

Video Compression with Wavelet Transform Using SPIHT and Arithmetic Coding Technique

Omeshwari Kurre, Mr. Amit Kolhe

Department of Electronics & Telecommunication, Rungta Engineering College, Bhilai (C.G.)

Abstract: Motion estimation is a common element in many video processing algorithms, serving as a valuable way to capture the spatial-temporal correlation in video signals. However, the robustness of motion estimation often suffers from problems such as ambiguities of motion trajectory (i.e. the aperture problem) and illumination variances. In this paper, we explore a new framework for video processing based on the recently proposed wavelet transform with SPIHT algorithm and fast implementations of arithmetic coding, which achieve optimal compression and higher throughput by better exploiting the great numerical capabilities of the current processors. Also, Instead of containing an explicit motion estimation step, the wavelet transform provides motion-selective sub band decomposition for video signals. We proposed a new and different approach for based on set partitioning in hierarchical trees (SPHIT) and arithmetic coding which provides better performance than the previous approaches. The result is calculated by comparing the actual video size and compressed video size, after decoding using proposed technique.

Keywords-Motion Estimation, Wavelet Transform, SPIHT, Arithmetic Coding,

I. INTRODUCTION

Motion information provides a powerful means of improving a variety of video processing systems. Many algorithms have been developed during the last thirty years for applications such as motion compensated (MC) *filtering* for noise reduction, MC *prediction* for coding and MC *interpolation* for video format conversion. The predictive coding area is the least demanding, and the first that led to widespread use of motion estimation (ME), as it is sufficient for this application that motion compensation decreases the average prediction error. Video format conversion is one of the most demanding, as it requires estimation of the *true motion* over a large range for picture rate conversion, and with sub-pixel accuracy for MC de-interlacing. Nevertheless, the last decade has brought us two generations of application-specific ICs1 (ASICs) [1, 2, 3] for MC video format conversion (MC-VFC) in consumer TV. Recently, real time DSP software for MC-VFC has also been demonstrated [4]. All these designs incorporate ME with a quality that required substantial breakthroughs.

In this paper, we explore a new framework for video processing based on the recently proposed wavelet transform [5]. Instead of containing an explicit motion estimation step, the wavelet transform provides motion-selective sub band decomposition for video signals. Similar approaches to video processing using various motion-selective 3-D transforms have been previously proposed by several researchers [6]–[8]. A potential advantage of the wavelet transform is that its directional resolution can be refined by invoking more levels of decomposition.

In our prior work this technique was used to improve the coding efficiency of a hybrid video coding scheme, which is based on wavelet-based residual coding [10] and SPIHT [9]. In this paper, we demonstrate that the arithmetic coding approach along with set partitioning in hierarchical tree techniques to improve compression ratio. Adaptation is best suited for the raw video of .cif and .qcif format.

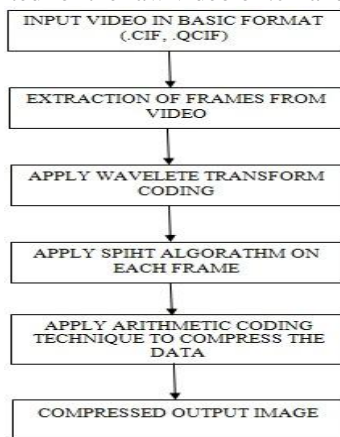


Figure 1.1 Basic steps of Video Compression using proposed method.

II. METHODOLOGY

The compression task is performed in following three steps:

STEP1: APPLY WAVELET TRANSFORM CODING:

First a wavelet transform is applied. This produces as many coefficients as there are pixels in the frame (there is no compression yet since it is only a transform). These coefficients can then be compressed more easily because the information is statistically concentrated in just a few coefficients. This principle is called transform coding. After that, the coefficients are quantized and the quantized values are entropy encoded. Here, Wavelet compression perform in following steps

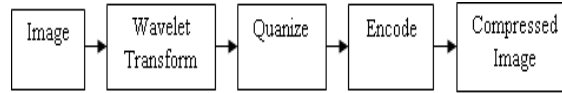


Figure 1.2 Wavelet transform based compression.

Similarly for wavelet decomposition following steps are used-

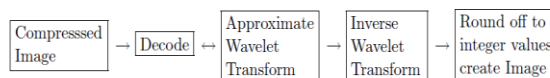


Figure 1.3 Wavelet transform based decomposition.

STEP2: SPIHT COMPRESSION:

The main algorithm works by partitioning the wavelet decomposed image into significant and insignificant partitions based on the following function:

$$S_n(T) = \begin{cases} 1, & \max_{(i,j) \in T} \{|c_{i,j}|\} \geq 2^n \\ 0, & \text{Otherwise} \end{cases} \dots\dots (i)$$

Where $S_n(T)$, is the significance of a set of co-ordinates T, and $c_{i,j}$ is the coefficient value at coordinate (i,j) . There are two passes in the algorithm - the sorting pass and the refinement pass. The sorting pass is performed on the list of insignificant sets (LIS), list of insignificant pixels (LIP) and the list of significant pixels (LSP). The LIP and LSP consist of nodes that contain single pixels, while the LIS contains nodes that have descendants. The maximum number of bits required to represent the largest coefficient in the spatial orientation tree is obtained and designated as n_{max} , which is-

$$n_{max} = \lceil \log_2(\max_{i,j} \{|c_{i,j}|\}) \rceil \dots\dots(ii)$$

During the sorting pass, those co-ordinates of the pixels which remain in the LIP are tested for significance by using eqn. 2. The result, $S_n(T)$, is sent to the output. Those that are significant will be transferred to the LSP as well as having their sign bit output. Sets in the LIS (which consists of nodes with descendants will also have their significance tested and, if found to be significant, will be removed and partitioned into subsets. Subsets with a single coefficient and found to be significant will be added to the LSP, or else they will be added to the LIP. During the refinement pass, the nth most significant bit of the coefficients in the LSP is output. The value of n is decreased by 1 and the sorting and refinement passes are repeated. This continues until either the desired rate is reached or $n = 0$, and all the nodes in the LSP have all their bits output. The latter case will result in almost perfect reconstruction as all the coefficients are processed completely. The bit rate can be controlled precisely in the SPIHT [14] algorithm because the output produced is in single bits and the algorithm can be terminated at any time. The decoding process follows the encoding exactly and is almost symmetrical in terms of processing time.

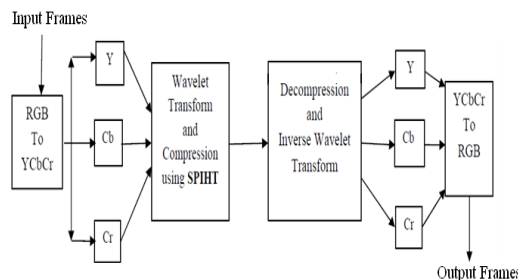


Figure 1.4 Steps of SPIHT based compression

STEP3: ARITHMETIC CODING:

Arithmetic coding assigns a sequence of bits to a message, a string of symbols. Arithmetic coding can treat the whole symbols in a list or in a message as one unit [11]. Unlike Huffman coding, arithmetic coding doesn't use a discrete number of bits for each. The number of bits used to encode each symbol varies according to the probability assigned to that symbol. Low probability symbols use many bit, high probability symbols use fewer bits [12]. The main idea behind Arithmetic coding is to assign each symbol an interval. Starting with the interval [0...1), each interval is divided in several subinterval, which its sizes are proportional to the current probability of the corresponding symbols [13]. The subinterval from the coded symbol is then taken as the interval for the next symbol. The output is the interval of the last symbol [1, 3].

Arithmetic coding algorithm is shown in the following figure:

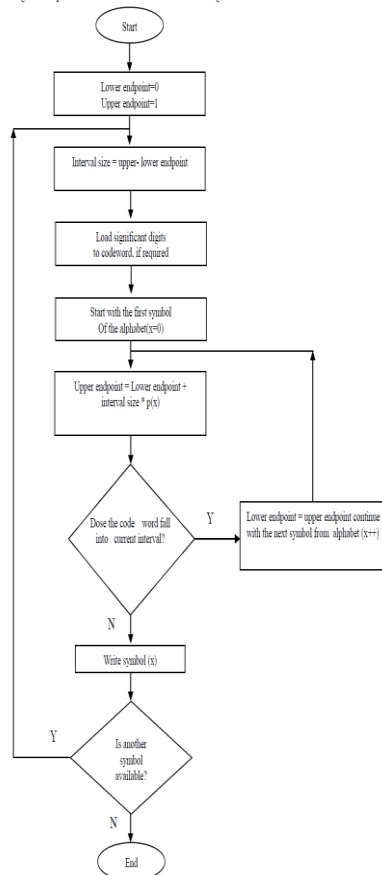


Figure 1.5 Flow chart of Arithmetic encoding technique used.

III. RESULT ANALYSIS

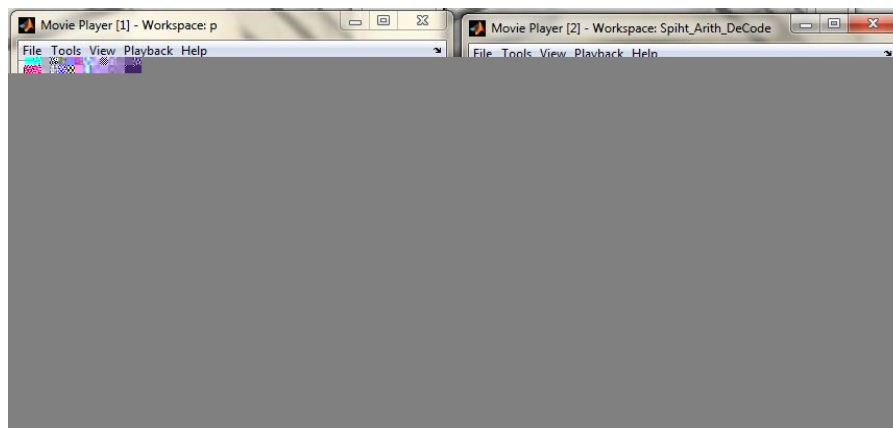


Figure 1.6 Input video and output Compressed video after applying proposed method.

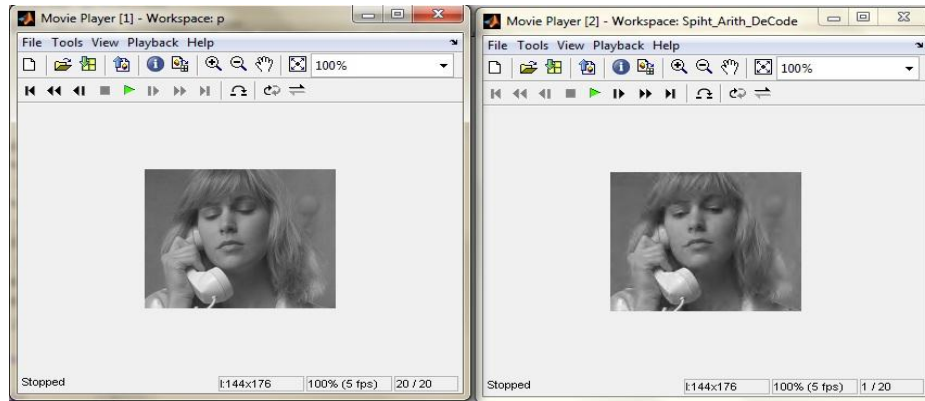


Figure1.7 Input video and output Compressed video after applying proposed method

Raw video	No. of frames compressed	Size (in KB) /frame (actual video)	Size (in KB)/frame (Compressed video)	Compression (In %)
Foerman_qcif	20	37.12 KB	1.45 KB	96.093
Akiyo_qcif	20	37.12 KB	1.45 KB	96.093
Suzie_qcif	20	37.12 KB	1.45KB	96.093

Table 1.1 Compression Analyses

IV. Conclusion

Here, we compress the first 20 frames of input video and figure 1.6 and figure 1.7 shows the input video and output video after compression. From table 1.1 it can be observed that from proposed algorithm compression may be achieved up to 96.093%.

REFERENCES

- [1] G. de Haan et al., "IC for motion compensated 100 Hz TV, with a smooth motion movie-mode", *IEEE Trans. CE*, May '96, pp. 165-174.
- [2] G. de Haan, "IC for motion compensated deinterlacing, noise reduction, and picture rate conversion", *IEEE Trans. CE*, Aug '99, pp. 617-624.
- [3] M. Shu et al., "System-on-silicon for motion compensated scan-rate conversion, picture-in-picture processing, split-screen applications and display processing", *IEEE Trans. CE*, Aug. '99, pp. 842-850.
- [4] R. Schutten and G. de Haan, "Real-time 2-3 pull down elimination applying motion estimation/ compensation on a programmable device", *IEEE Trans. CE*, Aug. '98, pp. 930-938.
- [5] Wim Sweldens, "The lifting scheme: A Construction Of Second Generation Wavelets (1997).
- [6] E. Chang and A. Zakhor, "Subband video coding based on velocity filters," in Proc. IEEE International Symposium on Circuits and Systems, May 1992.
- [7] F. A. Mujica, J.-P. Leduc, R. Murenzi, and M. J. T. Smith, "A new motion parameter estimation algorithm based on the continuous wavelet transform," *IEEE Trans. Image Proc.*, vol. 9, no. 5, pp. 873-888, May 2000.
- [8] I. W. Selesnick and K. Y. Li, "Video denoising using 2D and 3D dual-tree complex wavelet transforms," in *Proc. of SPIE conference on Wavelet Applications in Signal and Image Processing X*, San Diego, USA, August 2003.
- [9] Amir Said, William A. Pearlman "A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees"
- [10] G. Heising, D. Marpe, H.L. Cycon, and A.P. Petukhov "Wavelet- Based Very Low Bit-Rate Video Coding Using Image Warping and Overlapped Motion Compensation",
- [11] Rissanen, J. J. and Langdon, G. G.: 'Arithmetic Coding'. IBM Journal of Research and Development, 23(2):146-162, March 1979.
- [12] Redmill, D. W. and Bull, D. R.: 'Error Resilient Arithmetic Coding of Still Images'. Image Communications Group, Centre for Communications Research, University of Bristol, Bristol.
- [13] Kavitha, V. and Easwarakumar, K. S. : 'Enhancing Privacy in Arithmetic Coding'. ICGST-AIML Journal, Volume 8, Issue I, June 2008.
- [14] Haoming Wang; Xiaozhong Pan; Eng. Coll. of the APF, Xi'an, China, "Video compression coding based on the improved 3D-SPIHT", Computer Application and System Modeling (ICCSAM), 2010 International Conference, Oct. 2010.