained using JPEG, JPEG 2000 and the DCT-based-MSPIHT algorithm with/without deblocking function. The values in ( ) are the PSNR of the reconstructive image decoded by DCT-based-MSPIHT algorithm without deblocking function. And the visual quality of the DCT-based-MSPIHT algorithm with deblocking is obviously better than JPEG and JPEG 2000.

4. CONCLUSIONS

This work proposed a novel algorithm for encoding of the still image and reducing of blocking artifact in transform coded images. The challenges posed by imaging involve the development of a compression algorithm that has a computational complexity near that of the DCT, yet supports the DWT-based high-compression ratios to reduce storage, transmission, and processing.

This method is to represent the DCT coefficients similarity to the subband coefficients and then encoding the DCT coefficients by DCT-based-MSPIHT. This algorithm utilizes combination and dictator to eliminate the correlation in the same level subband for encoding the DCT-based images. The coding complexity of the proposed algorithm for DCT coefficients is just close to JPEG but the performance is higher than JPEG2000.

The proposed deblocking algorithm is based on the 1-D filtering of block boundaries. Observations of local image characteristics across block boundaries show that using four filtering modes to promote effective reducing blocking effect. The filtering in smooth regions is sufficiently strong to eliminate noticeable blocking effects. In complex regions, preserving the details of an image is desirable while performing the desired deblocking. An intermediate filtering mode is proposed to balance in smooth regions and weak filtering in complex regions. The steep mode is used to remove that an edge happened on block boundaries may be enhanced because of blocking artifact. Also, due to its low complexity, the algorithm is believed to real time processing and hardware implementation.

Objective and subjective testing have established the favorable results of the proposed technique. The proposed algorithm is close to JPEG2000 in terms of both PSNR and the perceptual results at the same bit rate.

5. ACKNOWLEDGEMENTS

The author would like to thank the National Science Council of the Republic of China, Taiwan, for financially supporting this research under Contract No. NSC94-2213-E-235-001-

REFERENCES

[1] Dony, R.D., Haykin, S, "Optimally adaptive transform coding**,"** *IEEE Transactions on Image Processing*, vol. 4 no. 10 pp. 1358-1370, Oct. 1995

[2] N. Ahmed, T. Natarajan and K. R. Rao, "Discrete cosine transform," IEEE Transations Computer.," vol. C-23, pp. 90-93, Jan. 1974

[3] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Transaction on Image Processing*," vol. 1, no. 2, pp. 205-220, Apr. 1992

[4] N. G. Kingsbury, "Image processing with complex wavelets," Philos. Trans. R. Soc. London A, vol. 357, pp. 2543–2560, Sep. 1999

[5] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG2000 still image compression standard," IEEE Signal Processing Mag., vol. 18, no. 5, pp. 36–58, Sept. 2001

[6] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Transaction on 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 Transactions Circuits and System for Video Technology, vol. 7,no. 3, pp. 243-250, June 1996

[8] D. Taubman, "High performance scalable image com pression with EBCOT," IEEE Transactions on Image

Figure 7(a)

Figure 7 (b) Figure 7 (c)

Bit-rate (bpp)	JPEG (dB)	JPEG2000 (dB)	DCT-based-MSPIHT (dB)
0.03		24.17	23.71 (23.48)
0.12	24.91	27.01	27.05 (26.97)
0.28	28.69	30.53	30.34 (30.96)
0.53	31.40	33.99	35.30
0.94	34.60	37.97	39.58
1.60	38.75	42.03	44.55

Table 1. PSNR values for JPEG, JPEG 2000 and DCT-based-MSPIHT algorithm at various bit rates in woman test image

Processing, vol. 9, no. 7, pp. 1158–1170, 2000

[9]Edwin S. Hong and Richard E. Ladner, "Group Testing for Image Compression," *IEEE Transactions On image processing*, vol. 11, no. 8, pp. 901-911, Aug. 2002

[10] Kewu Peng; Kieffer, J.C "Embedded image compression based on wavelet pixel classification and sorting," *IEEE Transactions on Image Processing*, Vol. 13, no. 8, pp.1011 – 1017, Aug. 2004

[11] Tai, S.C, Chen, Y. Y., and Yan, W. C, "New, High Fidelity Medical Image Compression Based on Modified SPIHT", *Optical Engineering*, vol. 42, no. 7, pp. 1956-1963, Jul. 2003