BLOCK INTER MODE DECISION FOR FAST ENCODING OF H.264

D. Wu, S. Wu, K. P. Lim, F. Pan, Z. G. Li, X. Lin

Institute for Infocomm Research(I^2R)
Agency for Science Technology And Research (A*STAR)
21 Heng Mui Keng Terrace, Singapore 119613
email: {djwu,swu,kplim,efpan,ezgli,linxiao}@i2r.a-star.edu.sg

ABSTRACT

This paper presents a fast block INTER mode decision algorithm to significantly improve the time efficiency of the encoder in H.264. It makes use of the spatial homogeneity of video object's textures and temporal stationarity characteristics inherent in video sequences. Specifically, homogeneity decision of a block is based on edge information, and MB differencing is used to judge whether the MB is time-stationary. Based on the above analysis, only parts of inter prediction modes are chosen for RDO calculation. The experiment results show that the new scheme is able to achieve a reduction of 30% encoding time on average, with a negligible average PSNR loss of only 0.03 dB and a mere 0.6% bit rate increase compared with the original H.264 reference software.

1. INTRODUCTION

Currently, a new standard for coding natural video pictures known as H.264 is being finalized. The experimental results have shown that H.264 greatly outperform existing video coding standards in terms of both PSNR and visual quality [1]. This is due to the new techniques, such as spatial prediction in intra coded blocks, integer transform, variable block size motion estimation/compensation, multiple reference frame motion estimation/compensation, loop filter and context-based adaptive binary arithmetic coding (CABAC), etc. used in the standard. Among these techniques, rate distortion optimization (RDO) is one of the essential parts of the whole encoder to achieve the much better coding performance in terms of minimizing compressed video data bits and maximizing coding quality. In RDO, the encoder tries all possible mode combinations such as different block sizes for intra prediction, inter-frame motion compensation, multiple-reference frames in the case of inter modes and chooses the best one in terms of least RDO cost. This requires much computational resources even with regard to state-of-the-art hardware technology. Thus, techniques which can speed up H.264 encoding while maintaining the reconstructed video quality will be very useful for practical H.264 real-time implementation.

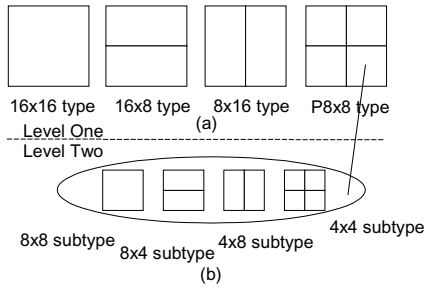
Up to date, a number of efforts have been made to explore the fast algorithms in motion estimation for H.264 video coding [2, 3]. Instead of trying every search point in the search window, they aim to do motion estimation using only a small portion of points. Since motion estimation is very time consuming, these techniques achieve much better performance in terms of time savings without much loss of video quality or increase of bit-rates. A fast algorithm in intra prediction for H.264 has also been proposed [4]. In the proposal, the authors use the local edge information to reduce the amount of calculations in intra prediction. With the use of edge direction histogram derived from the edge map of the picture, only a small number of most probable intra prediction modes are chosen for RDO calculation. Therefore the fast mode decision algorithm can increase the speed of intra coding greatly.

In H.264, blocks with different sizes (16x16, 16x8, 8x16, 8x8, 8x4, 4x8 and 4x4) are used in inter-frame motion compensation. For each macroblock, when RDO optimization is used, all the sizes are tried before a final decision of block size is made in the end. This "try all and select the best" philosophy is optimal in terms of bit rate reduction, but at the cost of high computational complexity. However, we observed that this is redundant especially in cases when a large portion of homogeneous regions exists in video sequences. Besides that, many natural sequences contain moving objects with a stationary background. These two types of regions, namely homogeneous and/or stationary regions, that exist in video sequences are mostly encoded in big block size such as 16x16 block size. This prompts us to propose a method in which if a 16x16 block is homogeneous and/or stationary, computations on most of the other smaller block sizes are skipped . Edge information is used to judge the homogeneity of a 16x16 or 8x8 block, and MB (16x16 block) differencing is adopted to decide the stationarity of a 16x16 block. The approach proposed is capable of reducing up to 45% of total encoding time with negligible decrease in video quality or increase in bit-rates. This algorithm has been adopted as part of the reference model for H.264 [5]. The rest of the paper is organized as follows. Section 2 gives an overview and analysis of inter coding in H.264. Section 3 presents in detail the methods used in fast mode decision for inter prediction. Experimental results are presented in section 4 and conclusions are given in section 5.

2. OVERVIEW OF INTER MODE DECISION

2.1. Inter Mode Decision in H.264

As specified in the documents of H.264, there are conceptually 7 different block sizes (16x16, 16x8, 8x16, 8x8, 8x4, 4x8 and 4x4) that can be used in inter-frame motion estimation/compensation. These different block sizes actually form a one or two level hierarchy inside a macroblock. Comprising only the first level, the block size can be 16x16, 16x8, or 8x16. In the case of two levels, the marcoblock is specified as P8x8 type, of which each 8x8 block can be one of the subtypes such as 8x8, 8x4, 4x8 or 4x4. The four macroblock type sizes and four macroblock subtype sizes are shown in Fig. 1.



(a) Sizes for a macroblock type (b) Sizes for a macroblock subtype in P8x8 mode

Fig 1. Different block sizes in a macroblock

Currently, by trying all the possible block sizes, motion estimation and RDO are performed to find the best block sizes in the macroblock, resulting in very heavy computational load at the encoder. The best block size is determined by finding the one that gives the best rate-distortion performance as shown in Fig. 2.

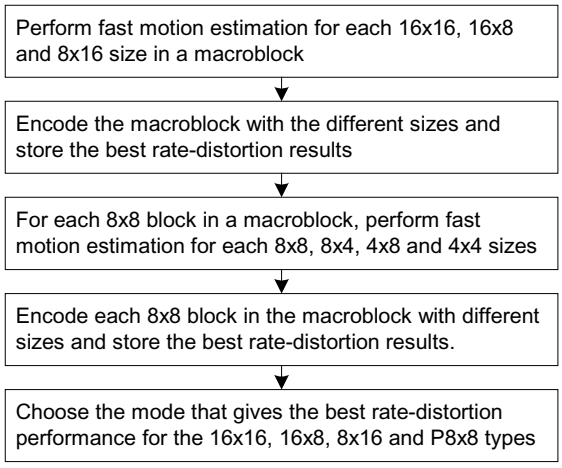


Fig 2. Best INTER mode determination in H.264

In the current practice, for each position in the search window, a large number of motion estimation are performed to find the motion vector for each variable-sized block, and only the motion vectors that perform the best RDO result are used while the rest are discarded at the end. It is obvious that the approach is a waste of computational resources.

2.2. Motivation

It is observed that when video objects move, the various parts of the video objects move together [6][7]. One of the main reasons for using different block sizes in H.264 is to represent motion of video objects more accurately. Since homogeneous regions tend to move together, homogeneous blocks in the frame should have similar motion and should not be split into smaller blocks. Therefore significant time savings could be achieved for the motion estimation and RDO computations if the block size is accurately predicted earlier. Experiments on various video sequences justify this observation. Fig 3 shows a typical frame of the QCIF sequence 'News'. Overlaid white boxes on the image represent the different block modes that are selected after H.264 encoding. It shows that some regions are coded in smaller size blocks and others in bigger size blocks when there are smooth regions.

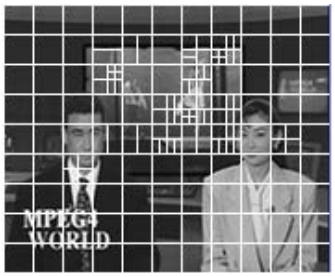


Fig 3. Best INTER mode determination in H.264

From Fig. 3, it is easy to see that the homogenous areas such as the background, black suit of the man are encoded using 16x16 block sizes. On the other hand, the boundary area of white suite of the lady is non-homogenous but contains some strong edges. Due to the temporal stationarity (the object remains still during some time interval), that area is also encoded using 16x16 block size. Since the dancers in the upper part of the image are relatively smaller and contain much motion, they are coded in smaller size blocks. Therefore, the spatial homogeneous areas and temporal stationary areas are good indication to choose an optimal block size in the process of motion estimation, and we could make use of this observation to skip unnecessary block type mode trials in order to reduce the time complexity of H.264.

3. HOMOGENITY AND STATIONARITY REGIONS DETERMINATION

3.1. Homogeneous Regions Determination

There exist many techniques for determining homogeneous regions in an image [8][9][10]. A region is homogeneous if the textures in the region have very similar spatial property. The simplest method is to use statistical measurement such as standard deviation, variance, skewness and kurtosis [8]. In [9], texture is modeled using Gaussian Markov Random Field. The different textures are labeled separately using a hypothesis-and-test-based method on variable window sizes of the textures. This technique is very effective but is computationally intensive. Therefore, the ideal technique chosen should be able to detect homogeneous regions effectively while at the same time must also have low time-complexity. An effective way of determining homogeneous regions is to use the edge information, as the video object boundary usually exhibits strong edges. Because the edge detection has already been performed in the fast INTRA mode decision algorithm, there will be very little computation required [4].

An edge map is created for each frame in [4] using Sobel operator. For a pixel at position (*i, j*) with value $v_{i,j}$, $i \in 1,2,...,R, j \in 1,2,...,C$,in an image frame, the edge vector, $\vec{E}_{i,j} = \{Ex_{i,j}, Ey_{i,j}\}$, is computed as follows:

$$Ex_{i,j} = v_{i-1,j+1} + 2 \times v_{i,j+1} + v_{i+1,j+1} - v_{i-1,j-1} - 2 \times v_{i,j-1} - v_{i+1,j-1} \quad (1)$$

$$Ey_{i,j} = v_{i+1,j-1} + 2 \times v_{i+1,j} + v_{i+1,j+1} - v_{i-1,j-1} - 2 \times v_{i-1,j} - v_{i-1,j+1} \quad (2)$$

where $Ex_{i,j}$ and $Ey_{i,j}$ represent the degree of difference in vertical

and horizontal directions respectively. The amplitude of the edge vector is computed by,

$$Amp\ (\vec{E}_{i,j}) = \left|Ex_{i,j}\right| + \left|Ey_{i,j}\right|. \quad (3)$$

Homogeneity of a block with size NxN, where N is 16 or 8, is determined by using the amplitude of the edge vector in the block using Equation (3). If the sum of the magnitude of the edge vectors at all pixel locations in the block is less than Thd_H, it is classified as homogeneous block, otherwise, it is non-homogeneous. The block homogeneity threshold Thd_H is a preset parameter. If r and c refers to the index of the row and column of the block $B_{c,r}$, the block homogeneity measure $H_{c,r}$ is set to value as follows:

$$H_{r,c} = \begin{cases} 1 & \sum_{i,j \in NxN \text{ block}} Amp \quad (\vec{E}_{i,j}) < Thd_H \\ 0 & \sum_{i,j \in NxN \text{ block}} Amp \quad (\vec{E}_{i,j}) \geq Thd_H \end{cases} \quad (4)$$

where $H_{c,r} = 1$ indicates that the NxN block $B_{c,r}$, is homogeneous and is non-homogeneous if $H_{c,r} = 0$. It must be emphasized that the edge amplitude computation is already done prior to fast INTRA mode decision and the only additional task at this stage is the addition operations in Equation (4).

3.2. Stationary Regions Determination

Compared to spatial homogeneity inside a single frame, stationarity refers to the "stillness" between neighboring frames temporally. Through the analysis of H.264 performance on different video sequences, we found that for some sequences, even though not many homogenous regions exist, the background or the part of the image remains almost stationary and the block size used is still 16x16 after RDO computations. Thus, we can use MB difference (a variant to frame difference) to first judge if this MB changes or not. If there are little changes between consecutive frames, the MB is classified as stationary and 16x16 mode is used for motion estimation, and all the other modes are skipped. Although this MB difference could be done on a pixel basis, decimated pattern can also be used to reduce time complexity. We use the 1:16 decimation pattern in the calculation of the MB difference algorithm since it achieves a better balance between speed and accuracy.

3.3. Overall Algorithm

As mentioned previously, when a 16x16 block is determined as homogeneous block, 16x16 block size is chosen. In addition, 16x8 or 8x16 block size is also considered. The reason for including RDO computation on 16x8 or 8x16 blocks structure is to cater for the situation when 16x16 homogeneous block is at the edge of object boundary and part of it is covered area in the previous frame. If this happens, the encoder cannot find a good prediction of the 16x16 homogeneous block from the previous frames. Note however, that these occurrences are rare since most homogeneous regions do not split into smaller block sizes. The selection of 16x8 or 8x16 block depends on the results of fast INTRA mode decision [4]. If the selected INTRA mode is vertical prediction, we will use 8x16 block. If the selected mode is horizontal prediction, we will use 16x8 block instead. Otherwise, only 16x16 block is used. Similarly, when a 8x8 block is detected as homogeneous region, the size selected is simply 8x8 block, thus skipping the RDO computations on the 8x4, 4x8 and 4x4 block sizes.

When video objects in the sequences are stationary, there is a very high tendency that it will be encoded using 16x16 block motion estimation. Therefore after motion estimation is performed on 16x16 block, motion vector is found to be zero and the macroblock difference is small, only 16x16 block size will be used for RDO whereas all the other block sizes are skipped. List. 1 shows the detailed steps of our approach. Note that Step 1 and Step 2 are used in fast INTRA mode decision approach [4], and thus are not repeated when the two algorithms are combined.

Step 1. Edge operator is used to generate the edge map of one image frame.
Step 2. The edge direction histogram is generated.
Step 3. Check if the current 16x16 block has zero motion. If not, proceed to Step 6.
Step 4. Otherwise, compute the MB difference of the 16x16 block. If the sum is greater than threshold, go to Step 6.
Step 5. If it is smaller than or equal to the threshold, perform motion estimation on the 16x16 block and encode it. Go to Step 3 for the next 16x16 block.
Step 6. Determine if the 16x16 block is homogeneous.
Step 6.1. If the 16x16 block is homogeneous, the encoder performs RDO on the 16x16 and/or 16x8 or 8x16 block. The best mode is chosen from the modes just computed. Computations for other sizes are skipped.
Step 6.2. If the macroblock is non-homogeneous, RDO and motion estimation is performed on 16x16, 16x8 and 8x16 blocks. The results for the best mode of the three are saved.
Step 6.2.1. For each 8x8 block in the macroblock, if it is homogeneous, only fast motion estimation is performed on the 8x8 block and the best type is selected to be 8x8.
Step 6.2.2. If it is non-homogeneous, then RDO is performed on 8x8, 8x4, 4x8 and 4x4 blocks.
Step 7. Steps 6.2.1 and 6.2.2 are repeated until all the best 8´8 block subtype is determined.
Step 8. Determine the best mode between P8x8 type and the best type from Step 6.2.

List. 1. Overall algorithm description

4. EXPERIMENT RESULTS

The fast INTER mode prediction was implemented into JM5.0c encoder with the fast motion estimation algorithm and fast INTRA prediction algorithm. The fast motion estimation algorithm used is from JVT-F017 [3] and the fast INTRA prediction technique is from JVT-G013 [4]. We compared our proposed technique (fast motion estimation + fast INTRA + fast INTER) with the original (fast motion estimation + fast INTRA). According to the specifications [11], the test conditions are as follows: 1) MV search range is ±32 pixels; 2) Hadamard transform is used; 3) optimization is enabled; 4) reference frame

number equals to 5; 5) CABAC is enabled; 6) MV resolution is ¼ pixel; 7) GOP structure is IPPP or IBBP; 8) the number of frames in a sequence is 150.

A group of experiments were carried out on the test sequences with the 4 quantization parameters, i.e., QP=28, 32, 36, and 40 as specified in [12]. In the experiments, the block homogeneity threshold Thd_H is set to 20000 for 16x16 block and one-fourth of the value for 8x8 block. Calculation of average PSNR differences and average bit rate differences follows the specification in [13]. The results are tabulated in Table 1 and Table 2 corresponding to the picture type of IPPP and IBBP respectively. In the table positive values mean increments, and negative values mean decrements.

4.1. Experiments on IPPP Sequences

From the experimental results in Table 1, it is observed that the proposed approach has reduced the encoding time by 30% on average. It has shown consistent gain in coding speed for all video sequences with the least gain of 9.97% in mobile video sequence and most gain of 45.16% in silent video sequence. The maximum PSNR loss is 0.065 dB and thus negligible. The bit rate increase is also negligible with 1.28% maximum. For the sequences of Silent and News, the gain in coding speed is high because both the sequences shows strong spatial homogeneity and temporal stationarity in the frames. On the other hand, the sequence of Mobile has a lot of small moving objects, such as the letters on the calendar. Therefore, there are not many homogeneous regions in a frame or stationary regions between frames. It explains the reason why the time reduction of this sequence is not as much compared to other sequences.

Table 1. Results for IPPP sequences

Sequence	Time(%)	Psnr(dB)	Bits(%)
Foreman(qcif)	-25.18	-0.062	1.28
News(qcif)	-42.62	-0.065	1.18
Container(qcif)	-36.25	-0.012	0.30
Silent(qcif)	-45.16	-0.022	0.47
Paris(cif)	-31.90	-0.040	0.87
Mobile(cif)	-9.97	-0.005	0.13
Stefan(cif)	-17.37	-0.015	0.33

4.2. Experiments on IBBP Sequences

In Table 2, the experiment results shows that our approach has similar performances on IBBP as on IPPP. On average, encoding time has been reduced by 30%. Consistent gain in speed for all video sequences is achieved with the least gain of 9.21% in mobile video sequence and most gain of 45.92% in silent video sequence. The PSNR loss is negligible with the highest loss at 0.055 dB. The bit rate increase is also negligible with the highest increase at 1.21%. The overall consistency of experimental results between IBBP and IPPP sequences is expected since homogeneity refers to regions inside one same frame and are not affected by reference frames whereas stationarity means stationary parts during a relatively long time interval, and will not change much with regard to nearby reference frames.

Table 2. Results for IBBP sequences

Sequence	Time(%)	Psnr(dB)	Bits(%)
Foreman(qcif)	-24.61	-0.050	1.15
News(qcif)	-40.52	-0.029	0.54
Container(qcif)	-39.05	-0.027	-0.45
Silent(qcif)	-45.92	-0.055	1.21
Paris(cif)	-26.54	-0.040	0.91
Mobile(cif)	-9.21	-0.002	0.06
Stefan(cif)	-16.31	-0.015	0.35

5. CONCLUSION

A fast INTER mode decision technique that makes use of the homogeneity of video object's textures and temporal stationarity characteristics in video sequences is proposed. Homogeneity decision of a block is decided based on edge information, and MB differencing is used to judge whether the MB is time-stationary. The new technique is able to achieve a reduction of 30% encoding time on average, with a negligible average PSNR loss of 0.03 dB and 0.6% bit rate increase.

6. REFERENCES

[1] "Information technology - Coding of audio-visual objects - Part 10: Advanced video coding," Final Draft International Standard, ISO/IEC FDIS 14496-10.

[2] Xiang Li, Guowei Wu, "Fast Integer Pixel Motion Estimation," JVT-F011, 6th Meeting, Awaji Island, Japan, December 5-13, 2002.

[3] Zhibo Chen, Peng Zhou, Yun He, "Fast Integer Pel and Fractional Pel Motion Estimation for JVT," JVT-F017, 6th JVT Meeting, Awaji Island, Japan, December 5-13, 2002.

[4] F. Pan, X. Lin, R. Susanto, K. P. Lim, Z. G. Li, G. N. Feng, D. J. Wu, and S. Wu, "Fast Mode Decision Algorithm for Intra Prediction in JVT", JVT-G013, 7th JVT meeting, Pattaya, March 2003.

[5] K. P. Lim, S. Wu, D. J. Wu, S. Rahardja, X. Lin, F. Pan, and Z. G. Li, "Fast Inter Mode Decision," JVT-I020, 9th JVT Meeting, San Diego, United States, September 2003.

[6] B. K. P. Horn and B. G. Schunk. Determining optical flow. Artificial Intelligence, 17:185--203, 1981.

[7] A. M. Tekalp. Digital Video Processing. Prentice Hall, 1995.

[8] K. R. Castleman, Digital Image Processing, Prentice Hall Inc, 1996.

[9] T. Uchiyama, N. Mukawa and H. Kaneko, "Estimation of Homogeneous Regions for Segmentation of Textured Images", IEEE Proceedings in Pattern Recognition, 2000, pp. 1072-1075.

[10] X. W. Liu, D. L. Liang and A. Srivastava, "Image Segmentation Using Local Spectral Histograms", IEEE International Conference on Image Processing 2001, pp. 70-73.

[11] Gary Sullivan, "Recommended Simulation Common Conditions for H.26L Coding Efficiency Experiments on Low Resolution Progressive Scan Source Material," VCEG-N81, 14th meeting: Santa Barbara, USA. Sept. 2001.

[12] JVT Test Model Ad Hoc Group, "Evaluation Sheet for Motion Estimation," Draft version 4, Feb. 2003.

[13] Gisle Bjontegaard, "Calculation of Average PSNR Differences between RD-curves," VCEG-M33, 13th meeting: Austin, Texas, USA, April 2001.