

FAST INTRA MODE DECISION ALGORITHM FOR H.264/AVC VIDEO CODING

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

Institute for Infocomm Research

21 Heng Mui Keng Terrace, Singapore 119613

C. Zhu, W. Ye & Z. Liang

School of Electrical and Electronic Engineering,
Nanyang Technological University, Singapore

ABSTRACT

The emerging H.264/AVC video coding standard aims to significantly improve compression performance compared to all existing video coding standards. In order to achieve this, a robust rate-distortion optimization (RDO) technique is employed to select the best coding mode and reference frame for each macroblock. As a result, the complexity and computation load increase drastically. This paper presents a fast mode decision algorithm for H.264 intra prediction based on local edge information. Prior to intra prediction, an edge map is created and a local edge direction histogram is then established for each sub-block. Based on the distribution of the edge direction histogram, only a small part of intra prediction modes are chosen for RDO calculation. Experimental results show that the fast intra mode decision scheme increases the speed of intra coding significantly with negligible loss of PSNR.

1. INTRODUCTION

The Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG is finalizing a new standard for the coding of natural video images. The new standard [1] is known as H.264 and also MPEG-4 Part 10, Advanced Video Coding (AVC). Some new techniques, such as spatial prediction in intra coding, adaptive block size motion compensation, 4×4 integer transformation, multiples reference pictures and content adaptive binary arithmetic coding (CABAC), are used in this standard. The testing results show that H.264 can greatly outperform existing video coding standards in both PSNR and visual quality [2].

To achieve the highest coding efficiency, H.264 uses rate distortion optimization (RDO) technique to get the best coding result in terms of maximizing coding quality and minimizing bitrates. This means that the encoder has to code the video by exhaustively trying all the mode combinations including the different intra and inter prediction modes. Therefore, the complexity and computation load of video coding in H.264 increase drastically compared to any previous standards.

To reduce the complexity of H.264, a number of efforts have been made to explore the fast algorithms in motion estimation, intra mode prediction and inter mode prediction for H.264 video coding [3-6]. Fast motion estimation is a well-studied topic and is widely applied in the existing standards such as MPEG-1/2/4 and H.261/H.263. On the other hand, fast mode decision is a new topic in H.264 coding, and no previous work

exists so far. It is believed that fast mode decision algorithms are also very important in reducing the overall complexity of H.264. We have made two contributions to H.264 related to the fast mode decision algorithms which are adopted as part of non-normative reference model for H.264 [4-5]. This paper presents one of the contributions, a fast intra mode decision algorithm for H.264 intra prediction by using local edge information. It is shown that the presented algorithm considerably reduces the amount of calculations of intra prediction.

It is observed that the pixels along the direction of local edge are normally of the similar values, and a good prediction could be achieved if the pixels are predicted using those neighboring pixels that are in the same direction of the edge. Therefore, an edge map that represents the local edge orientation and strength is created, and a local edge direction histogram is then established for each sub-block. Based on the distribution of the edge direction histogram and the concept of majority votes, only a small number of prediction modes are chosen for RDO calculation during intra prediction. Experimental results show that the fast intra mode decision algorithm increases the speed of intra coding significantly.

The rest of the paper is organized as follows. Section 2 and 3 present in detail the fast intra prediction algorithm based on the edge direction histogram. Experimental results are presented in section 4 and conclusions are given in section 5.

2. DETERMINING THE PRIMARY EDGE DIRECTION

H.264 video coding is based on the concept of rate distortion optimization (RDO) which means that the encoder has to encode the intra block using all the mode combinations and choose the one that gives the best RDO performance. According to the structure of intra prediction in H.264, the number of mode combinations for luma and chroma blocks in an MB is $M8 \times (M4 \times 16 + M16)$, where $M8$, $M4$ and $M16$ represent the number of modes for 8×8 chroma blocks, 4×4 and 16×16 luma blocks respectively. This means that, for a macroblock, it has to perform $4 \times (9 \times 16 + 4) = 592$ different RDO calculations before a best RDO mode is determined. As a result, the complexity and computation load of the encoder is extremely high.

As is mentioned previously, to achieve good prediction, the pixels should be predicted using those neighboring pixels that are in the same direction of the edge. Figure 1 shows a few 4×4 edge patterns and their preferred intra prediction directions. The algorithm described in the following is based on an edge detection technique to get the local edge directional information due to its simplicity computationally.

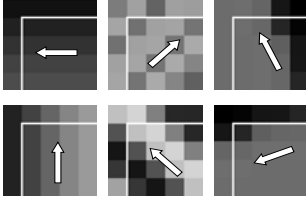


Figure 1. 4x4 patterns and the preferred prediction directions

2.1. Edge Map

In order to obtain the edge information in the neighborhood of the intra block to be predicted, the edge map of the video picture is generated by applying the Sobel edge operators. Each pixel in the video picture will then be associated with an element in the edge map, which is the edge vector containing its edge direction and amplitude.

Sobel operator has two convolution kernels which correspond to degree of difference in vertical direction and horizontal directions. For a pixel $p_{i,j}$ in a luma (or chroma) picture, the corresponding edge vector, $\vec{D}_{i,j} = \{dx_{i,j}, dy_{i,j}\}$, is defined as,

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

$$dy_{i,j} = p_{i+1,j-1} + 2 \times p_{i+1,j} + p_{i+1,j+1} - p_{i-1,j-1} - 2 \times p_{i-1,j} - p_{i-1,j+1}$$

where $dx_{i,j}$ and $dy_{i,j}$ represent the degree of difference in vertical and horizontal directions respectively. Therefore, the amplitude of the edge vector can be roughly estimated by,

$$Amp(\vec{D}_{i,j}) = |dx_{i,j}| + |dy_{i,j}| \quad (2)$$

In fact the amplitude could be obtained more accurately by using the rooted sum of the squares of $dx_{i,j}$ and $dy_{i,j}$. As the latter is more computationally expensive therefore Equation (2) is preferred and proposed here. The direction of the edge (in degree) is decided by the hyper-function,

$$Ang(\vec{D}_{i,j}) = \frac{180^\circ}{\pi} \times \arctan\left(\frac{dy_{i,j}}{dx_{i,j}}\right), \quad |Ang(\vec{D}_{i,j})| < 90^\circ \quad (3)$$

It must be noted that in the actual implementation of the algorithm, Equation (3) is not necessary. This is due to the fact that in H.264 there is only limited number of directions that intra prediction could be applied and thus a simple threshold method can be used to build up the edge direction histogram instead.

2.2. Edge Direction Histogram

In order to decide whether the image block contains an edge, and how strong this edge is, an edge direction histogram is calculated from all the pixels in the block by summing up the amplitudes of those pixels with similar directions in the block.

2.2.1. 4x4 luma block edge direction histogram

For a 4x4 luma block, there are 8 directional prediction modes, plus a DC prediction mode. The border between any two adjacent directional prediction modes is the bisectrix of the two corresponding directions. For example, the border of mode 1 (0°) and mode 8 (26.6°) is the direction at 13.3° , this is because that for Mode 8 (Horizontal-Up), prediction is done at an angle of approximately 26.6° above horizontal direction. It is important to note that mode 3 and mode 8 are adjacent due to circular symmetry of the prediction modes. The mode of each pixel is

determined by its edge direction. Therefore the edge direction histogram of a 4×4 luma block is decided as,

$$Histo(k) = \sum_{(m,n) \in SET(k)} Amp(\vec{D}_{m,n}), \quad (4)$$

$$SET(k) \in \{\{(i_0, j_0)\}, \{(i_1, j_1)\}, \dots, \{(i_k, j_k)\}, \dots, \{(i_8, j_8)\} \mid Ang(\vec{D}_{i,j_k}) \in a_k, k \neq 2\}$$

$$\text{while } a_0 = (-103.3^\circ, -76.6^\circ]$$

$$a_1 = (-13.3^\circ, 13.3^\circ]$$

$$a_3 = (35.8^\circ, 54.2^\circ]$$

$$a_4 = (-54.2^\circ, -35.8^\circ]$$

$$a_5 = (-76.7^\circ, -54.2^\circ]$$

$$a_6 = (-35.8^\circ, -13.3^\circ]$$

$$a_7 = (54.2^\circ, 76.7^\circ]$$

$$a_8 = (13.3^\circ, 35.8^\circ]$$

where $k=0,1,3,4,\dots,8$ refers to the 8 directional prediction modes. Note also that in the above equation, the direction has a period of 180° . Figure 2 shows an example of the edge direction histogram. It can be shown from this figure that this block exhibits strong edge in the Horizontal Down direction.

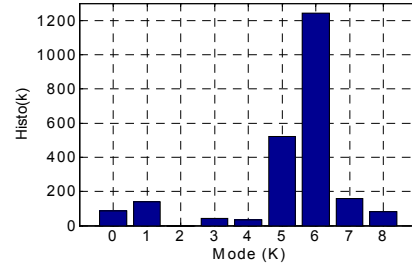


Figure 2. An example of 4x4 block edge direction histogram

2.2.2. Edge direction histogram for 16x16 and 8x8 blocks

For the 16x16 luma and 8x8 chroma blocks, there are only two directional prediction modes, plus a plane prediction and a DC prediction modes. Therefore, the edge direction histogram for this case will be based on three directions, i.e., horizontal, vertical and diagonal (plane) directions. The edge direction histogram for 16x16 and 8x8 blocks is constructed as follows,

$$Histo(k) = \sum_{(m,n) \in SET(k)} Amp(\vec{D}_{m,n}), \quad (5)$$

$$SET(k) \in \{\{(i_0, j_0)\}, \{(i_1, j_1)\}, \{(i_3, j_3)\} \mid Ang(\vec{D}_{i,j_k}) \in a_k\},$$

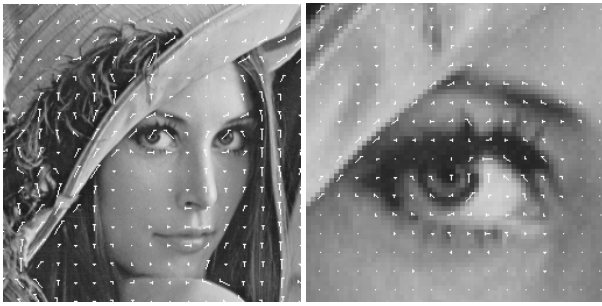
$$\text{while } a_0 = [-112.5^\circ, -67.5^\circ)$$

$$a_1 = [-22.5^\circ, +22.5^\circ)$$

$$a_3 = [-67.5^\circ, -22.5^\circ)$$

where $k=0,1$ and 3 refer to the vertical, horizontal and the plane prediction mode respectively.

As mentioned above, each cell in the edge direction histogram sums up the amplitudes of those pixels with similar edge directions in the block. Alternatively, edge direction histogram counts the number of pixels with the similar edge directions. Therefore, based on the principle of majority votes, the cell k with the maximum amplitude indicates that there is a strong edge along this direction and as such are considered as the preferable prediction direction. The mode whose direction complies with such k is chosen as the primary prediction mode. Figure 3 illustrates two images overlaid with their edge orientation. It clearly shows that the edge orientation provides good guidance as to which direction the intra prediction should be performed.



(a) block size = 16×16 (b) block size = 4×4 (zoom in)

Figure 3. Prediction directions along the edge

3. MODE DECISION FOR INTRA PREDICTION

Based on the primary prediction mode determined previously, the fast mode decision algorithm selects a small number of the prediction modes as the candidates to be used in RDO computation. It should be noted that, the actual RDO computation in H.264 intra coding is based on the reconstructed images. While the edge directional histogram is calculated from the original lossless images, the primary prediction mode decided above will not always be the best RDO mode in actual coding. A number of ways in deciding the number of preferred prediction modes were explored, and experimental results have shown that, besides the primary prediction modes, the best RDO mode is always one of the two adjacent modes (in terms of direction) of the primary prediction modes.

3.1. 4×4 Luma Block Prediction Modes

In the edge detection based approach, the histogram cell with the maximum amplitude is the best candidate for intra prediction. When all the cells have similar amplitudes, DC mode will be a better choice, thus an amplitude threshold is needed in deciding whether the intra block exhibits strong edge presence or is just a flat region. However, it is difficult to pre-define a universal threshold that suit for different block contexts and different video sequences. Therefore DC mode is always chosen as the second candidate for participation of the RDO operation. Extensive experiments also show that besides the primary prediction mode, the best RDO mode is one of the two adjacent modes (in terms of direction) of the primary prediction modes. Hence, two additional candidate prediction modes are selected to be the two neighbors of the primary prediction mode. For example, if the primary prediction mode is mode 1, then two additional candidate prediction modes will be mode 8 and mode 6. Note that mode 8 and mode 3 are adjacent modes in terms of directions due to the symmetry of the circle.

In summary, for the 4×4 luma block intra coding, the histogram cell with the maximum amplitude, and its two adjacent cells, plus DC mode are chosen to take part in RDO calculation. Therefore, for each 4×4 luma block, we will only perform 4 modes RDO calculation, instead of 9.

3.2. 16×16 Luma Block Prediction Modes

Based on the same observation above, only primary prediction mode decided by edge direction histogram is considered as a candidate of best prediction mode, and DC mode is also chosen

as the next candidate. Therefore, for each 16×16 luma block, we will only perform 2 modes RDO calculation, instead of 4.

3.3. 8×8 Chroma Prediction Modes

For the chroma blocks, there are two different edge direction histograms, one from component U and the other from V. The primary prediction modes from the two components are both considered as candidate modes. Same as before, DC mode will also be used in the RDO calculation. Note that if the primary prediction modes from the two components are the same, there could only be 2 candidate modes for RDO calculation; otherwise, there will be 3. Thus, for each 8×8 chroma block, we will only perform 2 or 3 modes RDO calculation, instead of 4.

Table 1. Number of candidate modes

Block size	Total No. of modes	No. of modes used
4×4 (Y)	9	4
16×16 (Y)	4	2
8×8 (U or V)	4	3 or 2*

*The 2-chroma blocks have the same mode.

3.4. Algorithm Complexity Analysis

Table 1 summarizes the number of candidates selected for RDO calculation based on edge direction histogram. As can be seen from Table 1, the encoder with the fast mode decision algorithm would need to perform only 132 or 198 RDO calculations, which are much less than that of current H.264 video coding (592).

3.5. Early Termination of RDO Calculation

The RDO calculations need to have early termination capability should it foresee that the current mode will not be the best prediction mode at any stage. In RDO, the coding cost consists two parts: rate and distortion. After calculating the cost of rate, there might be cases that the cost of rate is larger than the coding cost of the best mode in the previous modes. The RDO calculation will be terminated and the calculation of the distortion is then eliminated because the current mode will not be the best mode anymore.

A macroblock is encoded by either intra 16×16 coding or intra 4×4 coding. In RDO, the selection between these two coding type is determined by the coding costs of the macroblock by each coding type. After intra 16×16 coding, intra 4×4 coding will be applied to the sixteen 4×4 blocks in the macroblock and the costs of each block will be accumulated. If the accumulated cost is larger than the coding cost of intra 16×16 coding, the remaining of 4×4 blocks in the macroblock will not be coded.

4. EXPERIMENTAL RESULTS

The proposed algorithm was implemented into JM6.1e provided by JVT according to the test specifications provided in [7]. A group of experiments were carried out on the recommended sequences with quantization parameters 28, 32, 36 and 40 as recommended by JVT Test Model Ad Hoc Group [8]. The comparison results were produced and tabulated based on the change of average data bits (Δ Bit), the change of average coding time (Δ Time), and the change of average PSNR (Δ PSnr), and the RD curve is also plotted by fitting the testing results according to reference [9].

4.1. Experiments on IPPPP Sequences

Table 2 shows the tabulated performance comparison of the proposed algorithm with JM6.1e for the sequences listed in [8]. In this experiment, the total number of frames is 300 for each sequence, and the period of I-frames is 100, i.e., there is one I-frame for every 100 coded frames. Note that in the table positive values mean increments, and negative values mean decrements. It can be seen that the fast intra prediction algorithm achieves consistent timesaving (average 25%) with negligible losses in PSNR and increments in bit rate. This means that, the fast intra algorithm only takes about $\frac{3}{4}$ of the time that is needed by JM6.1e. Figures 4 show the RD curves of the two sequences “news” and “mobile”. Again, this figure has shown that the fast intra prediction algorithm has the similar R-D performance as that of JM6.1e.

Table 2. Results for IPPPP sequences

Sequence	Time(%)	Δ Psnr(dB)	Δ Bits(%)
Foreman(qcif)	-21.800	-0.077	1.536
News(qcif)	-23.109	-0.067	1.226
Container(qcif)	-20.784	-0.081	1.803
Silent(qcif)	-21.941	-0.033	0.857
Coastguard (qcif)	-21.198	-0.017	0.496
Paris(cif)	-26.901	-0.023	0.504
Mobile(cif)	-27.665	-0.018	0.451
Tempete(cif)	-26.723	-0.029	0.812
Bus(cif)	-26.048	-0.013	0.325
Stefan(cif)	-26.224	-0.017	0.406
Goldfish(cif)	-25.092	-0.177	3.390
Bike(cif)	-26.843	-0.045	0.736

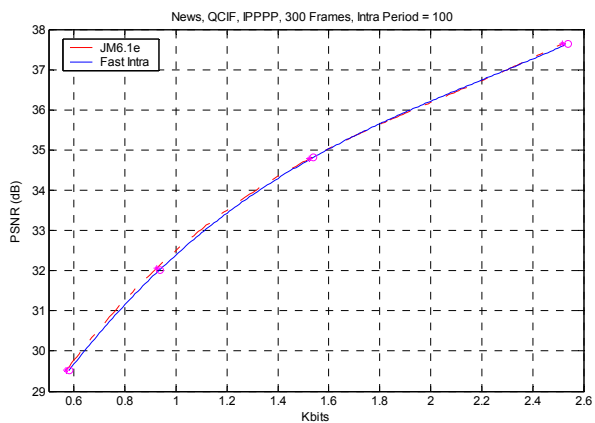


Figure 4. News, Δ Psnr = -0.067dB, Δ Bits =1.226 %

4.2. Experiments on All Intra Frames Sequences

In this experiment, the period of I-frames is set to 0, i.e., all the frames in the sequence are intra coded. It can be seen from the Table 3, the fast intra prediction algorithm achieves very high timesaving (average 60%), which means that the fast intra algorithm only takes about 40% of the time that is needed by JM6.1e. The average loss of PSNR is about 0.24dB, or equivalently, a slight increment in bit rate of about 3.7%.

Table 3. Results for all intra frame sequences

Sequence	Time(%)	Δ Psnr(dB)	Δ Bits(%)
Foreman(qcif)	-65.378	-0.285	4.437
News(qcif)	-55.339	-0.294	3.902
Container(qcif)	-56.357	-0.234	3.695
Silent(qcif)	-65.170	-0.183	3.540
Coastguard(qcif)	-55.026	-0.106	2.361
Paris(cif)	-57.779	-0.230	3.21
Mobile(cif)	-59.086	-0.255	3.168
Tempete(cif)	-57.697	-0.229	3.514
Bus(cif)	-58.118	-0.218	3.849
Stefan(cif)	-57.972	-0.242	3.717
Goldfish(cif)	-67.083	-0.393	5.938
Bike(cif)	-62.896	-0.262	3.216

5. CONCLUSIONS

This paper presents a fast mode decision algorithm for intra prediction in H.264 video coding. By making use of the edge direction histogram, the number of mode combinations for luma and chroma blocks in an MB that takes part in RDO calculation is reduced significantly from 592 to as low as 132. Other techniques such as early termination of RDO mode calculation are also used to further reduce the computation time. This results in a great reduction of the complexity and computation load of the encoder. Experimental results show that the fast algorithm has a negligible loss of PSNR compared to the original scheme.

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] “Preliminary Results of Subjective Assessment of Responses to Video Call for New Tools to Further Improve Coding Efficiency”, ISO/IEC JTC1/SC29/WG11, N4355, July 2001.
- [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 2002.
- [4] F. Pan, X. Lin, S. Rahardja, K. P. Lim, Z. G. Li, G. N. Feng, D. J. Wu, and S. Wu, “Fast Mode Decision Algorithm for JVT Intra Prediction,” JVT-G013, 7th JVT Meeting, Pattaya, Thailand, 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] Bojun Meng, Oscar C. Au, Chi-Wah Wong, Hong-Kwai Lam, “Efficient Intra-Prediction Mode Selection for 4x4 Blocks in H.264,” 2003 IEEE Inter. Conf. on Multimedia & Expo, Baltimore, Maryland, July 6-9, pp. III-521 – III-524.
- [7] 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.
- [8] JVT Test Model Ad Hoc Group, “Evaluation Sheet for Motion Estimation,” Draft version 4, Feb. 2003.
- [9] Gisle Bjontegaard, “Calculation of Average PSNR Differences between RD-curves,” VCEG-M33, 13th meeting: Austin, Texas, USA, April 2001.