Bit-rate Scalable Video Coder Using a 2x2x2 DCT for Progressive Transmission

Seock Hoon Woo, Jin Hyung Park, Chee Sun Won

Dept. of Electronic Eng. Dongguk University, Seoul, Korea

E-mail : cswon@dgu.ac.kr

Abstract

In this paper, we propose a progressive transmission of a video using a 2x2x2 DCT. First of all, the video data is transformed into multiresolution represented video data using a 2x2x2 DCT. Then, it is represented by a 3-D EZT(Embedded ZeroTree) coding for the progressive transmission with a bit-rate scalability. The proposed progressive transmission algorithm needs much less computations and buffer memories than the higher-order convolution based wavelet filter. Also, since the 2x2x2 DCT requires independent local computations, parallel processing can be applied.

I. Introduction

Progressive video transmission is a technique to transmit a video shot on two or more stages. At the first stage, the sender (or the server) transmits a rough but recognizable video; at later stages, further transmissions of the detail of the video can be made on the request of the receiver (or the client). An application of the progressive video transmission is the internet shopping, where the client would request more information of merchandise, interactively to see the detail of the video shot for the merchandise. Or, if not interested, the client can terminate the transmission at any stage. Consequently, progressive video transmission hierarchically reorganizes the information in a video with the aim of increasing the efficiency of the transmission process. The primary advantage of progressive image transmission is that the rough structural information of the video shot appears immediately at the beginning of transmission, which makes it possible for the receiver to decide whether a further detail of the video is necessary or not [1].

Conventional progressive transmission researches have been conducted for still images only and, to the best of author's knowledge, the results of the progressive video transmission using the wavelet method have not been reported yet. In this paper, we propose to use a 2x2x2 DCT for the progressive video transmission. The proposed algorithm consists of two processing steps, namely multiresolution representation of video using a 2x2x2 DCT and the coding of the coefficients using an EZT(Embedded ZeroTree) [2]. Depending on the traffic of the internet or the request of the receiver, each level of the image data is transmitted subsequently for the progressive transmission. For the multiresolution representation of a video, we first divide a group of image frames with N_1xN_2 spatial resolution and F frames into non-overlapping 2x2x2 video blocks. Then, on each 2x2x2 video block, we take a 2x2x2 DCT. Then, we rearrange the 2x2x2 DCT coefficients of all video blocks such that the same frequency components gather together. On the DC components, we take a 2x2x2 DCT-rearrangement process recursively. Then, we obtain a 3-dimensional multiresolution representation of the 2x2x2 DCT coefficients. Finally, we apply an EZT on the multiresolution represented video data for the progressive transmission. The EZT represented data is transmitted with various bit-rates.

The proposed multiresolution representation algorithm needs much less computations and buffer memories than the higher-order convolution based wavelet filter. Also, since the 2x2x2 DCT requires independent local computations, parallel processing can be applied.

The organization of this paper is as follows; in section II, we introduce the three-dimensional multiresolution representation of video using a 2x2x2 DCT. Section III presents the three-dimensional ZeroTree Coding. Section IV presents the progressive video transmission algorithm with the bit-rate scalability. In Section V, we present the simulation results of this algorithm. Finally, the conclusion of this work follows in Section VI.

II. Three-Dimensional Multiresolution Representation of Video Using a 2x2x2 DCT

To represent multiresolution of a video, we first divide a group of image frames with N_1xN_2 spatial resolution and T frames into non-overlapping 2x2x2 video blocks. Then, on each 2x2x2 video block, using the Equations (1)-(8), we take a 2x2x2 DCT as shown in Figure 1-(a). That is, starting with the full image resolution (i.e., k=1), we can obtain the DCT coefficients with the (m,n,f)th block for $m=0,1,\cdots, N_1/2^k-1$, $n=0,1,\cdots, N_2/2^k-1$, $t=0,1,\cdots, T/2^k-1$, and $k=1,2,3$ as follows

$$F^k_{mnt}(0,0,0)=[f^k_{mnt}(0,0,0)+f^k_{mnt}(0,0,1)+f^k_{mnt}(0,1,0)+f^k_{mnt}(0,1,1) \\ +f^k_{mnt}(1,0,0)+f^k_{mnt}(1,0,1)+f^k_{mnt}(1,1,0)+f^k_{mnt}(1,1,1)]/8 \quad (1)$$

$$F^k_{mnt}(0,1,0)=[f^k_{mnt}(0,0,0)+f^k_{mnt}(0,0,1)-f^k_{mnt}(0,1,0)-f^k_{mnt}(0,1,1) \\ +f^k_{mnt}(1,0,0)+f^k_{mnt}(1,0,1)-f^k_{mnt}(1,1,0)-f^k_{mnt}(1,1,1)]/8 \quad (2)$$

$$F^k_{mnt}(1,0,0)=[f^k_{mnt}(0,0,0)+f^k_{mnt}(0,0,1)+f^k_{mnt}(0,1,0)+f^k_{mnt}(0,1,1) \\ -f^k_{mnt}(1,0,0)-f^k_{mnt}(1,0,1)-f^k_{mnt}(1,1,0)-f^k_{mnt}(1,1,1)]/8 \quad (3)$$

$$F^k_{mnt}(1,1,0)=[f^k_{mnt}(0,0,0)+f^k_{mnt}(0,0,1)-f^k_{mnt}(0,1,0)-f^k_{mnt}(0,1,1) \\ -f^k_{mnt}(1,0,0)-f^k_{mnt}(1,0,1)+f^k_{mnt}(1,1,0)+f^k_{mnt}(1,1,1)]/8 \quad (4)$$

$$F^k_{mnt}(0,0,1)=[f^k_{mnt}(0,0,0)-f^k_{mnt}(0,0,1)+f^k_{mnt}(0,1,0)-f^k_{mnt}(0,1,1) \\ +f^k_{mnt}(1,0,0)-f^k_{mnt}(1,0,1)+f^k_{mnt}(1,1,0)-f^k_{mnt}(1,1,1)]/8 \quad (5)$$

$$F^k_{mnt}(0,1,1)=[f^k_{mnt}(0,0,0)-f^k_{mnt}(0,0,1)-f^k_{mnt}(0,1,0)+f^k_{mnt}(0,1,1) \\ +f^k_{mnt}(1,0,0)-f^k_{mnt}(1,0,1)-f^k_{mnt}(1,1,0)+f^k_{mnt}(1,1,1)]/8 \quad (6)$$

$$F^k_{mnt}(1,0,1)=[f^k_{mnt}(0,0,0)-f^k_{mnt}(0,0,1)+f^k_{mnt}(0,1,0)-f^k_{mnt}(0,1,1) \\ -f^k_{mnt}(1,0,0)+f^k_{mnt}(1,0,1)-f^k_{mnt}(1,1,0)+f^k_{mnt}(1,1,1)]/8 \quad (7)$$

$$F^k_{mnt}(1,1,1)=[f^k_{mnt}(0,0,0)-f^k_{mnt}(0,0,1)-f^k_{mnt}(0,1,0)+f^k_{mnt}(0,1,1) \\ -f^k_{mnt}(1,0,0)+f^k_{mnt}(1,0,1)+f^k_{mnt}(1,1,0)-f^k_{mnt}(1,1,1)]/8 \quad (8)$$

Note that the subscript mnt in $F^k_{mnt}(i,j,h)$ represents the block index and (i,j,h) is a pixel location in that block. For example, $F^k_{000}(m,n,t)$ represents the (m,n,t)th pixel at the (0,0,0)th block for the kth level of the hierarchy. We also note that $F^k_{mnt}(0,0,0)$ in (1) (equivalently, all elements in $F^k_{000}(m,n,t)$) always satisfy $0\leq F^k_{mnt}(0,0,0)\leq 255$, because they are just a collection of DC values.

Equations (1)-(8) consist only of simple additions, subtractions, and shifts operations, so we can reduce the computational burdens substantially comparing to the convolution based wavelet transform with the higher-order filters.

Now, to collect the same frequency components of all 2x2x2 DCT coefficients in the video space we need to rearrange them for all $m=0,1,\cdots, N_1/2^k-1$, $n=0,1,\cdots, N_2/2^k-1$, $t=0,1,\cdots, T/2^k-1$ as follows;

$$F^k_{000}(m,n,t)=F^k_{mnt}(0,0,0) \quad (9)$$

$$F^k_{010}(m,n,t)=F^k_{mnt}(0,1,0) \quad (10)$$

$$F^k_{100}(m,n,t)=F^k_{mnt}(1,0,0) \quad (11)$$

$$F^k_{110}(m,n,t)=F^k_{mnt}(1,1,0) \quad (12)$$

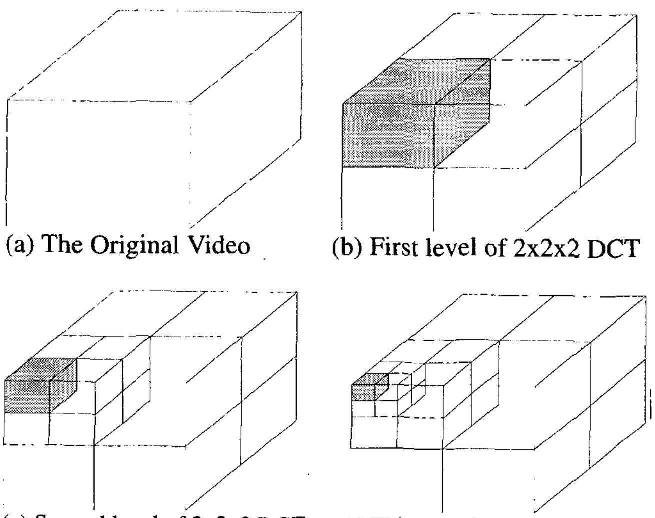
$$F^k_{001}(m,n,t)=F^k_{mnt}(0,0,1) \quad (13)$$

$$F^k_{011}(m,n,t)=F^k_{mnt}(0,1,1) \quad (14)$$

$$F^k_{101}(m,n,t)=F^k_{mnt}(1,0,1) \quad (15)$$

$$F^k_{111}(m,n,t)=F^k_{mnt}(1,1,1) \quad (16)$$

After the rearrangement, the upper-left corner block $F^k_{000}(m,n,t)$, $m=0,1,\cdots, N_1/2^k-1$, $n=0,1,\cdots, N_2/2^k-1$, $t=0,1,\cdots, T/2^k-1$, consists of DC components only and it should resemble the original video in the spatial domain. Therefore, at the next level of the hierarchy, if we execute the 2x2x2 DCT to $F^k_{000}(m,n,t)$, $m=0,1,\cdots, N_1/2^k-1$, $n=0,1,\cdots, N_2/2^k-1$, $t=0,1,\cdots, T/2^k-1$, recursively by changing the notations as $f^{k+1}_{mnt}(0,0,0)=F^k_{000}(2m,2n,2t)$, $f^{k+1}_{mnt}(0,1,0)=F^k_{000}(2m,2n+1,2t)$, $f^{k+1}_{mnt}(1,0,0)=F^k_{000}(2m+1,2n,2t)$, $f^{k+1}_{mnt}(1,1,0)=F^k_{000}(2m+1,2n+1,2t)$, $f^{k+1}_{mnt}(0,0,1)=F^k_{000}(2m,2n,2t+1)$, $f^{k+1}_{mnt}(0,1,1)=F^k_{000}(2m,2n+1,2t+1)$, $f^{k+1}_{mnt}(1,0,1)=F^k_{000}(2m+1,2n,2t+1)$, $f^{k+1}_{mnt}(1,1,1)=F^k_{000}(2m+1,2n+1,2t+1)$, for all $m=0,1,\cdots, N_1/2^k-1$, $n=0,1,\cdots, N_2/2^k-1$, $t=0,1,\cdots, T/2^k-1$, we can obtain an octave-band representation of the video as shown in Figure 1.



(a) The Original Video (b) First level of 2x2x2 DCT

(c) Second level of 2x2x2 DCT (d)Third level of 2x2x2 DCT

Figure 1. Multiresolution representation of the 2x2x2 DCT Coefficients

To get the original video back from the multiresolution represented data, we just do the reverse rearrangement-IDCT process from the highest image level (i.e., the lowest resolution). Then, since the DCT and IDCT are unitary operations, we can perfectly reconstruct the original image data. That is , we can execute equations (17)-(24) for the combined rearrangement-IDCT process for all m=0,1,⋯ , $N_1/2^k-1$, n=0,1,⋯ , $N_2/2^k-1$, t=0,1,⋯ , $T/2^k-1$.

$$f^k_{mnt}(0,0,0)=F^k_{000}(m,n,t)+F^k_{001}(m,n,t)+F^k_{010}(m,n,t)+F^k_{011}(m,n,t)+F^k_{100}(m,n,t)+F^k_{101}(m,n,t)+F^k_{110}(m,n,t)+F^k_{111}(m,n,t) \quad (17)$$

$$f^k_{mnt}(0,1,0)=F^k_{000}(m,n,t)+F^k_{001}(m,n,t)-F^k_{010}(m,n,t)-F^k_{011}(m,n,t)+F^k_{100}(m,n,t)+F^k_{101}(m,n,t)-F^k_{110}(m,n,t)-F^k_{111}(m,n,t) \quad (18)$$

$$f^k_{mnt}(1,0,0)=F^k_{000}(m,n,t)+F^k_{001}(m,n,t)+F^k_{010}(m,n,t)+F^k_{011}(m,n,t)-F^k_{100}(m,n,t)-F^k_{101}(m,n,t)-F^k_{110}(m,n,t)-F^k_{111}(m,n,t) \quad (19)$$

$$f^k_{mnt}(1,1,0)=F^k_{000}(m,n,t)+F^k_{001}(m,n,t)-F^k_{010}(m,n,t)-F^k_{011}(m,n,t)-F^k_{100}(m,n,t)-F^k_{101}(m,n,t)+F^k_{110}(m,n,t)+F^k_{111}(m,n,t) \quad (20)$$

$$f^k_{mnt}(0,0,1)=F^k_{000}(m,n,t)-F^k_{001}(m,n,t)+F^k_{010}(m,n,t)-F^k_{011}(m,n,t)+F^k_{100}(m,n,t)-F^k_{101}(m,n,t)+F^k_{110}(m,n,t)-F^k_{111}(m,n,t) \quad (21)$$

$$f^k_{mnt}(0,1,1)=F^k_{000}(m,n,t)-F^k_{001}(m,n,t)-F^k_{010}(m,n,t)+F^k_{011}(m,n,t)+F^k_{100}(m,n,t)-F^k_{101}(m,n,t)-F^k_{110}(m,n,t)+F^k_{111}(m,n,t) \quad (22)$$

$$f^k_{mnt}(1,0,1)=F^k_{000}(m,n,t)-F^k_{001}(m,n,t)+F^k_{010}(m,n,t)-F^k_{011}(m,n,t)-F^k_{100}(m,n,t)+F^k_{101}(m,n,t)-F^k_{110}(m,n,t)+F^k_{111}(m,n,t) \quad (23)$$

$$f^k_{mnt}(1,1,1)=F^k_{000}(m,n,t)-F^k_{001}(m,n,t)-F^k_{010}(m,n,t)+F^k_{011}(m,n,t)-F^k_{100}(m,n,t)+F^k_{101}(m,n,t)+F^k_{110}(m,n,t)-F^k_{111}(m,n,t) \quad (24)$$

Then, for the next level of the hierarchy (i.e., for the next higher resolution), we set $F^{k-1}_{000}(2m,2n,2t)=f^k_{mnt}(0,0,0)$, $F^{k-1}_{000}(2m,2n+1,2t)=f^k_{mnt}(0,1,0)$, $F^{k-1}_{000}(2m+1,2n,2t)=f^k_{mnt}(1,0,0)$, $F^{k-1}_{000}(2m+1,2n+1,2t)=f^k_{mnt}(1,1,0)$, $F^{k-1}_{000}(2m,2n,2t+1)=f^k_{mnt}(0,0,1)$, $F^{k-1}_{000}(2m,2n+1,2t+1)=f^k_{mnt}(0,1,1)$, $F^{k-1}_{000}(2m+1,2n,2t+1)=f^k_{mnt}(1,0,1)$, $F^{k-1}_{000}(2m+1,2n+1,2t+1)=f^k_{mnt}(1,1,1)$, for all m=0,1,⋯ , $N_1/2^k-1$, n=0,1,⋯ , $N_2/2^k-1$, t=0,1,⋯ , $T/2^k-1$. Other frequency components for Equations (17)-(24) such as $F^{k-1}_{010}(,)$, $F^{k-1}_{100}(,)$, $F^{k-1}_{001}(,)$, $F^{k-1}_{011}(,)$, $F^{k-1}_{101}(,)$, and $F^{k-1}_{111}(,)$ are available from the corresponding blocks at (0,1,0), (1,0,0), (1,1,0), (0,0,1), (0,1,1), (1,0,1), and (1,1,1), respectively.

Ⅲ. Three-Dimensional ZeroTree Coding

This section outlines the major concepts of a 3-D zerotree coding algorithm. Shapiro's zerotree encoding algorithm defines a data structure called a zerotree[2]. In this paper, we extend the Shapiro's 2-D zerotree coding algorithm to 3-D space.

The multiresolution represented video has the parent-child structure as shown in Fig 2. In 2-D, a zerotree starts as a single coefficient then it has 4 child coefficients. In 3-D, a zerotree starts as a single coefficient. Then, for the next level, it will have 2x2x2=8 child coefficients, each of which has another eight, and so on.

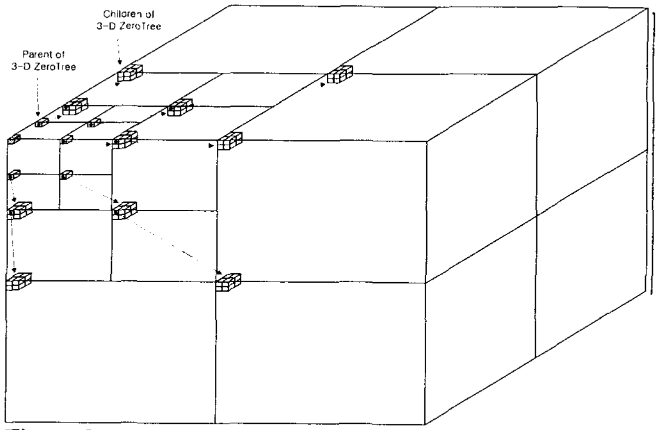


Figure 2. Parent-Child Structure of the 3-D zerotree.

Fig 3 shows the scan order for encoding a significant map. As shown in Fig 3, the coefficients are scanned from the highest level to the lowest level. Each coefficient is determined to ZTR(ZeroTree Root), IZ(Isolated Zero), POS(Positive Significant), and NEG(Negative Significant) from parent-child structure and scan order[2].

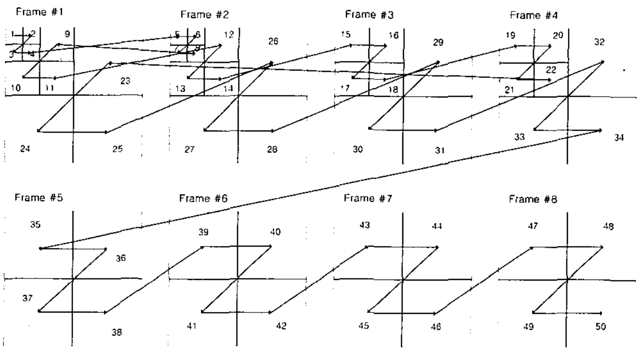


Figure 3. Scan Order of the 3-D zerotree.

Ⅳ. Progressive Video Transmission with the Bit-Rate Scalability

Progressive video transmission with the bit-rate scalability is composed of the following steps. First, the original video data is represented by the octave-band multiresolution using the 2x2x2 DCT as shown in Figure 1. Then, the multiresolution represented data is compressed using the 3-D zerotree. Second, the compressed video data is transmitted with various bit-rates. For example, the original video data can be progressively transmitted to the receiver as follows. Suppose that the multiresolution represented N_1xN_2xT data is compressed by, say, A bpp (bit per pixel). Then, the sender (the server) can transmit a compressed data with B_1 ($0<B_1<A$) bpp which corresponds to the higher rank of the significant map of the 3-D zerotree. After that, the differential compressed data from B_2 ($B_1< B_2<A$) bpp to B_1 bpp is transmitted on the request of the receiver. Subsequently, the sender transmits the differential compressed data from B_n bpp and B_{n-1} bpp ($B_{n-1}<B_n<A$). Finally, the remaining differential compressed data that A bpp and B_n bpp is transmitted on the request of the receiver. At any step, the receiver can terminate the transmission. Figure 4 shows the block diagram of the progressive video transmission with bit-rate scalability.

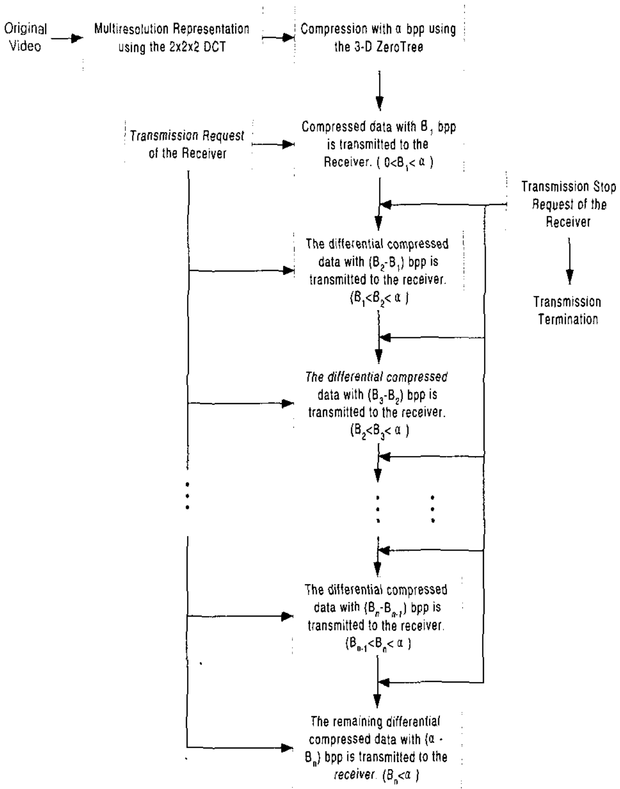
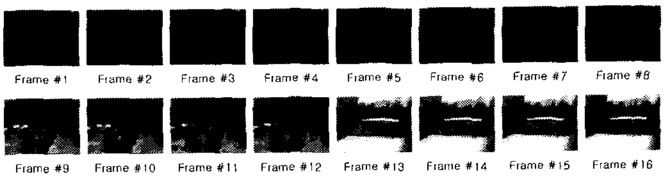


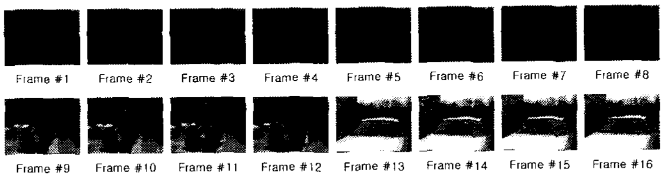
Figure 4. The block diagram of the progressive video transmission with bit-rate scalability.

Experimental Results

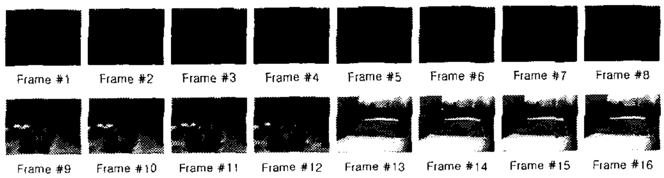
In this section, we present the simulation results of the multiresolution progressive video transmission with the bit-rate scalability using the 2x2x2 DCT. The 2x2x2 DCT consists only of simple additions, subtractions, and shifts operations, so we can reduce the computational burdens substantially comparing to the convolution based wavelet transform with the higher-order filters. Figure 5-(a) shows the decoded video with the first level of the video data (bpp=0.16). Figure 5-(b) shows the decoded video with the second level of the image data (bpp=0.4). Figure 5-(c) shows the decoded video with the third level of the image data (bpp=0.8). Table 1 shows the average PSNR's of the decoded video quality at each bit-rate.



(a) Progressively received video with 0.16 bpp



(b) Progressively received video with 0.4 bpp



(c) Progressively received video with 0.8 bpp

Figure 5. The progressively received video with bit-rate scalability

Table 1. The decoded video quality of a 2x2x2 DCT

Bit-rate	Bpp = 0.16	Bpp = 0.4	Bpp = 0.8
Average PSNR	29.09 dB	32.73 dB	35.25 dB

Conclusions

In this paper, we propose a progressive video transmission with bit-rate scalability using a 2x2x2 DCT. Since the 2x2x2 DCT and its inverse needs simple computations and less memories, it is suitable for real-time video transmission applications such as internet shopping. Experimental results on the progressive transmission with the bit-rate scalability show that the video transmitted on each bit-rate progressively on the request of the receiver and the video quality is sufficient for the internet applications.

References

[1] Limin Wang and Morris Goldberg, "Progressive Image Transmission Using Vector Quantization on Images in Pyramid Form," IEEE Transactions on Communications, Vol.37, No.12, December 1989.

[2] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," IEEE Transactions on Signal Processing, Vol.41, pp.3445-3462, December 1993.