2012 International Conference on Information and Computer Networks (ICICN 2012)
IPCSIT vol. 27 (2012) © (2012) IACSIT Press, Singapore

An Improved Method for the Transmission of Large Image over Internet

Kamrul Hasan Talukder and Koichi Harada

Graduate School of Engineering, Hiroshima University, Japan.
{kht,hrd}@hiroshima-u.ac.jp

Abstract. We have proposed a wavelet-based image transmission scheme in this paper which has used discrete wavelet transform to transform a digital image from spatial domain into frequency domain. Haar wavelet transformation has been chosen and used for wavelet transformation phase. Use of concurrent computing has significantly reduced computation time overhead as well as the transmission time up to a great extent as compared to the existing sequential discrete wavelet based transmission. The experimental result shows that the proposed scheme maintains the accuracy of reconstructed image.

Keywords: Wavelet Transform, Concurrent computing, Progressive image transformation, Haar wavelet transformation.

1. Introduction

The browsing image on the internet has become a common activity as a means of communication with the emergence of World Wide Web. However, it often becomes frustrating due to low speed internet connection. As there are several phases like encoding and decoding of image while transmitting over the internet, it takes much time to transmit the whole of a large sized image. Fast downloading of high quality images is of increasing importance in many applications including photo agencies, geographical information systems, medical databases, distance learning etc. So, requiring long time to complete the whole image transmission process is a big problem. Sometimes, the image presented is not the actual one that the user wants. So, it is necessary to allow the user to have a preview before the whole transmission is completed so that he is allowed to save time by aborting the transmission if required.

Here a better technique has been proposed to send the core portion of the original image and then refining it progressively. The technique divides the original image into several parts. The sender transmits the image to the receiver via different stages, and the receiver has to combine the data from all stages to recover the image from initially blurred to progressively clear. Thus, using this method, user can get a glance of the image earlier and can decide whether it is the correct image or abort it. In case of large sized images, the encoding and decoding processes for transmitting image also takes mentionable time, because the traditional way followed for these are computationally rigorous.

The target of our method is to encode the original image into a data stream from which image can be reconstructed efficiently. Moreover, it provides a fast glance of the image to the user. The encoding and decoding processes have been manipulated in such a way that it fulfills the criteria. This paper has proposed a scheme where concurrent computing has been used in both encoding and decoding phase to reduce the computation time. Also concurrent technique is applied during the data stream transmission. Though concurrent computing technique has been applied in both phases the quality of reconstructed image has been ensured.

2. Haar Wavelet Technique

The Haar wavelet is the simplest type of wavelet. In discrete form, Haar wavelets are related to a mathematical operation called the Haar transform. The Haar wavelet's mother wavelet function $\psi(t)$ can be described as-

$$\psi(t) = \begin{cases} 1 & 0 \leqslant t < \frac{1}{2}, \\ -1 & \frac{1}{2} \leq t < 1, \\ 0 & \text{otherwise} \end{cases}$$

and its scaling function $\varphi(t)$ can be described as

$$\varphi(t) = \begin{cases} 1 & 0 \leq t < 1, \\ 0 & \text{otherwise} \end{cases}$$

In case of image transformation there exist three types of detail images for each resolution: horizontal (HL), vertical (LH) and diagonal (HH). The operations can be repeated on the low low (LL) band using the second stage of identical filter bank. Thus, a typical 2D DWT generates the hierarchical structure shown in Figure 1.

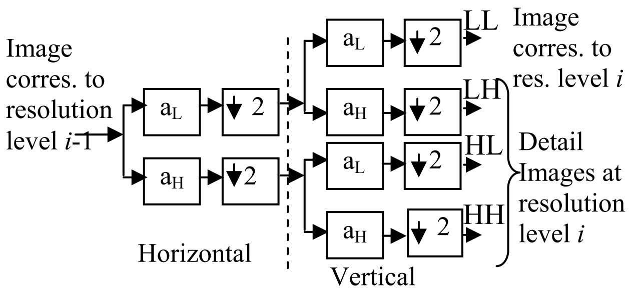


Figure 1. One Filter Stage in 2D Discrete Wavelet Transform

While the 2D Haar wavelet transformation is performed, the image is comprised of pixels represented by numbers. Then the operation of averaging and differencing is performed to arrive at a new matrix representing the same image in a more concise manner.

3. Related Works

Progressive image transmission (PIT) has been studied by many researchers and several schemes have been proposed. Two sub-bands refining method is based on wavelet. This PIT method consists of three steps: decomposition, transmission and reconstruction. In the decomposition step, the scheme performs Haar DWT on the image. Then the wavelet coefficients are represented by sub-bands [2].

In the transmission step, the sender transmits the sub-band along with some selected coefficients in the next sub-band to the receiver for image reconstruction. If the coefficients in the sub-band have been submitted, then the scheme transmits all the coefficients in the next sub-band. A concurrent scheme [3] for wavelet transform of image was proposed earlier. There the image plane is divided into n horizontal sections which are horizontally transformed concurrently. After then the image is divided into n vertical sections which are then vertically transformed concurrently. This system allows beginning of vertical transformation on some vertical sections before horizontal transformation in all sections is completed. Vertical sections that are already horizontally transformed can be vertically transformed.

----							1
----							2
----							3
----							…
----							…
----							…
----							…
----							n

Figure 2: Ordering the transformation

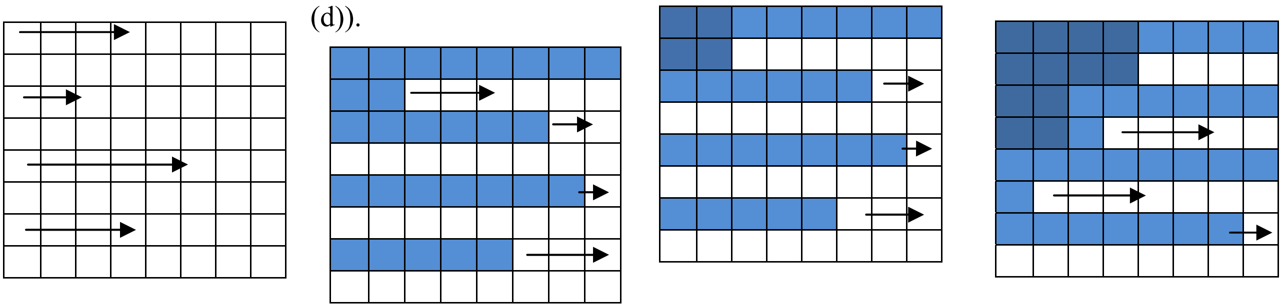
Here it is possible for threads that completed horizontal transformation to go on to vertical transformation without having to wait on other threads to complete horizontal transformation. The gray color in Figure 2 indicates sections of image data that are horizontally transformed. The white color indicates sections of image data that are not yet horizontally transformed. The gray vertical section with dotted spots can be assigned to a thread for vertical transformation.

4. Proposed Method

This part demonstrates how the concurrent computing technique works over Discrete Wavelet Transformation (DWT). There are several algorithms for wavelet based compression such as Embedded Zerotree Wavelet (EZW), Set Partitioning in Hierarchical Trees (SPHIT), Wavelet Difference Reduction (WDR), Adaptively Scanned Wavelet Difference Reduction (ASWDR) etc. However, here we have considered the very basic technique called Haar Discrete Wavelet Transformation for color image transformation.

First of all, the image matrix is mapped from the digital image. The horizontal transformation threads are initialized for horizontal transformation. The separate process for every pair of rows of the matrix is used with a little change with the algorithm of traditional Haar wavelet transform and thus the vertical transformation process is also embedded in the row transformation process to speed up the computation process and to avoid complexity. Separate threads for transforming red, green and blue (RGB) components are used. Thus multiple threads are started at a time. The threads normally transform the rows following the discrete Haar wavelet transformation. But as soon as the pair of elements of the same column is transformed the column transformation for those two components is also done. In the concurrent transformation scheme proposed in [3] vertical transformation starts after first element of last row has horizontally transformed. With our new proposal it speeds up the transformation process as no waiting time required for the column transformation after row transformation has been finished. After completion of encoding process the matrix with detail coefficients and average coefficients is found. Then the required thresholding is applied on it and the final matrix is ready to transmit to the receiver. At the transmission phase we have transmitted the matrix by pair of columns. This is also done by several threads. The decoding process is done as the reverse of encoding. Thus the reconstruction of image requires comparatively lower time.

As illustrated in Figure 3(a), threads for horizontal transformation are started for each pair of rows at the same time. The region blue shaded region is horizontally transformed. In figure 3(c) horizontal transformation for the first two elements of the second row is completed. Then the vertical transformation is also done for available elements. The blue shaded region with dotted spots is vertically transformed (Figure 3 (d)).



(a) (b) (c) (d)

Fig. 3: Concurrent transformation a) Threads for each pair of row, b) Row transformation in progress, c) Column transformation done when pair of column elements are ready, d) Row and column transformation are in progress,

This method forms the output image with fractions as illustrated in figure 4. But as the fractions are parts of the original image without modification and it appears very fast the person can identify whether the image is the actual one or not within very short time.

(i) (ii) (iii) (iv) (v) (vi)

Figure 4. Reconstructed image after receiving pairs of column concurrently, PSNR values (i) 25.2837dB (ii)26.5822 dB (iii) 27.2201 dB (iv) 28.1388 dB (v) 30.6938dB (vi) Reconstructed image 40.71 dB

5. Experimental Result

In Table 1, the time required for one stage of encoding and decoding for different images for normal discrete wavelet transformation and concurrent computing are shown. Column A's represent the time required for concurrent wavelet transformation and column B's represent the time required for normal discrete wavelet transformation.

Table 1. Time comparison between normal discrete wavelet transformation and concurrent discrete wavelet transformation

	Encoding time (ms)		**Decoding time (ms)**	
Images	A	B	A	B
moon(1986x1986)	11	2173.1	5	2840
pepper(512x512)	0.9	144	0.3	234
lenna(512x512)	0.8	52	0.3	78
airplane(512x512)	2	143	0.7	216
baboon(480x480)	0.6	126.7	0.3	215

The data presented in Table 1 shows how much the encoding and decoding time has been reduced by applying concurrent computing with comparison to normal sequential computing. Though the transmission time may vary because of transmission lines, the time required will be reduced much as the two main phases requires far less time. From our experimental result we have found that the required time for encoding and decoding is up to 90% or more less than the sequential computing. The performance may vary on different machines but still it gives much better time overhead.

The required time to complete the whole process of image transmission for several images is shown in table 2. In the two sub band refining method image data is divided into sub bands and then transmitted. Here all the sub bands have to be transmitted one by one. Thus, the time required for completing the whole transmission process is as same as if the image data is transmitted in a sequential way. Thus, the transmission time required without applying concurrency can be considered as the time required for sub band refining method. Table 2 shows the extent of better performance found after applying concurrent computing rather than sequential computing.

Table 2. Comparison of required time of transmission

Images	**with concurrency(ms)**	**without concurrency(ms)**
airplane	120	677
moon	1644	9524
pepper	120	650
baboon	128	597
lenna	124	646

A computer with Intel Core 2 CPU (2 GHz) is used as client and another one with Intel Pentium 4(2.66 GHz) CPU is used as server to collect data. On average applying concurrent computing it requires 80% to 82% less time than the sequential discrete wavelet transformation. Also the quality of reconstructed image holds acceptable visual quality.

A commonly used measure for quantifying the error between images is the Peak Signal to noise Ratio (PSNR). Generally, when the PSNR is 40dB or larger, then the original image and the reconstructed image after transmission are virtually indistinguishable by human observers. Thus the PSNR of 37 dB or above is acceptable. In this case we've found that with our method, it provides good level of details to the output. Comparing with other wavelet-based PIT schemes [2] the output is acceptable. We have tested several images and found good quality of the reconstructed image compared with other PIT schemes. C.C. Chang has shown the PSNR [2] of 'lenna' found with his proposed approach is 33.03. With our approach this measure has found 40.71. In Table 3 we have shown PSNR values for different images.

Table 3. PSNR of reconstructed image

Images	PSNR of concurrency method	PSNR of Two sub-bands refine method
lenna	40.71	33.03
barbara	36.34	37.16
baboon	38.97	38.46
airplane	38.76	38.24
moon	39.74	38.76
pepper	40.06	38.58

6. Conclusion

An enhanced method for the progressive image transmission has been presented which uses concurrent computing on Discrete Wavelet Transformation to get the enhancement and thus reduces the image browsing time with a very few distortions in the reconstruction phase and within far less time than the traditional one.

7. References

[1] K.H. Talukder and K. Harada, "Haar Wavelet Based Approach for Image Compression and Quality Assessment of Compressed Image", IAENG International Journal of Applied Mathematics, 36:1, IJAM_36_1_9.

[2] C. C. Chang and T. C. Lu, "A Wavelet-based Progressive Digital Image Transmission Scheme", Proceedings of the First International Conference on Innovative Computing, Information and Control – Volume 2, 2006, pp. 681-684.

[3] K.H. Talukder and K. Harada, "A Scheme of Model Verification of the Concurrent Discrete Wavelet Transform (DWT) for Image Compression", International Journal of Signal Processing, Volume 5 Number 1.

[4] S. Saha, "Image Compression - from DCT to Wavelets : A Review" , The ACM Student Magazine, http://www.acm.org/crossroads/xrds63/sahaimgcoding.html

[5] K. H. Tzou, "Progressive Image Transmission: A Review and Comparison of Technique," Optical Engineering, Vol. 26, No. 7, 1987, pp. 581-589.

[6] R. C. Chang, T. K. Shih, "Innovative Decomposition for Progressive 1Image Transmission ", Asian Journal of Health and Information Sciences, Vol. 1, No. 2, pp. 204-227, 2006.

[7] K. H. Talukder and K. Harada, "A Scheme of Wavelet Based Compression of 2D Image", Proc. IMECS, Hong Kong, pp. 531-536, June 2006.

[8] G. K. Wallace, "The JPEG Still Picture Compression Standard, Comm. ", ACM, vol. 34, no. 4, April 1991, pp. 30-44.

[9] A. Wesley, B. Ari, Mordechai, "Principles of Concurrent and Distributed Programming (2nd ed.).", By ISBN 978-0-321-31283-9.

[10] G. Strang and T. Nguyen, "Wavelets and Filter Banks", Wellesley-Cambridge Press, Wellesley, MA, 1996, http://www-math.mit.edu/~gs/books/wfb.html.

Appendix

Algorithm: Encoding

Here pixel matrix is defined by 'mat'. The height of the image is 'height' and width of the image is 'width'. Component' defines the color component of the pixel.

```
segment initialize_thread(object reference){
    for offset = 1 to height
            interlock(reference) =true;
            thread_pool(offset,component) //calling thread
            offset:= offset+2;
    end loop
}
segment thread_pool(object  state, int component){
            float tmp[width],
            for i:= state to (state+2)
             int k=0, int mid=width/2,int midh=height/2;
             if i is even then
                    for j:=1 to width
                  tmp[k]:=(mat[i,j,component] +mat[i,j+1,component])/2;
                     tmp[mid]: =mat[i,j,component]-tmp[k];
                     j: =j+2;  k++; mid++;
                    for t: =1 to width
                       mat[i,t,component] =tmp[t];
            end if
            if  i is odd then
                    for j:=1 to width
                  tmp[k]:=(mat[i,j,component]+ mat[i,j+1,component])/2;
                      tmp[mid]:=mat[i,j,component]-tmp[k];
                       mat_1[(i-1)/2,k,component]=(mat[i-1,k,component] + tmp[k])/2;
                  mat_1[midh,kk,component]=mat[i-1,k,component]-mat_1[(i-1)/2,  k,component)];
                       mat_1[(i-1)/2,mid,component]=(mat[i-1,mid,component]+tmp[mid] )/2;
                       mat_1[midh,mid,component]=mat[i-1,mid,component]-mat_1[(i-1)/2, mid,component];
                     j: =j+2;  k++; mid++; midh++;
             end if
            end loop
interlock(reference) =false;
return to thread pool;
}
```