RESEARCH ARTICLE OPEN ACCESS

Efficient video compression using EZWT

Mr. Amit S. Hatekar *, Ms. Poorva G. Waingankar**,***Dr. Sangeeta Joshi
*(Assistant Prof. Department of Electronics and telecommunication , Thadomal Shahani Engineering College , Mumbai University, Mumbai; PG Student of Thakur College of Engineering , Kandivali East Mumbai.)
** (Associate Professor and Head of Electronics Department, Thakur College of Engineering , Kandivali East, Mumbai University, Mumbai)
***(Professor of Electronics Engineering and Technical Advisor,Vidyalankar Institute of Technology , Wadala Mumbai.)

ABSTRACT
In this article, wavelet based lossy video compression algorithm is presented. The motion estimation and compensation, being an important part in the compression, is based on segment movements. The proposed work is based on wavelet transform algorithm Embedded Zeroed WaveletTransform (EZWT). Based on the results of peak signal to noise ratio (PSNR), mean squared error (MSE), different videos are analyzed. Maintaining the PSNR to acceptable limits the proposed EZWT algorithm achieves very good compression ratios making the technique more efficient than the 2-Discrete Cosine Transform (DCT) in the H.264/AVC codec. The method is being suitable for low bit rate video showing highest compression ratio and very good PSNR of more than 30dB.

I. INTRODUCTION

An important part in image and video compression systems is a transform [4] .A transform is used to transform prediction residuals of image intensities, such as the motion-compensation residual (MC residual), the resolution enhancement residual in scalable video coding, or the intra-prediction residual in H.264/AVC. A conventional transform 2-D DCT or the 2-D Discrete Wavelet Transform (2-D DWT), is carried out as a separable transform by cascading two 1-D transforms in the vertical and horizontal dimensions. The new transforms adapt to locally anisotropic features in images by performing the filtering along the direction where image intensity variations are smaller. This is achieved, for example,by directional lifting implementations of the DWT [4]. The discrete wavelet transform (DWT) [5] has gained wide popularity due to its excellent decorrelation property.

The significant coding efficiency of H.264 video compression technology is used in a wide range of streaming video applications over a variety of media. The H.264 standard supports motion estimation on blocks from 16x16 to 4x4 pixels.

II. H.264/MPEG-4 AVC codec overview

1.1 Coding structure

The basic coding structure is referred to as motion-compensated transforms coding structure. Coding of video is performed frame by frame. In earlier standards, a slice consisted of a sequence of macro blocks with each macro block (MB) consisting of 16X16 luminance (y) and associated two chrominance (Cb and Cr) components. Each macro block's 16X16 luminance is partitioned into 16X16, 16X8, 8X16, and 8X8, and further, each 8_8 luminance can be sub-partitioned into 8X8, 8X4, 4X8 and 4X4. The 4X4 sub-macro block partition is called a block. Currently, only 4:2:0 chroma format and 8-bit sample precision for luma and chroma pixel values is supported in the standard. In 4:2:0 chrome format, each macro block associates two 8X8 chroma components with 16 X 16 luminance. Research is being done to extend the standard to 4:2:2 and 4:4:4 chroma formats and higher than 8-bits resolution [1].

Slices are individually coded and are the coding units, while pictures plus associated data can be considered as being the access units. There are three basic slices types:-

a) I—(Intra),
b) P—(Predictive), and B—(Bi-predictive)

Slices. This is basically a nomenclature as well as functionality extension of the I , P , and B-picture concept of earlier standards. In H.264/MPEG-4 AVC standard, I-slice macro blocks are compressed without using any motion prediction (also true of all earlier standards as well) from the slices in other pictures. P-slices consist of macro blocks that can be compressed by using motion prediction, but P-slices can also have intra macro blocks. Macro blocks of a P-slice when using motion prediction must use one prediction only (uni-prediction). Also, both I- and P-slices may or may not be marked as used for reference. Macroblocks of a B-slice when using motion prediction can use two predictions (biprediction). Like. Such B-slices that are used as reference for motion prediction, are informally called stored B-slices due to the need for storing them unlike traditional B-slices.

1.2 Encoder

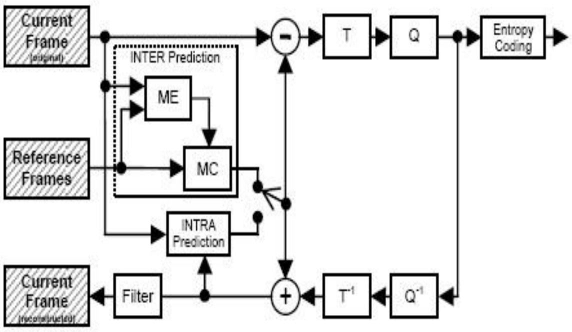


Fig 1a .H.264 Encoder

As shown in fig1a above, each macroblock is encoded in intra or inter mode and, for each block in the macroblock, a prediction is formed based on reconstructed frames. In Intra mode, prediction is done on the basis of samples in the current frame that has been first encoded, decoded and reconstructed In Inter mode, prediction is done by taking into account the motion-compensated prediction from one or two reference frames This prediction is then subtracted from the current block to produce a residual (difference) block that is transformed (using a block transform using) and quantized to give a set of quantized transform coefficients which are reordered and entropy encoded [1].The entropy-encoded coefficients, together prediction modes, quantizer parameter, motion vector information, etc. form the compressed bitstream which is passed to a Network Abstraction Layer (NAL) [1] for transmission or storage.

1.3 Decoder

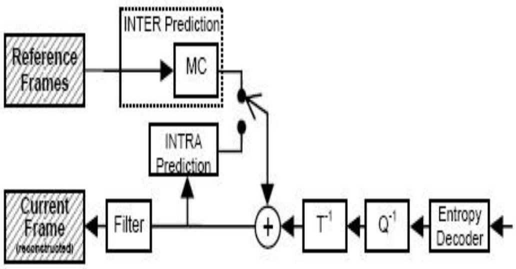


Fig 1B .H.264 Decoder

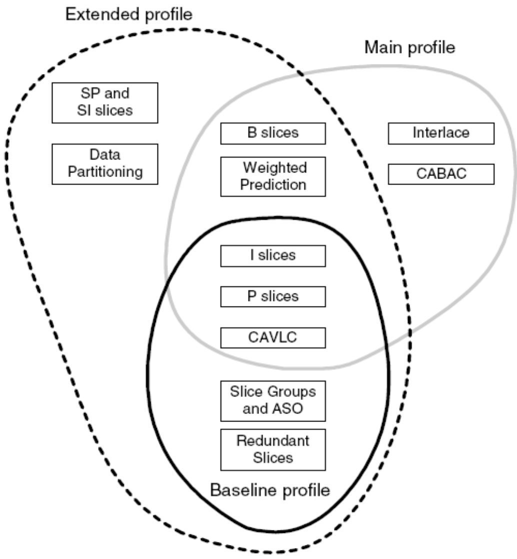
As shown in fig 1b above, the decoder after receiving a compressed bitstream , the entropy decoder decodes the data to generate a set of quantised coefficients. These are scaled and inverse transformed. Using the header information decoded from the bit stream, the decoder creates a prediction block, identical to the original prediction formed in the encoder. This block is then filtered to create a decoded block which resembles the reconstructed frame.

1.4 Overview of profiles

The following three basic profiles [1] are defined :-

a) Baseline profile,
b) Extended profile,
c) Main profile.

Figure 3 below shows the categorization of different types of profiles . And as seen from Table 1 below , our focus is to be on base line profile as it is used for low bit rate video conferencing.



H.264 Baseline, Main and Extended profiles

FIG 2 .H.264 PROFILES [1]

The Baseline profile includes I- and P-slice coding, enhanced error resilience tools (flexible macroblock ordering (FMO), arbitrary slices and redundant slices), and CAVLC. It does not include [1] B-slices, SI- or SP-slices, interlace coding tools, and entropy coding with arithmetic coding (CABAC). It was designed for low delay applications, as well as for applications that run on platforms with low processing power and in high packet loss environment. Among the three profiles, it offers the least coding efficiency.

The Extended profile being a superset of the Baseline profile , it includes B-, SP- and SI-slices, data partitioning, and interlace coding tools. It however does not include CABAC. It is thus more complex but also provides better coding efficiency. Its intended applications were streaming video.

The Main profile includes all , the I-, P- and B-slices, interlace coding, CAVLC and CABAC. It does not include some error resilience tools (e.g. FMO), data partitioning, or SI and SP slices. It shares common tools such as I- and P-slices, and CAVLC with both the Baseline and Extended profiles. In addition it shares B-slices and interlaced coding tools with the Extended-profile. The Main profile was intended to provide the highest possible coding efficiency[1] whic is not required in low bit rate video.

Application	Requirements	H.264 Profiles
Broadcast television	Coding efficiency, reliability (over a 'controlled' distribution channel), interlace, low-complexity decoder	Main
Streaming video	Coding efficiency, reliability (over an 'uncontrolled' packet-based network), scalability	Extended
Video storage and playback (e.g. DVD)	Coding efficiency, interlace, low-complexity decoder	Main
Videoconferencing	Coding efficiency, reliability, low latency, low-complexity encoder and decoder	Baseline
Mobile video	Coding efficiency, reliability, low latency, low-complexity encoder and decoder, low power consumption	Baseline
Studio distribution	Lossless or near-lossless, interlace, efficient transcoding	Main

TABLE 1. APPLICATION REQUIREMENTS OF H.264 PROFILES [1]

III. Embedded Zeroed Tree Wavelet (EZWT)Encoding

Embedded Zero trees of Wavelet transforms (EZWT) is a lossy image compression technique , useful at low bit rates. [3] Here high compression ratios are achieved , with most of the coefficients produced thereon with the wavelet transform reaches zero, or very close to zero. This is due to the fact that "real world" images tend to contain mostly low frequency information (highly correlated). By considering the transformed coefficients as a tree (or trees) with the lowest frequency coefficients present at the root node and with the children of each tree node being spatially correlated with the coefficients in the next higher frequency sub band, there is a high probability that one or more sub trees will contain coefficients which are zero or nearly zero. Such subtrees are called **zerotrees** [6] .

In zerotree based image compression scheme such as EZWT, the statistical properties of the trees are used, in order to efficiently code the locations of the significant coefficients. According to the theory of EZWT wherein most of the coefficients will be zero or close to zero, the spatial locations of the significant coefficients contribute to a large portion of the total size of a typical compressed image. A coefficient is considered significant [7] if its magnitude (or magnitudes of a node and all its descendants in the case of a tree) is above a particular threshold. Setting a threshold close to the maximum coefficient magnitudes and then gradually and iteratively decreasing the threshold [7] , it is possible to create a compressed representation of an image. Due to the structure of the trees, it is very likely that if a coefficient in a particular frequency band is insignificant, then all its descendants (the spatially related higher frequency band coefficients) will also be insignificant.The compression algorithm consists of a number of iterations through a dominant pass and a subordinate pass[3] [7], the threshold is updated (reduced by a factor of two) after each iteration. The dominant pass encodes the significance of the coefficients which have not yet been found significant in earlier iterations, by scanning the trees and emitting one of the four symbols. The subordinate pass emits one bit (the most significant bit of each coefficient not so far emitted) for each coefficient which has been found significant in the previous significance passes. The subordinate pass is therefore similar to bit-plane coding [3][7]

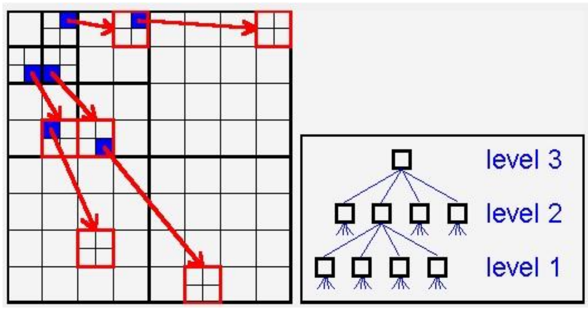


FIG 3. THE DIFFERENT WAVELET COEFFICIENTS AND THE TREE STRUCTURE [3]

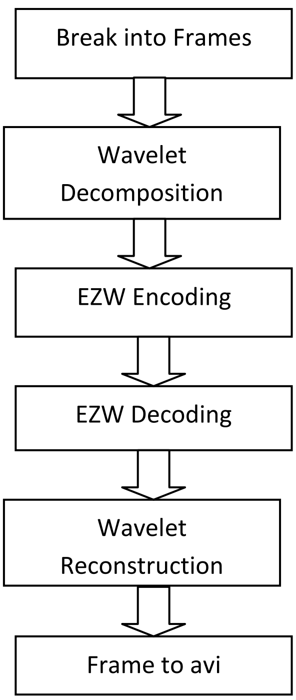
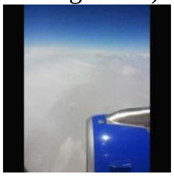


FIG 4. FLOW CHART OF TRANSFORM PROCESS

IV. EXPERIMENTAL RESULTS

Different videos were analysed using EZWT video compression in H.264. Three videos were taken :

1) The **Plane video** (wherein minimal changes in the background)



2) The **News reader** video(in which lip movements showed a variance) [courtesy : Aajtak channel]

and,

3) **Cartoon video** (wherein a lot of motion was present)

All these videos were analyzed in MATLAB in terms of compression factor, PSNR (Peak Signal to Noise ratio) and MSE (Mean Square Error).

A) FOR THRESHOLD OF 50, (LOW DATA RATE)THE RESULTS ARE AS OBTAINED BELOW:

Name of video	M.S.E	PSNR	Input 3 sec	Output 3 sec	Encoding time / frame	Decoding time / frame
Plane Video	18.98 (0.08 bpp)	35.35 dB	757 KB	556 KB (0.73)	50.45 sec	18.32 sec
News Reader	28.50 (0.09 bpp)	33.58 dB	820 KB	699 KB (0.85)	54.37 sec	21.08 sec
cartoon	49 (0.12bpp)	31.14 dB	990 KB	456 KB (0.46)	69.11 sec	25.21 sec

B) FOR THRESHOLD OF 5,(HIGH DATA RATE) THE RESULTS ARE AS OBTAINED BELOW:

Name of video	M.S.E	PSNR	Input 3 sec	Output 3 sec	Encoding time / frame	Decoding time / frame
Plane Video	22.92 0.33bpp	34dB	757KB	590KB (0.78)	156.03sec	53.28sec
News Reader	10.78 0.88bpp	37.81dB	820KB	645KB (0.786)	221.19sec	76.69sec
cartoon	3.42 0.98bpp	42.79dB	990KB	740KB (0.7462)	369.46 sec	131.72 sec

C) FOR THRESHOLD OF 100,(LOWEST DATA RATE) THE RESULTS ARE AS OBTAINED BELOW:

Name of video	M.S.E	PSNR	Input 3 sec	Output 3 sec	Encoding time / frame	Decoding time / frame
Plane Video	47.99 0.04bpp	31.32dB	757KB	418KB (0.553)	31.03sec	12.42sec
News Reader	70.34 0.04bpp	29.66dB	820KB	364KB (0.44)	30.13sec	11.91sec
cartoon	110.41 0.05bpp	27.7dB	990KB	323KB (0.323)	37.87sec	14.54sec

TABLE 2. THE ANALYSIS BASED ON THE 3 VIDEOS.
*the figure shown in bracket corresponds to compression factor.

Fig. 5a

Fig. 5b

FIG. (5A) SHOWS THE RECONSTRUCTED IMAGE CORRESPONDING TO FRAME NO 2 OF A NEWS READER VIDEO, WITH TH = 5 .HAVING PSNR = 37.81DB AND MSE = 10.78 WITH 0.88BPP (BITS PER PIXEL. AND RESOLUTION (512*512)
FIG. (5B) SHOWS THE RECONSTRUCTED IMAGE CORRESPONDING TO FRAME NO 2 OF A NEWS READER VIDEO, WITH TH = 100 HAVING PSNR = 29.66DB AND MSE = 70.34 WITH 0.04BPP (BITS PER PIXEL AND RESOLUTION (512*512).

Figure 5a and 5b above shows the difference between the frame number 2 of the reconstructed frame.

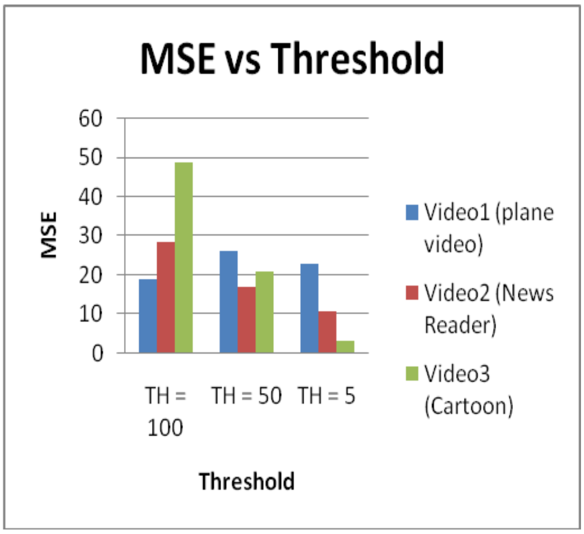


FIG. 6 MEAN SQUARE ERROR (MSE) FOR I-FRAME FOR 3 VIDEOS

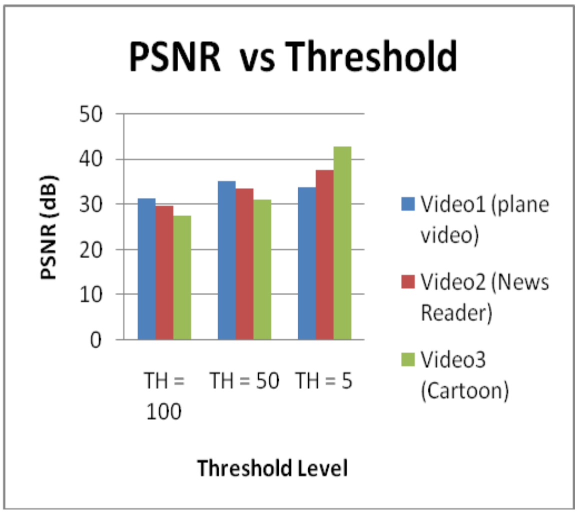


Fig. 7 PSNR VS THRESHOLD FOR I – FRAME FOR 3 VIDEOS

The fig. 6 above, shows the graphs of MSE vs the Threshold level set for the compression. Higher Threshold indicates less iterations and so reduction in picture quality.

Similarly the fig. 7 above, shows the graphs of PSNR vs the Threshold level. Highest PSNR values are observed for lower value of threshold .

V. APPLICATIONS

With the advent of mobile technology the need has arose to reduce size of uncompressed video streams and is obvious when you calculate its data rate, the rate at which its bits and bytes stream over a communication channel. For example, digital video created for the screen resolution of common video cell phones: QCIF (176x144 pixels), the cell phone would have to move 20 megabytes of data from storage to the screen every second that the video is running. This would be beyond the data handling capacity of cell phones. The computing capability of the mobile device being weak, low end mobile phones may not have enough processing power to achieve higher video frame rates. At the lower speeds (15-20 kbps) of 2nd generation networks, a frame rate of 5 or 6 fps should be targeted taking into account the battery life. Even at higher bit rates (50 kbps) most MOBILE users will be able only to play back at 7 or 8 fps. The need for compression is much greater in the mobile space when you consider the common denominator is not a 3G user.

VI. CONCLUSION

It was seen that the highest compression on an average was achieved for the first video i.e the "plane video". As the threshold value was increased from 50 to 100, the PSNR decreased drastically, depicting the reduction in the picture quality. Indirectly the MSE was observed to be increased. For example the MSE for the third video i.e the "cartoon video" increased from 49 to 110 for the increase in the threshold from 50 to 100 respectively. Also the decoding and encoding time for each frame increased by 3 fold corresponding to the threshold value increased from 5 to 100. And his method is being suitable for low bit rate video showing highest compression ratio.

VII. FUTURE WORK

Taking into account the future evolution of multimedia applications, for example, videoconferencing, the recommendation of H.264/MPEG-4 Part 10 is likely to be followed to standardize further transport and storage of MPEG-4 and H.264 coded video data , transport over IP networks . So this project can be developed for android platform using embedded processor which could drastically reduce the 3G charge incurred by the transmission of the data during video conferencing.

REFERENCES

[1] Iain E. G. Richardson, *H.264 ad Mpeg-4 Video Compression, Video Coding for next generation multimedia, The Robert Gordon University, Aberdeen, UK* John Wiley & Sons Ltd, The Atrium, Southern Gate,Chichester, West Sussex PO19 8SQ, England (2003).

[2] Iain Richardson , *Vcodex.com 2007-2013 Vcodex White Paper: An overview of H.264 Advanced Video Coding.*

[3] EZW ENCODING A REALLY FRIENDLY GUIDE TO WAVELETS THE FAST LIFTING WAVELET TRANSFORM : HTTP://WWW.POLYVALENS.COM/BLOG/WAVELETS/EZW/ AS ON 31/08/2014

[4] C.-L. Chang and B. Girod, "Direction-adaptive discrete wavelet transform for image compression," *Image Processing, IEEE Transactions on*, vol. 16, no. 5, pp. 1289–1302, May 2007.
[5] Olivier Rioul and Martin Vetterli, "Wavelets and Signal Processing", *IEEE Trans. on Signal Processing, Vol. 8, Issue 4, pp. 14 - 38 October 1991.*
[6] Jerome M. Shapiro ,"Embedded image coding using zerotrees of wavelet coefficients", *IEEE Transactions on Signal Processing Vol. 41 No. 12 Pg. 3445-3462* .
[7] Wikipedia (EN), http://en.wikipedia.org/wiki/Embedded_Zerotrees_of_Wavelet_transforms AS ON 31/08/2014