

MATLAB implementation of EPZS motion estimation in h.264 AVC

Abstract

In real time applications such as video streaming, it is important that the video encoding/decoding is fast. In video compression technique, most of the complexity comes from the H.264 encoder specifically in the motion estimation (ME) part. Enhanced Predictive Zonal Search (EPZS) is one of the best ME algorithms. In this paper a MATLAB implementation of EPZS algorithm is introduced to provide MATLAB environment for EPZS ME algorithm. The results of the proposed architecture for EPZS were compared with that of full search (FS) motion estimation algorithm for validation. A comparative study between EPZS, FS and the most popular ME algorithms is also performed. The simulation results show that EPZS is better than other fast search algorithms in terms of quality and speed.

Keywords: simulation results, video compression technique, motion compensation, motion compensated predictor, compression technique, comparative study

Volume I Issue I - 2017

Nahed Ali Bahran, Abdelhalim Zekry
Ain Shams University, Egypt

Correspondence: Nahed Ali Bahran, Department of Electronics and Communications, Ain Shams University, Egypt, Email albahran12@gmail.com

Received: June 14, 2017 | **Published:** June 28, 2017

Abbreviations: EPZS, enhanced predictive zonal search; ME, motion estimation; FS, full search; MC, motion compensation; BMA, block matching algorithm; SA, search area; SAD, sum of absolute difference; AVC, advanced video coder; DS, diamond search; ES, exhaustive search; MVFAST, motion vector field adaptive search technique; PMVFAST, predictive motion vector field adaptive search technique; PSNR, peak signal to noise ratio; EPZS, enhanced predictive zonal search

Introduction

Digital video compression is an important technique used to transmit or store digital videos in a limited bandwidth environments or storage areas. H.264/AVC standard^{1,2} is one of the modern video coding standards. Compared to past standards, it provides the best balance between the coding efficiency, implementation complexity and cost. Video compression technique depends mainly on the prediction mechanism. Prediction technique in H.264/AVC can be classified into: intra and inter prediction methods. Intra prediction removes spatial redundancies within the frames whereas inter prediction removes temporal redundancies between frames. Inter prediction can be divided into two main processes which is Motion Estimation (ME) and Motion compensation (MC) processes. ME is computationally extensive and consumes about 70% of the total time required for video encoder.

The most common ME technique is called Block Matching Algorithm (BMA) where the current frame is divided into Macro blocks (MBs). In BMA each MB in the current frame will be compared with the corresponding MBs in the Search Area (SA) of the reference frame to find the best match MB in the reference frame as shown in Figure 1. Once the best match block for the block in the current frame is located in the reference frame, the motion vector is calculated as the displacement between these two macro blocks. This best candidate will be the motion compensated predictor of the current MB and is subtracted from the current frame, the resulting error is called residual frame. This is called the Motion Compensation (MC) process. The motion vectors and residual frame are adequate information to be sent by H.264 AVC encoder. This information, in general, requires

significantly fewer bits than the direct coding of the original frame, and hence improving coding efficiency. The most popular matching criterion in Advanced Video Coder (AVC) is called Sum of Absolute Difference (SAD). The SAD of two $N \times N$ MBs is given by equation (1) as follows:

$$SAD = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |c_{ij} - R_{ij}| \quad (1)$$

Where: c_{ij} is a pixel of the current MB and R_{ij} is a pixel of the reference area.

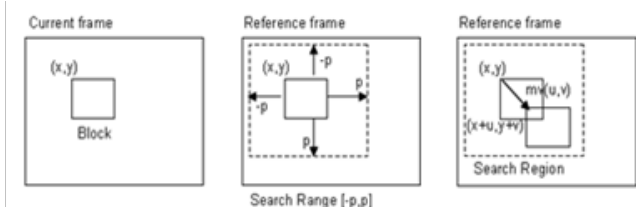


Figure 1 Block Matching Algorithm.

The basic BMA is called Full Search (FS) or Exhaustive Search (ES). Where all the candidates of the SA are considered in the search of the best match predictor. For large SA, FS is computationally extensive and cannot be effective for real time applications, so other suboptimal and fast ME algorithms such as Three Step Search (TSS)³ and Diamond Search (DS)³ have been introduced. These fast search algorithms tend to compromise the coding efficiency with the speed. For this reason, these algorithms may not be suitable for real time applications. As a result of the recent advent of the H.264/AVC standard, data-adaptive ME algorithms have been presented such as Motion Vector Field Adaptive Search Technique (MVFAST),⁴ Predictive Motion Vector Field Adaptive Search Technique (PMVFAST)⁵ and the Enhanced Predictive Zonal Search (EPZS).⁶ EPZS algorithm makes an improvement in the motion vector search method and its performance is better than PMVFAST or APDZS.

Several studies have been made on the software implementations of inter prediction for H.264 AVC. Previous comparative studies on

various ME algorithms have been conducted. Those studies presents the superiority of some ME algorithms than others.⁷⁻⁹ From JM reference software jm18.6, EPZS ME algorithm achieves the optimum results in terms of quality, speed, and data rate.¹⁰ In this paper a MATLAB design and implementation of EPZS ME algorithm for H.264 AVC is presented. An efficient implementation of inter prediction process using EPZS algorithm as the motion estimation technique is realized. The resulting motion vectors and Peak Signal to Noise Ratio (PSNR) are compared with that of the exhaustive search ME algorithm to verify its efficiency. The proposed implementation is used to compare the EPZS algorithm with the most probable ME algorithms. Also this implementation provides a MATLAB environment for further research and developments in EPZS ME algorithm. The paper is organized as follows: motion estimation algorithms are presented in Section 2. EPZS ME algorithm is then introduced in Section 3. The proposed hardware implementation architecture of EPZS ME is described in Section 4. The implementation and simulation results are discussed in Section 5. Finally, the conclusions are presented in the last section.

Enhanced predictive zonal search motion estimation

In this section we shall introduce the theory behind EPZS. EPZS ME algorithm depends on three essential processes: Predictor selection, adaptive early termination and motion vector refinement.

Predictor selection: Predictor selection is the process of selecting the most probable MV for the current block from the previous motion vectors of the blocks in the search area. This is the most important feature of EPZS algorithm. Instead of examining all the possible points in the search area, only a set of candidate MVs has been selected. These candidate MVs or predictors are divided into 3 sets: A, B and C.

- Set A is the median MV. It is the median of the motion vectors of the three adjacent blocks on the left, top and top-right from the current block as shown in Figure 2a.
- Set B includes the (0,0) motion vector and the three spatially adjacent motion vectors in the current frame as shown in Figure 2b.
- Set C contains the motion vector of the collocated block in the previous frame and the MVs of its four adjacent blocks as illustrated in Figure 3.

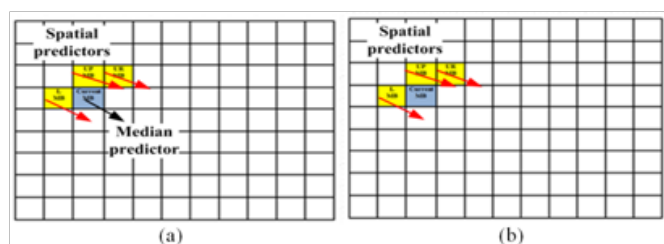


Figure 2 Spatial Predictors For The EPZS Algorithm In The Current Frame, (A) Median MV Of Set A, (B) The Spatial Predictors of Set B.

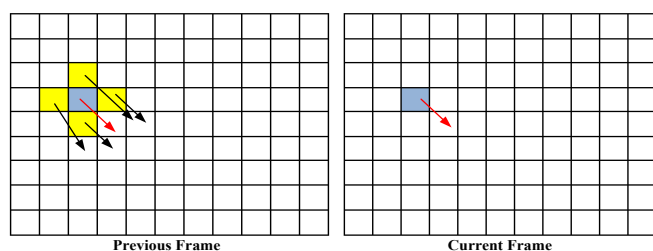


Figure 3 Motion vectors of the collocated MB and its surrounding blocks in the previous frame.

Adaptive early termination: The distortion of adjacent blocks tends to be highly correlated exactly like the MVs. According to this fact, a threshold for the early termination is calculated using the minimum SAD value of the three adjacent blocks in the current frame shown in Figure 2 and the collocated block in the previous frame. The adaptive threshold T_k is given by the following equation:¹¹

$$T_k = a_k \times \min(MSAD_1, MSAD_2, \dots, MSAD_n) + b_k$$

Where: a_k and b_k are fixed, MSAD is the minimum SAD value for each of the three spatial predictors in the current frame shown in (Figure 2b) and the collocated MB in the previous frame. At first, all the predictors of set A (median MV) are examined. If its SAD is smaller than the threshold ($T_1=256$), the ME process may terminate immediately and the median MV is selected as the final MV. Otherwise the algorithm continues to use set B. After examining all the predictors of set B, if the minimum SAD satisfies the threshold T_2 , the algorithm will be terminated. Otherwise it continues to use set C. Similarly, when all the predictors of set C are examined, and if the minimum SAD satisfies the threshold ($T_3=T_2$), the algorithm will be terminated. Otherwise it continues to the refinement pattern as shown in the following sub-section.

Motion vector refinement: If the early termination criteria are not satisfied, the best predictor of the above sets A, B, and C (which has minimum SAD value) can be additionally refined. This best predictor will be chosen as a center of the refinement pattern. The refinement pattern can be either a small diamond pattern or a square pattern (referred to here by EPZS²) as depicted in (Figure 4). The search continues until the best predictor located at the center of the search pattern is reached.

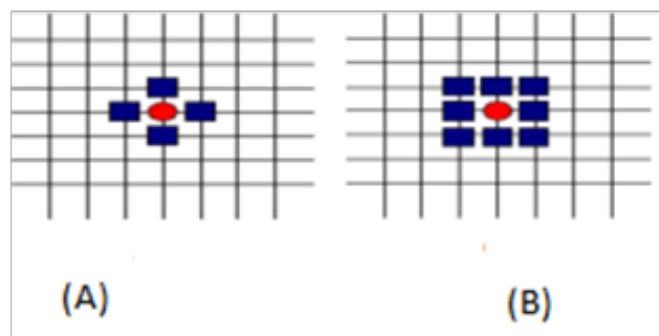


Figure 4 Search patterns used in EPZS, (A) small diamond pattern, (B) square pattern.

Matlab implementation

In this section the block diagram of the MATLAB implementation of the proposed H.264 inter prediction technique is presented as shown in Figure 5. The ME algorithm used is the EPZS algorithm and it is implemented according to the flow chart given by Figure 6.

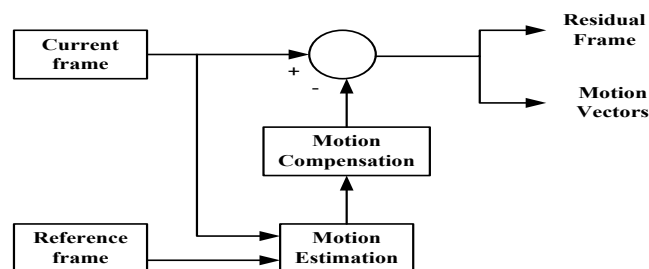


Figure 5 The block diagram of the implemented inters prediction technique.

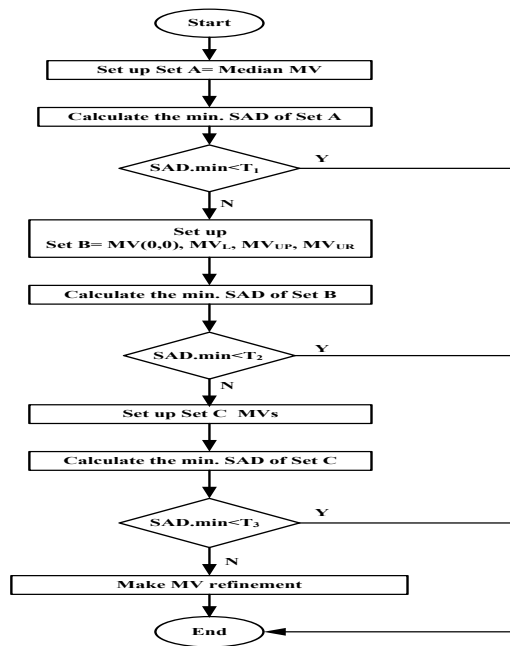


Figure 6 EPZS ME Algorithm Flow Chart.

Simulation results

The experiments were carried out on a CORE i7 Intel machine with 8 GB RAM memory. The environment for comparing the performances is MATLAB software. A distance of 2 between current frame and reference frame was used to generate the frame-by-frame results of the algorithms and a search window (SW) of 32. Figure 7 shows an example of the plot of current frame, reference frame, motion compensated reference frame and motion compensated residual frame. The simulation is carried out for various video sequences such as: Foreman, Stefan and Akaio video sequences of CIF and QCIF format with frame rate of 30 fps. Since, FS algorithm is the most accurate ME estimation algorithm, the results of the implemented inter prediction

process using EPZS motion estimation algorithm has been compared with that of FS motion estimation process. Table 1 shows the output PSNR of the motion compensated frames generated by the ME algorithms and the average no. of search points per MB of the EPZS, FS and some fast search algorithms such as Three Step Search (TSS),² New Three Step Search (NTSS),¹² 4-Step Search (4SS),¹³ Diamond Search (DS)³ and Adaptive Road Pattern Search (ARPS).¹⁴

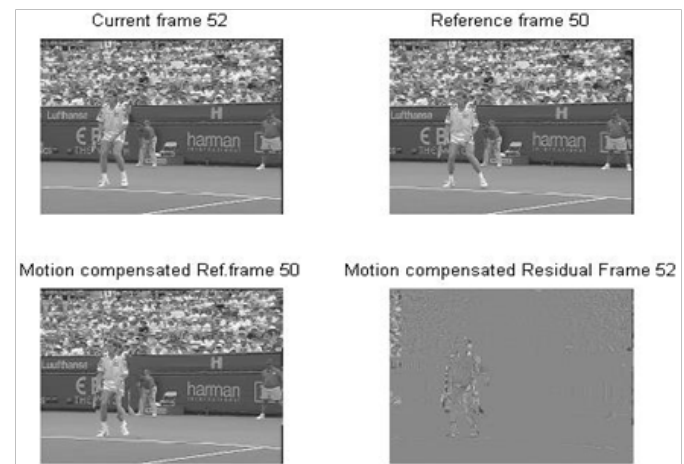


Figure 7 Pictures of the current, the reference, and the motion compensated residual Frame.

From Table 1, one can notice that EPZS ME algorithm requires the minimum average no. of search points with a slight degradation in PSNR values. This means reducing the complexity of the video encoder and minimizing the time required to achieve video encoding technique without affecting the quality of the video. Previous comparative studies on various ME algorithms demonstrates that ARPS greatly reduces the number of searching points in comparison to others^{7,8} with a small reduction in PSNR. Another study on the performance of EPZS over FS and DS is given UR Bala.⁹ This study exhibits that EPZS has a better performance than DS in terms no. of computations and PSNR and has a PSNR smaller than that of FS by less than 1%.

Table 1 Comparison of the PSNR and No. of Search Points of EPZS and other ME Algorithms

Video sequences ME algorithms	Foreman QCIF		Akaio QCIF		Stefan CIF	
	PSNR	Computations	PSNR	Computations	PSNR	Computations
EPZS	25.6969	14.5408	38.6632	3.051	25.2295	7.0861
FS	26.0698	886.0101	38.6632	886.0101	25.5755	984.9192
TSS	24.1439	29.3131	38.6632	28.3131	25.2177	30.9293
NTSS	24.1157	34.5152	38.6632	14.8687	25.0231	35.3333
4SS	20.0052	22.3232	38.6632	14.6832	20.6187	20.158
DS	22.6815	37.8283	38.6632	11.5354	20.158	23.0631
ARPS	24.4228	17.0707	38.6632	5.1616	24.2563	10.5227

Conclusion

This paper is devoted to the MATLAB implementation of the EPZS motion estimation algorithm for sake of quantitative performance comparison with other motion estimation algorithms. Also, such implementation can be used in verifying the other hardware implementations such as VHDL on field programmable gate arrays FPGA.¹⁵ This paper is localized on EPZS motion estimation

algorithm. The simulation results point out the advancement of EPZS versus FS and some other fast search algorithms. The comparative study presented in this paper demonstrates that EPZS is superior on other ME algorithms such as DS and ARPS. EPZS ME algorithm has the smallest no. of search points; i.e has the least encoder complexity and the smallest time required for video encoder. The degradation in PSNR from FS is not more than 1.5%.

Acknowledgements

None.

Conflict of interest

Author declares that there is no conflict of interest.

References

1. Advanced video coding for generic audiovisual services (ITU-T Rec. H.264). *Joint Video Team (JVT) USA*. 2012.
2. T Koga, K Iinuma, A Hirano, et al. Motion compensated inters frame coding for video conferencing. *Proc Nat Telecommun*. 2011;2(1).
3. S Zhu, KK Ma. A new diamond search algorithm for fast block matching motion estimation. *IEEE Transactions on Image Processing*. 997;1(12):292–296.
4. PI Hosur, KK Ma. Motion Vector Field Adaptive Fast Motion Estimation. *In Second International Conference on Information, Communications and Signal Processing (ICICS 99) 7-10 Singapore*. 1999.
5. Alexis M Tourapis, Oscar C Au, Ming L Liou. Predictive Motion Vector Field Adaptive Search Technique (PMVFAST)-Enhancing Block Based Motion Estimation. *In proceedings of Visual Communications and Image Processing*. 2001;1(10):888–892.
6. AM Tourapis. Enhanced Predictive Zonal Search for Single and Multiple Frame Motion Estimation. *Visual Communications and Image Processing*. 2002;4671:1069–1079.
7. HA Choudhury, M Saikia. Comparative Study of Block Matching Algorithms for Motion Estimation. *International Journal of Advanced Computational Engineering and Networking*. 2013;1(10):2320–2106.
8. R Raghava, AK Sharma. Matlab Based Motion Estimation And Compression In Video Frames Using True Motion Tracker. *International Journal of Electronics and Communication Engineering & Technology (IJECEET)*. 2014;5(3):34–42.
9. UR Bala, SA Alice, BM Jebalincy, et al. Dynamic Video Compression Using Idle Scene Skipper. *International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)*. 2016;3(19).
10. Joint Video Team Reference software JM 18.6.
11. AM Tourapis. Highly efficient predictive zonal algorithms for fast block-matching motion estimation. *IEEE Transactions on Circuits and Systems for Video Technology*. 2002;12(10):934–937.
12. R Li, B Zeng, ML Liou. A new three step search algorithm for block motion estimation. *IEEE Transactions on Circuits and Systems for Video Technology*. 1994;4(4):438–442.
13. LM Po, WC Ma. A novel four-step search algorithm for fast block motion estimation. *IEEE Transactions on Circuits and Systems for Video Technology*. 1996;6(3):313–317.
14. Nie Y, Ma KK. Adaptive Rood Pattern Search for Fast Block-Matching Motion Estimation. *IEEE Trans Image Process*. 2002;11(12):1442–1449.
15. NA Bahran, A Zekry, RA Fathy, et al. Novel FPGA Implementation of EPZS Motion Estimation in H.264 AVC. *Recent Trends in Information and Communication Technology*. 2018.