Weighted Embedded Zero Tree for Scalable Video Compression

Roya Choupani [†] Stephan Wong[‡] Mehmet R. Tolun[§]

Abstract—**Video streaming over the Internet has gained popularity during recent years mainly due to the revival of videoconferencing and videotelephony applications and the proliferation of (video) content providers. However, the heterogeneous, dynamic, and best-effort nature of the Internet cannot always guarantee a certain bandwidth for an application utilizing the Internet. Scalability has been introduced to deal with such issues (up to a certain point) by intelligently separating any information stream into multiple streams. The reception of one, several, or all stream influences the perceived quality of the information as basic, improved, or best, respectively. In addition, wavelet-based scalability combined with representation methods such as embedded zero trees (EZWs) improves the decode-ability of the stream even when only the initial part of the streams have been received. In this paper, we propose a method to improve on the compression rate of the EZW for scalability purposes by reducing the number of levels used in the tree. Therefore, the proposed method should be able to deal more efficiently with the mentioned scalability issues in low bandwidth network. Initial experimental show that the first two layers of the generated EZW are about 22.6% more concise.**

Index Terms—**Video coding, Scalability, Wavelet, Embedded Zero Tree**

I. INTRODUCTION

In the past several years, the steady growth in bandwidth on the Internet has allowed increasingly more applications to incorporate streaming audio and video content [1],[2]. As the best-effort internet is not capable of providing a fixed bandwidth between a source and its destination(s), an objective of video coding has been to provide acceptable video quality at different bitrates. The impracticality of storing multiple copies of a video (sequence) - at different bitrates, resolution, etc. - created the need for scalability that allows video to be easily recoded to meet different requirements depending on a particular situation. Video scalability entails the logical subdivision of a single video stream into a base layer stream with multiple enhancement layer streams. The base layer stream encompasses the most rudimentary quality and/or resolution of the video. However, combined with the enhancement layer streams the following can be improved: resolution (spatial scalability), number of frames (temporal scalability), or video quality (SNR scalability). Consequently, the transmission of the base layer stream receives the highest priority while the other streams are less important. On top of this, the less than perfect nature of the Internet can lead to packet losses or lengthy packet retransmissions (also effectively losing packets when displaying video in 'real-time'). Therefore, coding schemes have been introduced in the recent past to deal with such conditions. One such scheme incorporate wavelet transforms that are effectively capable of generating different subbands with - in simple terms - important coefficients and less important coefficients. The ensuing embedded zero tree (EZW) method allows an ordered representation of these coefficient from important to less important. In conclusion, when a transmission fails, still a complete picture (not partially like in the older JPEG standard) can be decoded although at a much lower quality. In this paper, we propose an alternate representation method for the EZW coefficients that organizes coefficients such that the number of symbols in the first layers is less than standard EZW while this number is larger in remaining layers. This feature has the benefit of having smaller data size when only a low quality video is to be transmitted over a connection with narrow bandwidth. This property is achieved by applying a weight value to each layer which increases the probability of finding zero trees at higher threshold levels and reducing the symbol rate. The experiments with still images show that on average about 22.6% reduction in data size is achievable on the first two layers .

The remainder of this paper is organized as follows. In Section 2, we discuss the related work and highlight how we differ from these works. In Section 3, we introduce our proposed method and describe its encoding and decoding algorithms providing an example. In Section 4, we explain our experimental setup and discuss the results. In Section 5, we draw our conclusions.

II. RELATED WORK

Rapid advances in multimedia technologies together with the growth of the Internet have enabled many new applications and services. Although network bandwidth and digital devices' storage capacity are increasing rapidly, video compression continues to play an important role due to the exponential growth in size of multimedia content. Furthermore, many applications require not only high compression efficiency, but also enhanced flexibility for supporting real time usage. For example, in order to effectively deliver video over heterogeneous networks such as the Internet and wireless channels, error resilience and bit rate scalability are important; and in order to make a coded video bitstream usable by different types of digital devices regardless of their computational, display, and memory capabilities, resolution/temporal scalability is needed. Standard coders such as H.26x and MPEG-x are no longer

---

†Computer Engineering Department, Çankaya University, Ankara Turkey, and Delft University of Technology, Delft, The Netherlands, emails: roya@cankaya.edu.tr; roya@dutepp0.et.tudelft.nl

‡Computer Engineering Department, Delft University of Technology, Delft, The Netherlands, email: J.S.S.M.Wong@ewi.tudelft.nl

§Computer Engineering Department, Çankaya University, Ankara Turkey, email: tolun@cankaya.edu.tr

able to satisfy the above requirements with their incorporated scalability. The recent MPEG-4 standard [3] adopts object-based video coding in order to support more applications. However, scalability in MPEG-4 is limited. Experiments with MPEG-2 and H.263 in scalable mode show that, compared with non-layered coding [4] [5] [7], the average peak signal-to-noise ratio decreases about one dB with each layer. Furthermore, it is difficult for these coding schemes to achieve scalability because there is always a potential drifting problem [6] [9] associated with predictive coding. Although there are proposals for MPEG-4 streaming video profile on fine-granularity scalable video coding, these are limited to providing flexible rate scalability only and the coding performance is still about 1-1.5 dB lower than that of a non-layered coding scheme [6]. As an alternative to the predictive approaches in various video coding standards, wavelet video coding has been investigated recently by several researchers [8] [2] and shown to be competitive with standard motion compensated (MC) predictive coding. Although wavelet video coders usually require a larger buffer and incur a longer delay than standard coders, an important feature of the wavelet approach is the support of scalability in the compressed video. With embedded coding techniques such as embedded zero tree and set partitioning in hierarchical trees, wavelet video coding achieves continuous bit rate scalability. Furthermore, because of the multi-resolutional nature of wavelet analysis, both resolution and temporal (frame rate) scalabilities can be easily supported. Mallat [11] discusses the wavelet representation as a suitable tool for multi-resolution signal decomposition. Such a decomposition of video signals may permit temporal and spatial scalability. This important feature is investigated by many researchers. One of the main reasons for the success use of DWT in scalable video coding is the introduction of data structures to represent wavelet coefficients while minimizing the required memory space. Embedded Zero Tree (EZW) is one of these data structures which is widely used for organizing and transmission of the wavelet coefficients. A draw back of EZW is that the large coefficients in high frequency sub-bands effects the compression rate in low frequency sub-bands. This drawback stems from the fact that EZW is not basically designed for scalable video coding and therefore the present priority mechanism is based on the magnitude of the coefficient not the associated frequency of each coefficient. The proposed method described below is an improvement over EZW which enhances the priority encoding of the wavelet coefficients in EZW.

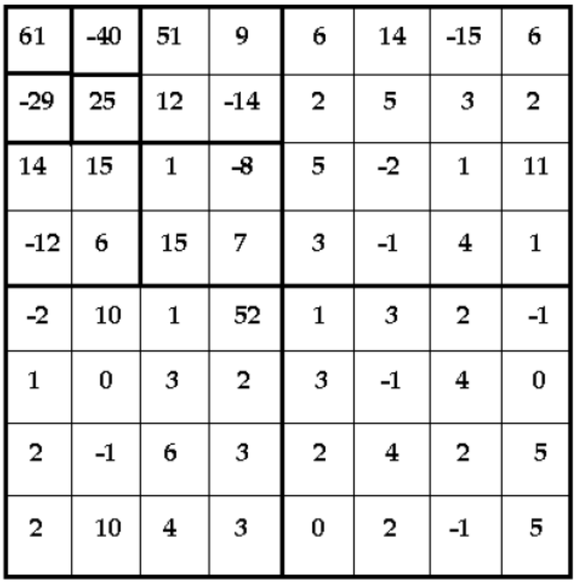
A. Embedded Zero Tree (EZW)

The Embedded Zero Tree Wavelet encoding is based on creating a quad-tree having four children at each node (except the root node that only has 3 children) and storing the obtained coefficients from the wavelet transform. The main reason to utilize this storage method is that the coefficients close to the root have a larger absolute value. Therefore, coefficients closer to the leaves of the tree are less significant and can be sometimes represented by a single symbol to greatly reduce the data size. The algorithm follows the procedure described below. The magnitude of each wavelet coefficient in the quad-tree, starting with the root of the tree is compared to a threshold T. If the magnitudes of all the wavelet coefficients in the (sub)tree - including the root of the (sub)tree - are smaller than T, the entire (sub)tree is substituted by a single symbol, the zero tree symbol ***t***. However, if a coefficient in the subtree excluding the root is larger than T, it is substituted by a symbol representing whether the subtree is either a significant (if the root is larger than T) or insignificant (if the root is smaller than T). In the latter two cases, descendant nodes (and their respective subtrees) are further examined to determine whether they contain zero trees or not. This process is carried out until all the nodes in all the trees are examined for possible sub zero tree structures. The significant wavelet coefficients in a tree are represented by one of two symbols, **P** or **N**, depending on whether their values are positive or negative, respectively. An insignificant coefficient which is not in a zero tree is represented by a **Z** symbol. The process of classifying the coefficients as **t**,**Z**, **P**, or **N** is referred to as the dominant pass. This is subsequently followed by a subordinate pass in which the significant wavelet coefficients in the image are refined by determining whether their magnitudes lie within the intervals [T,3T/2) and [3T/2,2T). Those wavelet coefficients whose magnitudes lie in the interval [T,3T/2) are represented by the symbol 0 (LOW), whereas those with magnitudes lying in the interval [3T/2,2T) are represented by the symbol 1 (HIGH). To refine the coefficients which were marked either as **P** or **N** in the subordinate pass, we push them in a FIFO in the dominant pass. These coefficients are reduced by current threshold value after refining. Subsequent to the completion of both the dominant and subordinate passes, the threshold value T is reduced by a factor of 2, and the process is repeated. An EZW decoder reconstructs the image by progressively updating the values of each wavelet coefficient in a tree as it receives the data. The following example illustrates two steps of EZW encoder.

Assuming the initial threshold value to be 32 we have:
Dominant Pass 1: PNZtPttttZttttttttPtt
Coefficients in FIFO : 61, -40, 51, 52
Subordinate Pass 1: 1011
The 2nd pass with a threshold of 16 is presented in the following:
Dominant Pass 2: ZtNPtttttttt
Coefficients in FIFO : 29, -8, 19, 20, -29, 25
Subordinate Pass 2: 100011

III. Proposed Method

It must be clear from the previous description that the initial number of large coefficients before any pass determines the amount of encoded data needed in the ensuing pass. Keeping in mind scalability issues in low bandwidth networks, we propose a method to reduce the amount of encoded data in the first pass and whenever additional bandwidth is available, the larger amount of encoded data



61	-40	51	9	6	14	-15	6
-29	25	12	-14	2	5	3	2
14	15	1	-8	5	-2	1	11
-12	6	15	7	3	-1	4	1
-2	10	1	52	1	3	2	-1
1	0	3	2	3	-1	4	0
2	-1	6	3	2	4	2	5
2	10	4	3	0	2	-1	5

Fig. 1. Sample data representing wavelet coefficients

in the subsequent passes can be transmitted. The priority mechanism of EZW has been improved by considering a level based weight. The coefficients stored in the quad tree are multiplied by a weight value before thresholding. Adjusting weight values in a decreasing order defers transmission of the significant coefficients coming from the high frequencies. The following subsections describe encoding and decoding procedures of the proposed method.

A. Encoding

In the example given above, the dominant pass in the first iteration includes four ***t***s followed by a **Z**, 7 ***t***s, a **P** and 2 ***t***s. If the element in the 5^{th} row and 4^{th} column which is equal to 52 (greater than the threshold) is replaced by a small number, the dominant pass string will be PNZtPttttttttttt which is four symbols shorter than the original string. Replacing this entry however, generates some difference with the original image. This deviation from the original values becomes more important when we increase the number of levels decoded at the receiver side since the elements at the lower levels of the quad tree correspond to high frequency content of the image. This means that in the coarse level replacing isolated elements with values less than the threshold will not create a major visual effect. To replace the large coefficient values in the low levels of the tree when encoding at the coarse stage, we propose a weighted form of representing the coefficients. Assume the quad tree has a height of n. For each level a weight factor w_i is defined as a positive number so that $w_1 = 1$ and $w_i < w_j$ if i>j. During the dominant pass the values are multiplied by their corresponding weight factors before comparing to the threshold. As the weight factors are less than one, the large coefficients will not appear in encoding with large threshold values (coarse stage). In the subordinate pass the bit strings are obtained using the weighted coefficients. The method applied to the sample data will produce the following results:

Weights: w_1= 1, w_2= 0.8, w_3= 0.6
Initial threshold value = 32
Dominant Pass 1: PNZttttt
Coefficients in FIFO : 61, -40
Subordinate Pass 1: 10
Pass 2, Threshold = 16
Dominant Pass 2: ZZNP Pttt tZtt tttt tttt tPtt
Coefficients in FIFO : 29, -8, -29, 25, 30 (51x0.6) , 20 (52x0.4)
Subordinate Pass 2: 101110

The large coefficients which do not appear in the first pass emerge in the second pass. This justifies the long symbol string in the second pass. In fact the method defers the processing of some of the symbols which contain large frequency information to subsequent passes which is more suitable for a scalable video coding scheme. In addition the coefficients in the FIFO have been truncated after being multiplied by their respective weight factors.

B. Decoding

To avoid loss of information the extracted coefficients are divided by the corresponding weight factors. This reconstruction of the wavelet coefficients introduces some errors due to the truncation process in the subordinate passes. In the above example the reconstructed values after the second pass are as follows.

EZW:

Original coefficient	Reconstructed coefficient
61	56
-40	-32
51	48
52	48
-29	-24
25	24

Weighted EZW:

Original coefficient	Reconstructed coefficient
61	56
-40	-32
51	40
52	40
-29	-24
25	24

Despite the fact that the reconstructed values for the coefficients in the first passes is farther from their true value, closer approximation is obtained in the next passes. This is also compliant with the main idea of the proposed method which is deferring the exact reconstruction to the late passes and reducing the string length in the first passes.

IV. Experimental Results

To evaluate the performance of the proposed method we have created the zero trees for a set of images. All images are gray scale images thus a single pixel value component is available. The number of levels in the zero trees has been considered as a parameter. An important issue in the proposed method is adjusting the weight values for each tree level. The larger weight values create a zero tree which is close to the EZW tree. Smaller weight values on the other hand pushes the larger list of coefficient values to the lower levels of the tree which means the smaller the weight factor, the more compact image size in upper levels. This feature is specially important since reducing image size when the bandwidth is limited for a given connection or display resolution is low, should be performed by increasing the number of levels in the hierarchical tree. However, increasing the number of levels in the generated zero trees means more processing time due to wavelet transform which is a serious consideration for real time applications. A suitable choice of weight value can adequately solve this problem. Figures 3 and 4 show the reconstructed images using first level only and first two levels only respectively. Despite the improvement in the compression rates, no noticeable quality degradation is present in the images.

Fig. 2. Reconstructed images using first level of zero trees. **Left:** Original EZW **Right:** Weighted EZW

The compression rates using EZW and Weighted-EZW methods have been compared with respect to the number of levels in the generated trees. The results of comparison are given in table 1. We have only considered the range of 3 to 5 for tree levels however, the results show there is an increase in the compression rate with increase in the number of levels.

Levels	Pass 1	Pass2	Pass 1	Pass2
3	69.3	50	87.5	54.4
4	73.2	71.1	90.2	80.33
5	78.8	76.43	94.11	89.6

Table 1. Rate comparison for EZW and Weighted-EZW

V. Conclusion

Large fluctuations in the communication bandwidth and heterogeneity of the Internet are two main reasons for using scalability in encoding video data. The modification introduced here for Embedded Zero Tree algorithm further reduces the data size in the low frequency and coarse stage of video coding and makes quick transmission of the data feasible. The method eliminates the need of more layers of wavelet tree for higher compression by pushing the details and therefore large data size to the lower layers of the tree. The experimental results show that using only a single level, 22.6% improvement can be achieved with almost no visually noticeable decrease in video quality. In case of using two layers for reconstruction the improvement rate drops to 9.13% which indicates that the method is suitable for low resolution display screens or low bandwidth channels where high rate of compression is necessary.

Fig. 3. Reconstructed images using first two level of zero trees. **Left:** Original EZW **Right:** Weighted EZW

References

[1] G. Conklin, G. Greenbaum, K. Lillevold, A. Lippman, and Y. Reznik, Video Coding for Streaming Media Delivery on the Internet, IEEE Trans. Circuits and Systems for Video Technology, March 2001.

[2] D. Wu, Y. Hou, W. Zhu, Y.-Q. Zhang, and J. Peha, Streaming Video over the Internet: Approaches and Directions, IEEE Transactions on Circuits and Systems for Video Technology, March 2001.

[3] Coding of Audio-Visual Objects, Part-2 Visual, Amendment 4: Streaming Video Profile, ISO/IEC 14 496-2/FPDAM4, July 2000.

[4] B. G. Haskell, A. Puri, and A. N. Netravali, "Digital Video: An Introduction to MPEG-2", New York: Chapman and Hall, Sept. 1996.

[5] R. Aravind, M. R. Civanlar, and A. R. Reibman, Packet loss resilience of MPEG-2 scalable video coding algorithm, IEEE Trans. Circuits Syst. Video Technol., vol. 6, pp. 426435, Oct. 1996.

[6] S. F. Chang and A. Vetro, "Video adaptation: Concepts, technologies, and open issues," Proc. IEEE, vol. 93, no. 1, pp. 148-158, Jan. 2005.

[7] "Video coding for low bit rate communication," Int. Telecommun. Union-Telecommun. (ITU-T), Geneva, Switzerland, Recommendation H.263, 1998.

[8] G. Strang, "Wavelets", American Scientist, Vol. 82, 1992, pp. 250-255.

[9] S. Nogaki, M. Ohta, "An overlapped block motion compensation for high quality motion picture coding", Proc IEEE Int. Symp. Circuits and Systems, vol. 1, pp 184-187, 1992

[10] J. M. Shapiro, "Embedded image coding using zerotrees of wavelets coefficients", IEEE Transactions on Signal Processing, vol. 41, no. 12, pp. 3445-3462, December 1993.

[11] S. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 11 (7),pp. 674-693, 1989.