Journal of
Software Engineering

ISSN 1819-4311

Academic
Journals Inc.

www.academicjournals.com

Journal of Software Engineering 9 (1): 66-76, 2015
ISSN 1819-4311 / DOI: 10.3923/jse.2015.66.76

Fast Intra Prediction Based on the Texture Feature for H.264/AVC Encoding

[1]Ping Wang, [1]Yichen Chong and [2]Liejun Wang
[1]School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, 710049, China
[2]School of Information Science and Engineering, Xinjiang University, Urumqi, 830046, China

Corresponding Author: Ping Wang, Department of Information and Communication Engineering, Room 611, School of Electronic and Information Engineering, Xi'an Jiaotong University, No. 28 Xianning West Road, Xi'an, 710049, China

ABSTRACT

Intra prediction in H.264/AVC high profile supports rich block types and prediction modes to achieve high coding efficiency. In order to obtain the best block type and prediction mode, the full search algorithm requires a lot of calculations. This study aims to reduce the computational complexity of intra prediction by choosing both appropriate block types and a few candidate modes. In the proposed algorithm, each macroblock is divided into sixteen 4×4 sub blocks and edge strength and direction of each sub block are obtained based on the edge detection templates. Then the mean absolute deviation of edge strengths among sixteen sub blocks is used to measure the smoothness degree of the macroblock and select the appropriate block types. For the selected block types, the candidate modes can be further chosen according to the edge direction of the block. Experimental results show that the proposed fast algorithm can achieve about 74% computational time saving on average for encoding the all intra-frame sequence with negligible loss of coding performance compared with the original full search algorithm.

Key words: H.264/AVC, intra prediction, block type decision, candidate modes decision

INTRODUCTION

The H.264/AVC video coding standard adopts several advanced techniques to achieve outstanding coding performance. One of them is directional intra prediction which supports rich block types and directional prediction modes (Wiegand *et al*., 2003). The best block type and the optimal prediction modes for each macroblock are determined by using the Rate-Distortion Optimization (RDO) technique. However, the RDO process has extremely high computational complexity which accounts for most of the computation of intra prediction. Therefore, fast intra prediction algorithms are needed to reduce the complexity.

Many efforts have been made to explore fast algorithms for intra prediction. The full search algorithm in the H.264 reference codec computes the Rate-Distortion cost (RD cost) for each possible block type and its all prediction modes and then finds the best one with the smallest RD cost. The RD cost is calculated based on the actual bit rate and distortion after entropy coding and reconstruction. Although, the complexity can be reduced by simplifying the computation of RD cost without reconstruction, these fast algorithms always result in much loss of coding performance. One feasible method is to pre-select prediction block types or decrease the number of candidate modes and then to calculate the RD cost only for these types and modes.

Fast block type decision algorithms reduce the complexity by selecting one or two from the original three block types. These algorithms usually choose the appropriate block types based on the smoothness of a macroblock. This is because that the larger block size tends to work well in smooth area while the smaller block size is more suitable for macroblocks with rich detailed information. To measure the smoothness degree of a macroblock, Tian *et al*. (2008) used the entropy feature; Zhang *et al*. (2008) selected the ratio of AC and DC coefficients energy in the transform domain, Bharanitharan and Tsai (2009) and Huang *et al*. (2010) both used the variance of the macroblock. These features perform well in the block type decision but cannot be used in the candidate modes decision.

Because the prediction modes in H.264 are directional, many mode decision algorithms are based on the edge direction of blocks. Pan *et al*. (2005) proposed a fast mode decision algorithm based on the local edge direction histogram by applying the Sobel operator to all pixels of the whole picture. Wang *et al*. (2007) presented a mode decision algorithm based on the Dominant Edge Strength (DES) by using the MPEG-7 edge histogram descriptors. Huang *et al*. (2010) improved the DES-based algorithm to achieve higher prediction accuracy by combining the context information. Although, these edge direction based algorithms can choose several candidate modes out of all modes, all three prediction block types have to be used.

In this study, in order to reduce the intra prediction complexity, a fast intra prediction algorithm based on the texture feature is proposed. It selects not only the block types but also the candidate modes. After obtaining the edge features of every 4×4 block, we calculate the deviation of all blocks' edge strengths among a macroblock. This feature can reflect the smoothness degree of a macroblock. So, the impossible block types can be removed firstly. Then the edge direction obtained from the edge features can be used to do the candidate modes decision. Therefore, the proposed algorithm only needs to calculate the RD cost for the candidate modes of the selected block types. Experimental results show that the proposed algorithm achieves better time savings with negligible loss of coding performance.

REVIEW OF INTRA PREDICTION AND DES-BASED MODES DECISION

Intra prediction in H.264/AVC high profile: The H.264/AVC performs Intra prediction in the spatial domain using the reconstructed pixels of adjacent blocks. The luma components and the chroma components are encoded, respectively. In H.264/AVC high profile, there are three prediction block types for luma components: 14, 18 and 116. The 14 type performs prediction based on the blocks of size 4×4 and has nine prediction modes. The prediction modes of 18 are same as those of 14 except that the block size is 8×8. However, 116 only has four prediction modes (Wiegand *et al*., 2003). Figure 1 and 2 show the prediction direction of 116 and 14, respectively. The prediction for chroma components is always performed on 8×8 chroma blocks using one of the four prediction modes.

In the reference H.264 codec JM10.2, when encoding a macroblock four chroma prediction modes are taken as a cycle and in each cycle the encoder examines all luma block types and all modes. It means that 4×(9×16+9×4+4×1) = 736 times of RD cost calculations are performed before the best mode is determined for a macroblock. Thus, the computational complexity of the intra prediction is extremely high. Fast algorithms are needed to reduce the times of RD calculations.

DES-based modes decision: As mentioned above, the prediction modes in H.264 are directional except the DC mode. The DES-based algorithm (Wang *et al*., 2007) is a classical mode decision

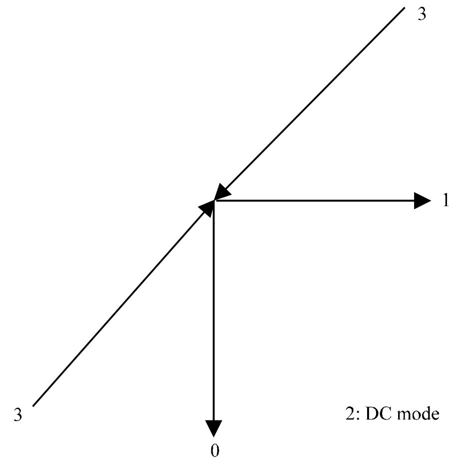

Fig. 1: Prediction modes for 116 type

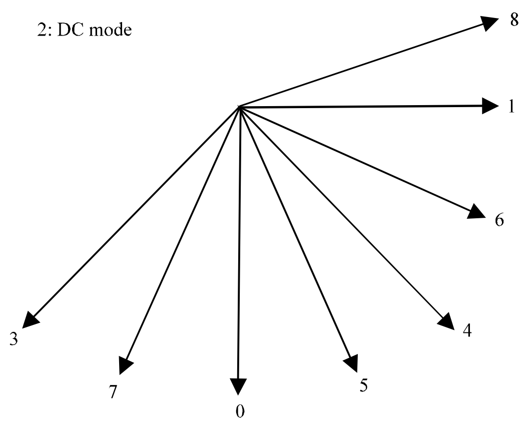

Fig. 2: Prediction modes for 18 and 14 types

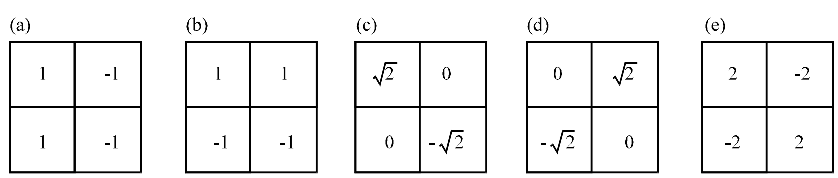

Fig. 3(a-e): Edge strength calculation templates at, (a) 90°, (b) 0°, (c) 45°, (d) 135° and (e) Non directional

algorithm which selects the candidate modes according to the dominant edge direction of current block. The current block is divided into four sub blocks and the averages of pixels in these four sub blocks are obtained separately. Based on these four averages, the edge detection templates shown in Fig. 3 are used to calculate four directional (vertical, horizontal, 45° and 135°) edge strengths and non-directional edge strength. Then the dominant edge direction of the block can be obtained by finding the maximum edge strength. Finally, the candidate modes are selected according to the dominant edge direction. There is one thing to be noticed that the candidate modes decision under different block types are performed separately. For instance, 116 prediction is to divide the whole macroblock into four 8×8-sized sub blocks and calculate the dominant edge direction of the macroblock. The 14 prediction is to divide every 4×4 sub block into four 2×2-sized sub blocks and calculate the edge direction of every 4×4 sub block. So, there is more extra calculation.

FAST INTRA PREDICTION BASED ON THE TEXTURE FEATURE

The DES-based algorithm is efficient for candidate modes decision but it cannot be used for block type selection. As mentioned above, some edge information has been obtained in the DES. Here, we introduced a new feature related to edge based on the DES. This feature can reflect the smoothness degree of the macroblock and the impossible block types can be filtered out before candidate modes decision. Moreover, we improved the candidate modes decision for the prediction of 18 and 116 block types in order to reduce the extra calculation. Thus, the RD costs are only calculated for the candidate modes of the selected block type.

Block type decision: In general, the 14 type is well suited for coding parts of a picture with significant details while the 116 type is more efficient for coding smooth or homogeneous regions of a picture. The computing time spent on the 116 type is definitely much shorter than that of the 14 type. Therefore, for those "smooth or homogeneous" macroblocks, if we can decide in advance the prediction type is 116, the whole process of 14 prediction can be omitted and a huge amount of calculation can be saved. So, in our algorithm, prior to intra prediction whether the macroblock is smooth will be firstly considered. If the macroblock is smooth or homogeneous, large block size will be selected, otherwise, small block size will be used. As for the features used to measure the smoothness degree, the mean value and the variance of the macroblock have been used in the previous studies. In this study, a feature related to edge will be considered; since, we want to make decision not only on the prediction type but also on the candidate modes.

One macroblock is divided into sixteen 4×4 sub blocks. If the macroblock is not smooth, the difference of the edge information of these sixteen sub blocks should be obvious. Otherwise, the difference is small. Here, we use edge strength as edge feature. The edge strength of a 4×4 sub block is calculated by:

$$ES = ES_v + ES_h \tag{1}$$

where, ES_v and ES_h are the sub block's edge strengths in vertical and horizontal direction, respectively. Besides, the edge direction of the 4×4 sub block can be inferred from the different directional edge strengths and candidate modes can be chosen later. To reflect the difference of sixteen sub blocks' edge strengths in a macroblock, the Mean Absolute Deviation of edge strengths (MAD_{ES}) among the sub blocks is calculated as follow:

$$MAD_{ES} = \frac{1}{16}\sum_{i=1}^{16}\left|ES_i - M_{ES}\right| \quad \text{and} \quad M_{ES} = \frac{1}{16}\sum_{i=1}^{16} ES_i \tag{2}$$

where, ES_i(i = 1,... 16) is the edge strength of each sub block and M_{ES} is the mean value of these sub blocks' edge strengths.

Empirically speaking, if the macroblock is smooth, the value of MAD_{ES} will be smaller and the macroblock should be encoded using large block size. On the contrary, when the macroblock is not a homogeneous region, difference among the sixteen sub blocks will bring the larger MAD_{ES} value. So, the small block size will be a good choice. Therefore, the MAD_{ES} value can be considered as an indication of macroblock smoothness. To observe the relationship between the MAD_{ES} feature of the

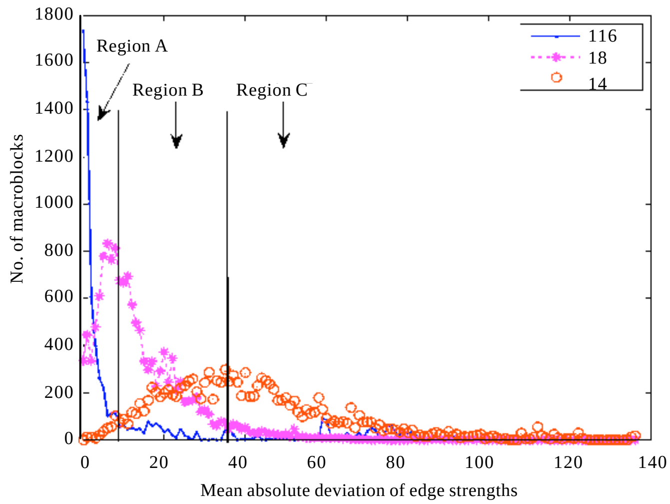

Fig. 4: Histograms of the MAD_{ES} feature for three block types

macroblock and the prediction block type in H.264, we investigate the prediction block type chosen in the full search algorithm under the different MAD_{ES} values. We obtain 32670 training macroblock samples via., H.264 reference codec and each sample contains the MAD_{ES} value of the macroblock and the actual coding block type selected in the codec. These macroblock samples are from 11 videos that represent different degrees of movement. Figure 4 shows the histograms of the MAD_{ES} feature for different prediction block types. The three curves point out the numbers of macroblocks using 14, 18 and 116 block types, respectively under different MAD_{ES} value. We notice that in region A, the MAD_{ES} feature is relative low, the numbers of macroblocks using the 14 type is few. This means that 116 and 18 block types are more possible to be selected. In region B, 14 and 18 block types are more possible to be selected. In region C, the numbers of macroblocks using both 116 and 18 type is few. This means that only 14 block type is more possible to be selected. Therefore, based on the observation above, two thresholds TH_1 and TH_2 are set to do prediction block type decision as shown in Eq. 3:

$$\begin{cases} \text{Block type:} \quad 116 \text{ and } \quad 18 & \text{when } MAD_{ES} \le TH_1 \\ \text{Block type:} \quad 14 \text{ and } \quad 18 & \text{when } TH_1 < MAD_{ES} < TH_2 \\ \text{Block type:} \quad 14 & \text{when } MAD_{ES} \ge TH_2 \end{cases} \tag{3}$$

How to set the values of two thresholds? In this study, they are determined by controlling the error decision probability. The threshold TH_1 is used to distinguish region A from region B. The 14 type can be skipped in region A and the 116 type can be skipped in region B. The 18 type is not sensitive with the MAD_{ES} feature and it is always selected in regions A and B. So, the error decision probability related to TH_1 includes two parts, one is for those macroblocks whose best block type selected by the full search is the 14 type when the MAD_{ES} feature is smaller than the threshold TH_1 and another is for those having best block type 116 when the MAD_{ES} is larger than the TH_1. We want to control this error decision probability as low as possible. There is another thing to be noticed that the thresholds used in the existing block type decision algorithms (Tian *et al*., 2008; Zhang *et al*., 2008; Huang *et al*., 2010) are almost constant but in our study the thresholds are considered to be related with the Quantization Parameter (QP) in the codec. This is based on the fact that the effect of smoothness will be more obvious if the QP is larger. So, the 116 type is more frequently selected with higher QP. Likewise, the lower the QP is the more the details and the 14 type is more frequently selected. So, the thresholds in block type decision consider the effect of QP factor. Figure 5 plots the error decision probability related to TH_1 for the training sequence which contains different movements under six different QPs. For the error probability curve under each QP and to achieve the highest block type decision accuracy, the MAD_{ES} value with the minimal error probability is selected as threshold.

A similar method is used to find the threshold TH_2 which distinguishes region B from region C. Only the 14 type is selected in region C. So, the error decision probability related to TH_2 is for those macroblocks whose best block type is the 18 type when the MAD_{ES} feature is larger than the threshold TH_2. Figure 6 shows the relation between this error probability and the threshold TH_2

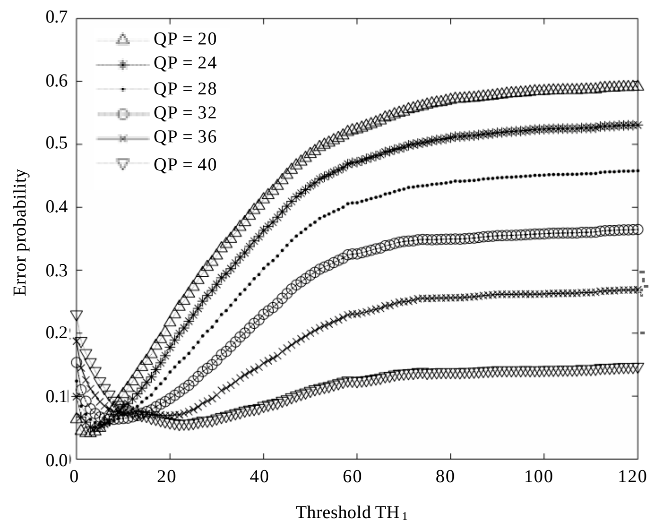

Fig. 5: Error decision probability related to TH_1

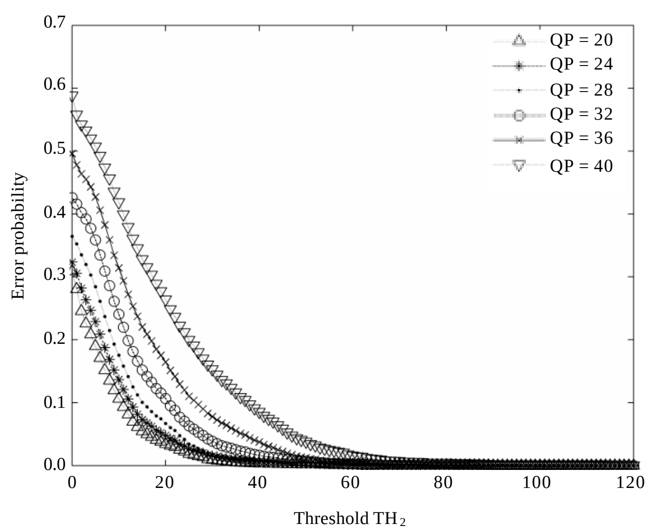

Fig. 6: Error decision probability related to TH_2

under different QPs for the training sequence mentioned above. It can be found that this threshold is also relative to the QP factor. Based on experimental results, the RD performance loss is insignificant in most testing sequences when the decision error is less than 5%. To achieve the more coding time saving, the lowest MAD_{ES} value with the error probability less than 5% is selected as threshold.

For the error decision probability curve under each QP, two thresholds can be obtained by controlling the error probability as low as possible. Next, we use curve fitting technique to fit the relation between the thresholds and the QP factor because we only obtain some thresholds under several QPs but actually QP is an integer from 1-51. Because the effect of QP to the cost function without RDO is linear and also simplicity, it is suitable to model the threshold as a linear relationship of QP. Using piecewise linearization method in the least squares sense, we obtain the:

$$\begin{cases} TH_1 = 0.4055\times QP\text{-}4.4456 \\ TH_2 = 1.3683\times QP+2.9446 \end{cases} \quad (4)$$

Candidate modes decision: After block type decision, to achieve more reduction on the calculation in intra prediction process, we carry out candidate modes decision for the selected block type further. As mentioned before, the edge strengths of each 4×4 block in a macroblock have been obtained. Then the dominant edge direction of the block can be judged and the candidate modes can be determined according to this direction.

If the 14 type is chosen in block type decision, the dominant edge direction of each 4×4 block is used to select the candidate modes out of all 9 modes. If the dominant direction is one of the vertical, horizontal, 45° and 135°, the mode corresponding to the dominant direction and its adjacent two modes are chosen. If the non-directional edge strength is the maximal, all 9 modes are needed. However, because the relationship between the edge direction of the block and the prediction mode was not strong enough by experiments, it could not achieve accurate enough mode decision by judging the edge direction only. In Huang *et al.* (2010) mode decision accuracy was improved by incorporating the contextual information. The most probable mode defined by the modes of upper and left blocks for current block is added to help selecting the candidate modes. Many literatures have pointed out that the probability for the most probable mode to be the best mode is high. So. the most probable mode is always selected. If the most probable mode is one of the three modes selected by dominant direction, DC mode is added in order to improve the accuracy. The decision rules of candidate modes for 14 are listed in Table 1.

For the 18 and 116 types, candidate modes are still selected according to the dominant edge direction of the block. But in the DES-based method, the edge strengths of 16×16, 8×8 and 4×4

Table 1: Candidate modes decision rules

Nine prediction modes (14 and 18)		Four prediction modes (116)	
Dominant edge direction	Candidate modes	Dominant edge direction	Candidate modes
0°	0, 5, 7, MPM or DC	0°	0, DC
90°	1, 6, 8, MPM or DC	90°	1, DC
45°	3, 7, 8, MPM or DC	45°	3, DC
135°	4, 5, 6, MPM or DC		
Non-directional	All		

blocks are obtained separately. That is to say the candidate modes decisions for three block types are independent of each other. This operation leads to more extra calculation. In fact, there is a relationship between the edge direction of a larger image block and the edge directions of the smaller size blocks within the larger block. So, in our study, we only calculate the edge strengths of each 4×4 sub block using the templates and the edge strengths of 8×8 block and 16×16 block are calculated not by the templates but by summing up the edge strengths of the same edge direction of 4×4 sub blocks within the current block. For the 18 prediction, five directional edge strengths of four 8×8 blocks are calculated by accumulation to select the candidate modes. For the 116 prediction, only three directional edge strengths of the whole macroblock need to be accumulated in account of three directional prediction modes. The candidate modes decision rules are also listed in Table 1.

Complexity analysis: The high complexity of intra prediction is mainly due to the fact that the RD costs of all possible modes need to be calculated and compared. Thus, the complexity of the algorithm is usually measured by the times of RD calculation. In the full search algorithm, 736 times of RD cost calculation are performed. In the DES-based algorithms (Wang *et al*., 2007; Huang *et al*., 2010), except for the candidate modes decision for luma components, two candidate modes are also selected for chroma components. The times of RD cost calculation is 164 in the best case (four candidate modes are chosen for the 14 type). In our study, we don't choose the candidate modes for chroma components. We observed by experiments that although the best block types under different chroma modes are possible different, the best prediction modes for three block types under different chroma modes are always same. Therefore, we only calculate the RD costs of the selected candidate modes under the first chroma mode and determine the best luma prediction mode. Then for the other three chroma modes, we only calculate the RD cost of the best luma mode. For the proposed algorithm, the RD cost calculation times will be 33 in the best case when the 116 and 18 block types are chosen and if the smaller block size 18 and 14 are chosen, the times will be 140 and if only the 14 type is selected, the times will be down to 112.

RESULTS

In our experiments, the proposed algorithm and other comparison algorithms were implemented on JM10.2 reference codec provided by JVT. The related parameters were set up as follows, the high profile was used, RD optimization was enabled, CABAC and Hadamard transform were used, frame rate was 30 and the number of frames were 100. The intra period was set to 1, namely, all the frames were intra-coded. The QPs were 24, 28, 32 and 36. Different types of video sequences were used. The simulation was performed on a PC with Pentium 3.0 GHz, 2012 RAM.

To compare the rate distortion performance, the comparison results based on the average Peak-Signal-to-Noise-Ratio (PSNR) differences and the average bit rate differences were calculated. And to compare the computational complexity, the average saving of encoding time was used. Note that all the comparison results were relative to the results of the full search algorithm. So, the positive number meant increasing and the negative number meant decreasing. Average PSNR and bit rate differences (denoted as BDPSNR and BDBR) were calculated according to the numerical averages between RD curves (Bjontegaard, 2001). The percentage of average time saving (denoted as TIME) was obtained by taking average of time savings of four different QPs.

Table 2 shows the comparison results for several block type decision algorithms. Zhang *et al*. (2008) used the ratio of AC and DC coefficients energy of the macroblock in transform domain to

Table 2: Performance comparison for block type decision algorithms

	Zhang *et al.* (2008)			Bharanitharan and Tsai (2009)			Proposed algorithm		
Sequence	BDPSNR (dB)	BDBR (%)	Time (%)	BDPSNR (dB)	BDBR (%)	Time (%)	BDPSNR (dB)	BDBR (%)	Time (%)
Foreman (QCIF)	-0.080	1.391	-36.8	-0.060	1.040	-43.2	-0.032	0.548	-66.2
News (QCIF)	-0.146	1.981	-38.2	-0.073	1.036	-43.8	0.002	0.037	-66.8
Container (QCIF)	-0.202	3.197	-36.9	-0.068	1.094	-45.8	-0.058	0.951	-71.9
Paris (CIF)	-0.117	1.613	-37.7	-0.059	0.813	-43.5	-0.025	0.346	-69.5
Mobile (CIF)	-0.042	0.482	-40.1	-0.027	0.315	-41.0	-0.053	0.620	-71.6
Tempete (CIF)	-0.030	0.444	-39.0	-0.039	0.566	-43.5	-0.015	0.222	-64.3
Average	-0.103	1.518	-38.1	-0.054	0.811	-43.5	-0.030	0.454	-68.4

Table 3: Performance comparison for candidate mode decision algorithms

	Wang *et al.* (2007)			Proposed algorithm		
Sequence	BDPSNR (dB)	BDBR (%)	Time (%)	BDPSNR (dB)	BDBR (%)	Time (%)
Foreman (QCIF)	-0.154	2.660	-48.7	-0.069	1.190	-60.4
News (QCIF)	-0.263	3.556	-51.7	-0.108	1.469	-61.1
Container (QCIF)	-0.236	3.733	-50.8	-0.087	1.384	-60.8
Paris (CIF)	-0.231	3.145	-52.6	-0.083	1.137	-61.9
Mobile (CIF)	-0.168	1.959	-50.1	-0.096	1.117	-59.3
Tempete (CIF)	-0.161	2.348	-51.5	-0.087	1.280	-61.0
Average	-0.202	2.900	-50.9	-0.088	1.263	-60.8

select the proper block type and another study used the variance of the macroblock to do the decision. The thresholds used in these algorithms are both constant and the feature used in them cannot be further used to select the candidate modes. From the data, it can be seen that the rate distortion performance loss of the proposed algorithm is very slight. The average PSNR degradation is about 0.03 dB and the average bit rate increase is only about 0.5%. It means that the rate distortion performance of the proposed block type decision strategy is very close to the full search algorithm. Such a good performance shows the proposed edge based feature in Eq. 2 is effective and the empirical equations about two thresholds in Eq. 4 used in proposed block type decision work well for different sequences. In terms of complexity reduction, one can see that the time saving of the proposed block type decision algorithm are evident for all types of sequences. The average encoding time saving is about 68.4% which is most compared to the other algorithms. It means that the block type decision algorithm can effectively decrease computational complexity. Moreover, the proposed algorithm can achieve more encoding time saving for some video sequences with smaller movement and more smooth regions such as Contain sequence. This is because more macroblocks choose the larger size block type and more complicated 4×4 intra prediction can be omitted.

Table 3 compares the performance between the DES-based and the proposed candidate modes decision algorithms. One can see that the proposed algorithm always achieves lower PSNR degradation and less bit rate increase than the DES-based candidate modes decision algorithm. It means that the proposed algorithm selects the candidate modes more accurately than the DES-based algorithm. The latter selected modes only by judging the dominant edge direction, whereas the proposed algorithm adds the most probable mode into selection which improves the mode decision accuracy distinctly. Moreover, although the numbers of candidate modes in these two algorithms are same, the extra calculation in the proposed algorithm is less. Therefore, the time savings of the proposed algorithm are more than the DES-based algorithm from the Table 3.

Table 4: Performance comparison for intra prediction algorithms

Sequence	Zhang *et al.* (2007) BDPSNR (dB)	Zhang *et al.* (2007) BDBR (%)	Zhang *et al.* (2007) Time (%)	Huang *et al.* (2010) BDPSNR (dB)	Huang *et al.* (2010) BDBR (%)	Huang *et al.* (2010) Time (%)	Proposed algorithm BDPSNR (dB)	Proposed algorithm BDBR (%)	Proposed algorithm Time (%)
Foreman (QCIF)	-0.176	3.042	-68.0	-0.124	2.148	-57.5	-0.106	1.840	-72.1
News (QCIF)	-1.254	16.88	-83.8	-0.163	2.267	-60.0	-0.106	1.491	-73.3
Container (QCIF)	-0.304	4.806	-68.5	-0.148	2.366	-61.8	-0.150	2.400	-76.7
Paris (CIF)	-0.208	2.847	-69.3	-0.136	1.869	-60.4	-0.113	1.552	-75.4
Mobile (CIF)	-0.197	2.292	-69.0	-0.111	1.300	-54.9	-0.164	1.906	-75.0
Tempete (CIF)	-0.186	2.710	-69.8	-0.116	1.691	-58.9	-0.107	1.559	-70.6
Average	-0.388	5.430	-71.4	-0.133	1.940	-58.9	-0.124	1.791	-73.9

Table 4 lists the comparison results of several algorithms combining the block type decision with the candidate modes decision. Zhang *et al.* (2007) selected the candidate modes based on the local edge direction histogram (Pan *et al.*, 2005) and selected the block type by comparing the maximal edge amplitude with thresholds. Because the 18 block type is not considered in this algorithm, whereas three block types are discussed in the proposed algorithm, so the 18 type is always selected in the comparison experiments. Huang *et al.* (2010) selected the block type by the variance of a macroblock and then selected the candidate modes by the improved DES algorithm. As shown in Table 4, the proposed algorithm achieves better rate distortion performance. The average PSNR degradation is about 0.12 dB and the average bit rate increase is only 1.8%. It means that the encoding performance is very similar to that of full search algorithm. The time savings of the proposed algorithm are very significant for all types of sequences and the average time saving is about 74%. It also achieves more time saving especially for low motion or smooth texture sequences. Moreover, compared to the algorithm proposed by Zhang *et al.* (2007), although the advantage of the encoding time saving achieved by proposed fast algorithm is not obvious, the improvement of rate distortion performance is outstanding. This is because the proposed algorithm improves the decision precision of not only block type but also candidate modes significantly. And compared to the algorithm proposed by Huang *et al.* (2010), the proposed algorithm achieves apparent encoding time reduction with a little of improving on rate distortion performance.

In summary, the proposed algorithm can not only effectively speed up the intra prediction process but also achieve similar PSNR and bit rate performance with the full search method.

CONCLUSION

In this study, a fast intra prediction algorithm for H.264/AVC high profile was presented. It reduced the computational complexity of intra prediction not only by the block type selection but also by the candidate modes decision. The intra prediction's block type is highly dependent on the smoothness of macroblock. After edge detection for every 4×4 sub block within one macroblock, the smoothness degree was characterized by using the mean absolute deviation of the sub blocks' edge strengths. Appropriate prediction block types were firstly chosen based on this feature related to edge and two thresholds related to QP. Then the candidate modes for the selected block type were chosen based on the edge direction of the current block and the mode information of adjacent blocks. The final block type and its best mode(s) were obtained after the RDO process of the first chroma mode. For the other three chroma modes, only the best modes were further calculated. Experimental results showed that the proposed algorithm saved approximately 74% encoding time for the all intra-frame sequence with negligible loss in PSNR and slight increase in bit rate.

ACKNOWLEDGMENT

This research was supported by the National Natural Science Foundation of China under Project No. 61202180, China.

REFERENCES

Bharanitharan, K. and A.C. Tsai, 2009. Efficient fast intra mode decision and classified block type algorithms for H.264/AVC Encoder. Proceeding 9th IEEE International Conference on Computer and Information Technology, October 11-14, 2009, Xiamen, pp: 172-176.

Bjontegaard, G., 2001. Calculation of average PSNR differences between RD curves. Document VCEG-M33, ITU-T SC16/Q6, 13th VCEG Meeting, Austin, Texas, USA., April 2001.

Huang, Y.H., T.S. Ou and H.H. Chen, 2010. Fast decision of block size, prediction mode and intra block for H.264 intra prediction. IEEE Trans. Circuits Syst. Video Technol., 20: 1122-1132.

Pan, F., X. Lin, S. Rahardja, K.P. Lim, Z.G. Li, D. Wu and S. Wu, 2005. Fast mode decision algorithm for intraprediction in H.264/AVC video coding. IEEE Trans. Circuits Syst. Video Technol., 15: 813-822.

Tian, G., T. Zhang, X. Wei and S. Goto, 2008. A block type decision algorithm for H.264/AVC intra prediction based on entropy feature. Proceedings of the IEEE Asia Pacific Conference on Circuits and Systems, November 30-December 3, 2008, Macao, pp: 1348-1351.

Wang, J.C., J.F. Wang, J.F. Wang, J.F. Yang and J.T. Chen, 2007. A fast mode decision algorithm and its VLSI design for H.264/AVC intra-prediction. IEEE Trans. Circuits Syst. Video Technol., 17: 1414-1422.

Wiegand, T., G.J. Sullivan, G. Bjontegaard and A. Luthra, 2003. Overview of the H.264/AVC video coding standard. IEEE Trans. Circuits Syst. Video Technol., 13: 560-576.

Zhang, K., C. Yuan, Q. Li and Y. Zhong, 2007. A fast block type decision method for H.264/AVC intra prediction. Proceeding of the 9th International Conference on Advanced Communication Technology, February 12-14, 2007, Gangwon-Do, pp: 673-676.

Zhang, T., G. Tian and S. Goto, 2008. A frequency-based fast block type decision algorithm for intra prediction in H.264/AVC high profile. Proceedings of the IEEE Asia Pacific Conference on Circuits and Systems, November 30-December 3, 2008, Macao, pp: 1292-1295.