Fast Mode Decision in H.264/AVC

P. Lambert, S. Mys, J. Škorupa, J. Slowack, R. Van de Walle
Multimedia Lab,
Ghent University – IBBT,
Belgium

M.Y.Yang, C. Grecos
School of Computing
University of the West of Scotland
Paisley
PA1 2BE
Scotland

V. Argyriou
School of Computing, Information Technology and Engineering
University of East London
London
E16 2RD
UK

ABSTRACT

The latest video coding standard (Wiegand, 2003), H.264/AVC, uses variable block sizes ranging form 16x16 to 4x4 to perform motion estimation in inter-frame coding and a rich set of prediction patterns for intra-frame coding. Then a robust RDO (Rate Distortion Optimization) technique is employed to select the best coding mode and reference frame for each macroblock. As a result, H.264/AVC exhibits high coding efficiency compared to older video coding standards [2, 3] and shows significant future promise in the fields of video broadcasting and communication. However, high coding efficiency also carries high computational complexity. Fast mode decision is one of key techniques to significantly reduce computational complexity for a similar RD (Rate Distortion) performance. This chapter provides an up-to-date critical survey of fast mode decision techniques for the H.264/AVC standard. The motivation for this chapter is twofold: Firstly to provide an up-to-data review of the existing techniques and secondly to offer some insights into the studies of fast mode decision techniques.

Keywords – survey, video coding standard, H.264/AVC, fast mode decision, Rate Distortion optimization, skip mode, intra mode, inter mode, intra/inter mode selection

1. INTRODUCTION

The H.264/AVC video coding standard is the newest video coding standard which is proposed by JVT (Joint Video Team). A number of new design features are adopted in this standard which significantly improve the rate distortion performance as compared to other standards. These features include variable block size and quarter sample accurate motion compensation with motion vectors even outside picture boundaries, multiple reference frames selection, decoupling of referencing from display order for flexibility and removal of extra delay associated with bi-predictive coding, bi-predictive pictures to be used as references for better motion compensation, weighted offsetting of prediction signals for coding efficiency in scenes including fades etc, improved “skipped” and “direct” mode inference for better RD performance in video sequences containing neighboring macroblocks (of the same scene object) moving in a common direction etc. H.264/AVC further allows directional edge extrapolation in intra coded areas for improving the quality of the prediction signal and allowing prediction from neighboring areas that are inter coded, in-loop de-blocking filter for removing compression artifacts as well as providing better quality reconstructed signals for subsequent motion compensation, and hierarchical block size transforms that enable signals with sufficient correlation to use longer basis functions than 4x4 transforms. There are also provisions for embedded processors such as exact match inverse transforms for “drift free” decoded representations and finally the standard provides advanced entropy coding techniques such as CAVLC (Context Adaptive Variable Length Coding) and CABAC (Context Adaptive Binary Arithmetic Coding) which are also present in the H.263 and JPEG2000 standards.

However, the improvements in the RD performance come with significant complexity increases. These new features not only increase the complexity of H.264/AVC encoders but also of the corresponding decoders. Variable block size motion estimation and compensation, Hadamard transform, RDO mode decision, displacement vector resolution and multiple reference frames are the main H.264/AVC encoding tools which increase the complexity of H.264/AVC encoders. In (Ostermann, 2004) an analysis of the complexity increase in the H.264/AVC video coding standard is presented and compared with previous standards. The significant computational complexity makes it very difficult to use the standard as it is in real-time applications. Reducing the complexity without degrading RD performance thus becomes a critical problem.

In order to understand the complexity of H.264/AVC more clearly, an experiment of complexity analysis is performed here. The Intel® VTune™ Performance Analyzer7.0 is used in this work as the evaluation tool to evaluate the software performance and obtain the complexity profile of an H.264/AVC encoder. In this experiment, the Foreman sequence (100 frames, QCIF (Quarter Common Intermediate Format) format, Baseline profile) is encoded on an Intel Pentium-4 3.09GHz PC with 768 MB memory and using the Microsoft Windows XP operating system. Figure. 1 shows the complexity proportion of different encoding modules in the H.264 JM8.1 [21] reference encoder.

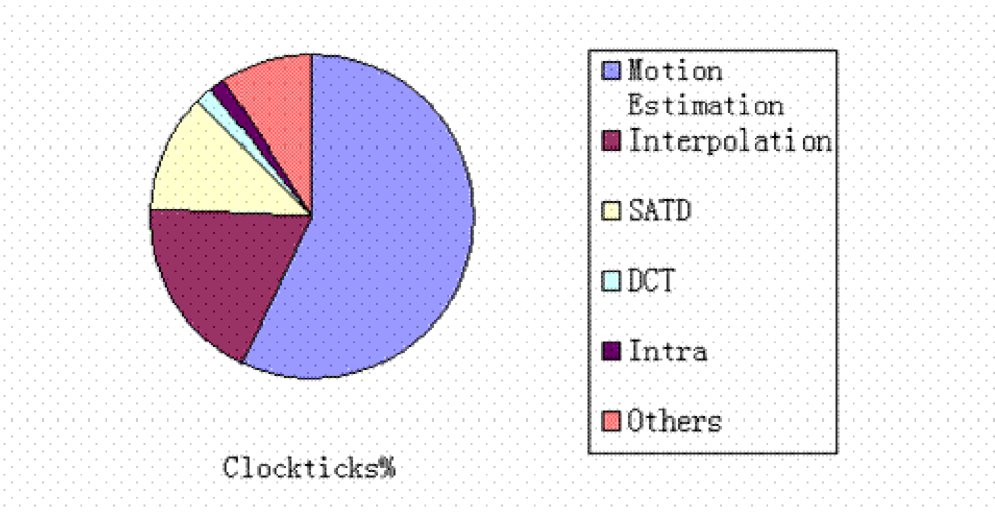

Figure1: Complexity proportion of different encoding modules in H.264/AVC encoder by Intel® VTune™

According to Figure 1, the most time-consuming modules of the H.264/AVC encoder are Motion Estimation, Interpolation, SATD (Sum of Absolute Transformed Differences), and DCT (Discrete cosine transform) which are all related to the RDO based motion estimation and mode decision. Because mode decision covers all these four aspects, a good fast mode prediction algorithm for H.264/AVC is a promising way to reduce the complexity of video encoders.

The relation between complexity reduction and seamless video communication can be best described by using Figure 2 (Hsu 1997) below:

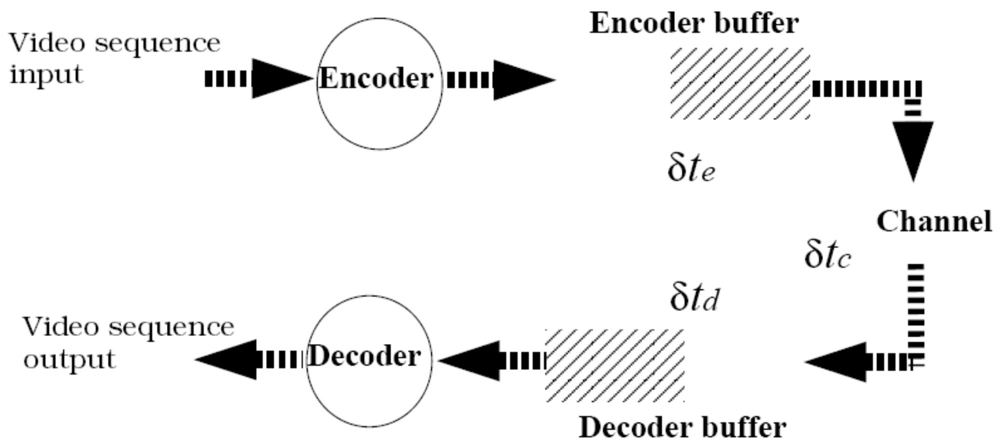

Figure 2: Time delay diagram of a video communication system

The total delay of the system is the sum of the delays in the encoder, decoder buffers and the channel delay. Evidently, there needs to be a balance among the rates of filling the encoder buffer with bits, emptying it to the channel, filling the decoder buffer with bits and emptying it for playing a specified number of video frames/sec. A computationally intensive encoding process will result in disturbing this balance since it will reduce (drastically) the rate of filling up the encoder buffer. This in turn will result in reducing the frames/sec at the decoder end, thus producing a non smooth visual experience (video will appear jerky or frozen at periods). It can be easily deduced that techniques to reduce the computational load at the encoder would restore the rate balance between different components of a communication system and thus enhance the end user experience. Provided that the relationships between the buffer sizes and the rates in different parts of the system are upheld, reduced computation techniques may also imply more frames/sec, thus smoother visual experience at the decoder end. In the context of H264 AVC, most of these techniques are mode decision techniques which are described and analyzed in this chapter.

The necessity of computational complexity reduction for using video on mobile handsets can be evidenced from the fact that consumers increasingly want, and expect their mobile devices to capture and playback HD video without adversely impacting battery life. Handset developers have a range of video codec options to satisfy this consumer demand such as fully customized hardware blocks integrated into system-on-chip (SoC) designs, optimized software codecs running on enhanced-instruction set RISC or DSP processors, or software running on standard processor cores, like ARM 9 or ARM11. In order to be able to bring high-definition video to the hands of users, mobile device designers must optimize the power consumption of all components. Therefore, minimizing the maximum power consumption figure of a chip often concentrates on finding the most power-efficient way to implement video processing algorithms. Again in the context of H264 AVC, most of these power efficient schemes are hardware solutions combined with the software mode decision techniques which are described and analyzed in this chapter. This combination is a classical case of hardware/software co-design.

This chapter is organized as follows: Section 2 gives a brief overview of the mode decision mechanism of H.264/AVC and focuses both on RDO based and on low complexity mode decision preliminaries. Section 3 focuses on speeding up RDO based mode decision techniques. It presents, analyses and compares skip/direct, inter, intra and selective inter/intra schemes. Section 4 concludes the paper and presents future research directions.

2. MODE DECISION IN H.264/AVC

Macroblocks in H.264/AVC standard have many mode candidates due to variable block size motion estimation and directional intra modes. Consequently, some criteria need to be used to decide which candidate mode is the best one for the current macroblock. The H.264/AVC standard suggests two mode decision schemes for encoder: a high-complexity mode decision, also known as Rate Distortion Optimised (RDO) based mode decision, and a low-complexity mode decision. Compared to the low-complexity mode decision, RDO-based motion estimation and mode decision improves PSNR (Peak Signal-to-Noise Ratio) (up to 0.35 dB) and bit rate (up to 9% bit savings) but also comes at the cost of significant computational complexity for common test video sequences. In the rest of this section we will briefly introduce these two mode decision schemes.

2.1 RDO Based Mode Decision for H.264/AVC

In the high complexity mode of the H.264/AVC standard, the macroblock mode is chosen by minimising the Lagrangian function:

$$J(s,c,MODE\,|\,QP,\lambda_{MODE}) = SSD(s,c,MODE\,|\,QP) + \lambda_{MODE} * R(s,c,MODE\,|\,QP) \quad (1)$$

In the above equation, J denotes the cost function and is dependent on s (the original signal macroblock), c (the reconstructed signal macroblock) and MODE (selected from a set of modes as explained later). J is found for a given QP (Quantization Parameter) and λ_{MODE} (the Lagrange multiplier for mode decision). SSD is the sum of the squared differences between the original macroblock and its reconstruction with QP and it also depends on the original and reconstructed macroblock, as well as the mode decision (MODE).

Finally, the rate $R(s,c,MODE\,|\,QP)$ depends on the original and reconstructed macroblock with QP, as well as the chosen MODE, and reflects the number of bits produced for header(s) (including MODE indicators), motion vector(s) and coefficients. It is worth mentioning that an encoder is free to calculate this rate by either measuring or by estimating it. In RDO mode, the reference encoder will actually measure it, in other words, it will code the current macroblock up to and including entropy encoding

In equation (1), MODE is chosen from the set of potential prediction modes as follows:

For Intra slices: $MODE \in \{INTRA4*4, INTRA16*16\}$ (2)

For P slices: {single reference forward or backward prediction}

$$\begin{aligned} MODE \in \{&INTRA4*4, INTRA16*16, SKIP, MODE\_16*16, MODE\_16*8,\\ &MODE\_8*16, MODE\_8*8\} \end{aligned} \quad (3)$$

For B slices: {bi-directionally predicted slices}

$$\begin{aligned} MODE \in \{&INTRA4*4, INTRA16*16, DIRECT, MODE\_16*16, MODE\_16*8,\\ &MODE\_8*16, MODE\_8*8\} \end{aligned} \quad (4)$$

DIRECT mode is particular to the bi-directionally predicted macroblocks in B slices, while SKIP mode implies that no motion or residual information will be encoded (only the MODE indicator is actually transmitted).

In the above mode sets, any mode with the prefix INTRA will result in encoding the spatially predicted signal rather than its temporal residual, while any mode with the prefix MODE_ refers to inter modes. Furthermore, when MODE is equal to INTRA4*4 or INTRA16*16, the best intra mode for each case is chosen through evaluation of the functional of equation 3-12 with mode choices from the following sets:

$$\begin{aligned} INTRA4*4 \in \{ & DC, HORIZONTAL, VERTICAL, DIAGONAL\_DOWN\_RIGHT, \\ & DIAGONAL\_DOWN\_LEFT, VERTICAL\_LEFT, VERTICAL\_RIGHT, \\ & HORIZONTAL\_UP, HORIZONTAL\_DOWN\} \end{aligned} \quad (5)$$

$$INTRA16*16 \in \{DC, HORIZONTAL, VERTICAL, PLANE\} \quad (6)$$

A similar functional minimisation results in the choice of the best 8*8 mode for P and B slices from the following set:

$$MODE\_8*8 \in \{INTER\_8*8, INTER\_8*4, INTER\_4*8, INTER\_4*4\} \quad (7)$$

Any mode with the prefix MODE_ in equations (3,4) and INTER_ in equation (7) assumes that the best motion vector is known for this mode and implies functional minimisations for each candidate motion vector m = (m_x, m_y) inside the search window of the form:

$$J(m, \lambda_{MOTION}) = D_{DFD}(s, c(m)) + \lambda_{MOTION} * R(m - p) \quad (8)$$

where λ_{MOTION} is the Lagrange multiplier for motion estimation, p=(p_x, p_y) is a predicted motion vector and *R(m-p)* is the number of bits for encoding motion residuals only. The rate term is computed from a look-up table, while the distortion term $D_{DFD}(s, c(m))$ depends on the original signal *s* and the reconstructed best match *c* that in turn depends on the candidate motion vector *m*. It has to be noted that the choice of λ_{MOTION} in equation (8) is affected by the choice of the distortion metric.

Once the best 8*8, INTRA4*4, INTRA16*16 modes are found, the minimal cost for the macroblock is evaluated by looping through the different mode possibilities (equations 3, 4).

A straightforward mode decision complexity assessment for a 16*16 macroblock (luma component only) reveals that we need 144 cost evaluations for the best INTRA4*4 mode. Adding 4 more evaluations for the INTRA16*16 case, 16 more for the best 8*8 inter mode and 7 more for selecting the minimal cost among all modes results in 148 evaluations for macroblocks in Intra slices and 171 evaluations for macroblocks in P or B slices. Coupled with similar cost evaluations for the chroma components and the fact that our complexity assessment did not consider evaluations for the best motion vector that depend on the size of the search window and on the sub-pixel accuracy, clearly shows that the mode decision process is computationally intensive.

2.2 Low-Complexity Mode Decision

In the low-complexity mode decision scheme, the block difference between the prediction for each candidate mode and the block to be encoded is calculated in the first step. In the second step, this block difference is either transformed to a single value using the Sum Absolute Transformed Difference (SA(T)D) or the absolute values are calculated and summed for each block location using the Sum of Absolute Difference (SAD) metric. The transform used for SA(T)D is the Hadamard transform. In any case, a set of predicted initial values for $SA(T)D_0$ is then calculated in the third step by using quantisation parameters and bit usage estimates for motion vectors (magnitude + labels). Finally, the minimum $SA(T)D_{min}$ is chosen as shown in the flow-chart of Figure 3.

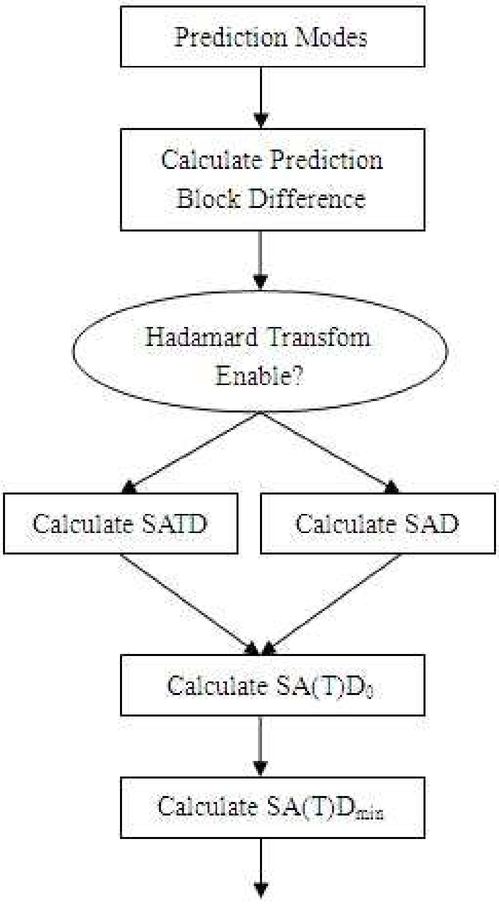

Figure 3: Flow Chart of Low Complexity Mode Decision

Low complexity mode decision uses SA(T)D which includes only subtraction and a simple convolution to represent the distortion term. It also uses $SA(T)D_0$ which only includes the table look up computation to find the bit rate term. No DCT, IDCT (Inverse Discrete Cosine Transform) and entropy coding are included in the low complexity mode decision scheme, which implies much lower computation as compared with the high complexity scheme. The average execution time of low-complexity mode decision is only 7% of that of high-complexity mode decision. However, low-complexity mode decision loses an average of 0.48dB in PSNR compared to the RDO based mode decision. In the following sections, unless we mention low-complexity mode decision explicitly, mode decision means RDO based mode decision.

3. SPEEDING UP RDO BASED MODE DECISION IN H.264/AVC

As explained before, the RDO based mode decision is a very computationally intensive process due to the multiple motion estimations involved. This fact inspired a variety of fast mode decision techniques to be developed for reducing the computational complexity, while retaining similar rate distortion performance. In this section, the most important four categories of fast mode decision techniques — fast skip/direct mode, fast inter mode, fast intra mode and fast intra/inter mode—will be discussed. Inside each category, different mode decision approaches will be presented and compared. The comparisons will enable us to better understand the assumptions, advantages and limitations of these approaches.

3.1 Fast Skip/Direct Mode Decision Techniques

In some prior standards, for example H.263+, a skipped macroblock is defined as a macroblock with zero Motion Vector and zero Quantized Transform Coefficients. A significant proportion of macroblocks are skipped (not coded), particularly in low-motion sequences and/or at higher quantizer step sizes (and hence lower bitrates). The difference between prior standards and H.264/AVC is that in prior standards, a skipped area of a predicted slice was assumed stationary, while in H.264/AVC a predicted motion vector derived directly from previously encoded information is used in skipped areas.

C. Grecos and M.Y Yang (2005) presented a fast skip mode decision technique based on spatio-temporal neighborhood information. The idea is based on the observation (which is particularly evident in slow moving sequences) that the areas of a slice consisting of macroblocks in SKIP mode, slowly change over time. If the macroblocks on the top and left of the one to be encoded in the current frame and the macroblocks on the right and bottom of the co-located macroblock in the previous frame are all in skip mode, then this mode pattern can be a good indication that the current macroblock can be skipped. To strengthen the accuracy of the skip mode prediction, the authors also add an extra condition that the SAD between the current macroblock and its co-located one should be less than the average SAD among the skipped macroblocks in the reference picture and their co-located predictors. The proposed scheme predicts SKIP modes without any motion estimation which can significantly reduce the computational complexity.

A.C.W. Yu, G. R. Martin and H. Park (2008) proposed a novel skip mode detection technique based on three layers. Considering that skip macroblocks tend to occur in clusters (in a similar idea to (Grecos, 2005), spatial and temporal skip mode information is used in the first layer. If the co-located macroblock in the reference frame or at least one of the upper or the left macroblock of the current macroblock in the current frame is in skip mode, the current macroblock passes through the first layer. Otherwise the current macroblock can not be predicted as SKIP mode directly. In the second layer, if the SAD between the current macroblock and its co-located macroblock in the reference frame is smaller than an adaptive threshold (the average SAD of the available skipped neighbors and their collocated ones in the first layer), the current macroblock can be considered as a potential SKIP mode macroblock. Otherwise the current macroblock can not be predicted as SKIP mode. A fast transform-quantized implementation based on (Malvar, 2003) is used in the third layer to detect whether all quantized coefficients of SATD between the current and co-located macroblocks are zeros. Finally, the current macroblock is in SKIP mode if it passes through all the three layers. The authors include the quantization parameter and the integer transform in the mode decision process which makes their work robust, however they do not consider motion information.

C.S. Kannangara et. al. (2006) also proposed a low-complexity SKIP mode decision scheme. The SKIP mode RD cost of the current macroblock can be calculated based on predicted motion vectors without performing any motion estimation. The best inter mode RD cost can be predicted through a model based on local sequence statistics and for a given Lagrange multiplier parameter. This model requires no motion estimation either. The early SKIP mode detection is made by comparing the two RD costs. The achievable computational savings for typical video sequences are in the range of 19%-67% in the baseline profile without significant loss of rate-distortion performance as compared to the standard. The experiments show that the technique achieves very similar Rate Distortion performance with the low complexity mode of H.264/AVC

A Bayesian framework for fast SKIP mode decision is proposed by M. Bystrom, I. Richardson and Y. Zhao (2008). The RD cost difference, which is the difference between the RD costs of the SKIP mode and the best other mode, is used as a discriminator. This difference depends on the QP and the frame content activity and its modeling is probabilistic. Firstly, the conditional probability density functions (PDF) of the RD cost difference are modeled for a set of training sequences at different bit rates. These PDF are measured for the SKIP mode and the other modes. The model of RD cost difference for the SKIP mode is simulated by a Gaussian PDF and the model for other modes is simulated by a Rayleigh PDF. The final RD cost difference can be calculated using Maximum Likelihood (ML) and Maximum a Posteriori (MAP) algorithms based on these conditional PDFs and the priori probabilities of the SKIP and other modes. The a priori probabilities of SKIP and other modes can be calculated as the average frequency of SKIP and other modes in previous frames. Alternatively a look-up table, from where the a priori probabilities can be indexed, can be built based on the video content activity factor and QP. Compared with (Jeon, 2003), this work can achieve 12%-58% more time savings with average 0.04dB PSNR decrease and 0.15% bit rate increase. However, the performance is dependent on the range of content for the training set, the accuracy of modeling and even the sequence resolution. For example, comparing with (Jeon, 2003) in the particular case of mobile video sequences, 13% less time saving is attained with 0.15dB PSNR decrease and 2.75% bit rate increasing.

A fast SKIP mode decision technique which resulted in contributions to the standard, is the work of Jeon et al. (Jeon 2003). According to this work, a macroblock can have SKIP mode in the baseline profile, when the following set of four conditions is satisfied:

1) The best motion compensation block size for this macroblock is 16x16 (MODE_16*16)
2) The best reference slice is the previous slice
3) The best motion vector is the predicted motion vector (regardless of this being a zero motion vector or a non-zero one)
4) The transform coefficients of the 16x16 block size are all quantized to zero.

This set of four conditions is non sufficient due to the assumption that the mode with the lowest RD cost is the inter MODE_16*16 (condition 1), which may be true or not. If it is true, then we can safely say that the macroblock can be skipped since $J_{SKIP} < J_{MODE\_16*16}$ for the same motion vectors as condition 3, thus making the SKIP mode the chosen one due to its lowest cost. If the first condition above is not true though, the algorithm will miss-predict macroblocks as skipped and the RD performance will suffer. The important point here is that although condition 1 makes the set of the above conditions non sufficient for SKIP mode decision, it is "good enough" (dependent on the video content of course). So in order to predict SKIP mode, the approach described in (Jeon, 2003) only needs to perform motion estimation for the 16x16 mode and thus motion estimation for the remaining mode types can be saved if the above conditions are satisfied. Experimental results show that the proposed method results in time savings of 15% on the average without any noticeable RD performance loss as compared to the standard. The proposed technique cannot achieve very significant time savings since motion estimation is still performed for MODE_16x16, however good RD performance is retained due to the use of sufficiently accurate temporal information from the motion estimation step.

J.Lee, B Jeon et al (2004) also presented similar ideas to predict direct mode for B slices. A macroblock is predicted as having SKIP mode when the following set of two conditions is satisfied:

1') The reference slices and the motion vectors are the same as the ones decided under the DIRECT mode.

2') The transform coefficients of the 8x8 sub-blocks of this macroblock are all quantised to zero.

With the same reasoning as above, we can observe that the set of conditions for SKIP mode decision in the B slices of the main profile is non sufficient either. This is due to the fact that no motion compensation takes place as can be seen in conditions 1' and 2' and it is implicitly assumed that the DIRECT_16*16 mode is the one with the minimal cost. If true, the RD performance is not hampered, otherwise it is. Condition 1' in fact enforces the mode of the sub-blocks to be DIRECT_8*8 and the mode of the whole macroblock to be DIRECT_16*16. Furthermore, $J_{SKIP} < J_{MODE\_16*16}$ for the same motion vectors and reference slices from condition 1', clearly implying that the macroblock can be safely skipped. The reader should note that in contrast to the baseline profile, there is actually no assumption in terms of the best reference slice for the SKIP mode detection in the B slices of the main profile. From the design of the standard, the DIRECT and SKIP modes both have the same reference slices and motion vectors, while they are different only in the fact that the SKIP mode needs to have all quantised coefficients in the 8x8 sub-blocks equal to zero, while some coefficients are not zero for the DIRECT mode (condition 2'). Theoretically, condition 2' is a relatively strong one in terms of skipping, since the likelihood of having non zero coefficients in a larger block size (16x16) is very small. Time savings of 52% on the average can be achieved with an average PSNR decrease of 0.01db and average bit rate increase of 0.26% as compared to the standard.

In summary, the above analysis shows that SKIP/DIRECT modes are especially useful for low bit rate coding. Early detection of these modes can lower the encoder complexity significantly (roughly in the order of 20% - 60%) with only small losses in quality. Most SKIP/DIRECT mode decision techniques exploit temporal and spatial neighbourhood information in combination with adaptive thresholds. However, to reduce the impact on the RD performance, such methods should incorporate the QP and the fact that SKIP/DIRECT modes have an inferred motion vector in H.264/AVC, something that is often omitted.

3.2 Fast Inter Mode Decision techniques

X. Jing and L. Chau (2004) propose a fast inter mode decision method which only depends on the pixel based Mean Absolute Difference (MAD) metric between the current and previous frames. The main idea is to use large blocks for smooth areas and small blocks for areas containing complex motions. If the MAD between the current and co-located macroblocks is smaller than the weighted Mean Absolute Frame Difference (MAFD) between the current and previous frames, large mode types {MODE_16*16, MODE_16*8, MODE_8*16} are chosen. Otherwise all mode types are examined. The disadvantage of this algorithm is that the weighting used in the MAFD metric is based on the Quantization Parameter (QP). This implies that initial offline training is needed for relating the weights to QPs for every sequence. Furthermore generalizing the use of these weights to arbitrary sequences is likely not to perform optimally due to differences in motion characteristics, thus making this algorithm not very practical in real time applications. The proposed algorithm can obtain up to 48% computational savings with similar rate distortion performance to the standard for a variety of test sequences.

A. Chang (2003) proposes another algorithm which uses pixel based Sum of Absolute Difference (SAD) information to predict inter modes. If the texture undergoes an integer-pixel translational motion, the texture will look exactly the same in the two consecutive frames and it can be predicted perfectly by integer-pixel motion estimation. If the edges of the texture have a half-pixel or quarter-pixel offset, they may be blurred thus sub-pixel motion estimation will be important. The SAD of the MODE_16*16 is calculated after integer-pixel motion estimation. If this SAD is smaller than an adaptive threshold based on the average SAD of the macroblocks having MODE_16*16 as their best mode, the current macroblock is assumed to have integer-pixel translational motion and MODE_16*16 is chosen as the best mode. Otherwise, the current macroblock may have a half-pixel or quarter-pixel offset. If MODE_16x16 is the best mode, texture analysis and segmentation will be subsequently performed on the best match to potentially refine the best mode to MODE_16*8 or MODE_8*16. However, (Chang, 2003) only considers three coding modes namely MODE_16*16, MODE_16*8 and MODE_8*16 which may be limiting in cases where more refined mode decision is required for improved RD performance due to motion characteristics of sequences. The improvements in RD come of course to the expense of CPU/ run time savings. By considering only three modes, 40.73% run time savings can be achieved with 0.04dB PSNR degradation and 0.92% bit rate increase as compared to the standard.

Motion vector information is used by (Ahmad, 2004) to predict inter modes. This algorithm is based on the principle of 3D recursive search algorithm (Haan, 1993) which provides a fast, convergent and highly accurate motion vector prediction, taking into account the total cost of some modes in the previous and current frames. The total cost of a mode is defined as the cost of the mode itself plus the motion vector cost for that mode. The motion vector cost in turn is calculated using Lagrange multipliers and motion vector magnitudes. Candidate modes for total cost calculations are chosen from the modes of the left, top, and top-left macroblocks of the current one and the macroblock which is two rows down and one column right of the collocated macroblock in the previous frame. The mode with the minimal total cost is chosen as the best mode for the current macroblock. Using this algorithm, a maximum increase of about 15% in bitrate is achieved at the same quality compared with standard. Evidently, the increase in bitrate is significant at the same quality and thus affects negatively the RD performance.

K.P.Lim et. al (2003) propose a new algorithm called "homogeneous regions detection" to classify the inter modes into groups. It is observed that in non-deforming, smoothly moving video sequences, the smooth regions of video objects move together. One of the main reasons for using variable block sizes in H.264/AVC 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 further split into smaller blocks. A region is homogeneous if the texture in the region has very similar pixel values. So some macroblocks which are homogeneous could belong to a specific subset {MODE_16*16, MODE_16*8, MODE_8*16} of modes, and do not need to be motion estimated for the rest of the modes. The authors use the edge map computed in the fast intra mode decision technique of (Pan, 2003) to decide which macroblocks belong to homogeneous regions. If the current macroblock is homogeneous, its mode belongs to the set {MODE_16*16, MODE_16*8, MODE_8*16}, otherwise all modes are tested. The results show a speed up of 30% in run times with a maximum of 0.08dB PSNR decrease and a maximum of 1.44% bit rate increase as compared to the standard. However, this algorithm needs a fixed threshold to decide which macroblock is homogeneous and thus will not be very suitable for different video sequences.

D. Zhu (2004) uses a 7-tap filter on horizontal and vertical directions of the original and reference images respectively to get down-sampled half resolution small images. The mode selection method of (Lim, 2003) is used to get a set of prediction mode candidates in small images. This set of mode candidates is then mapped to another set of mode candidates for the current macroblock in the original image. Because motion estimation is performed in small images, a small set of mode candidates is chosen and thus a lot of time savings can be achieved. The experimental result shows that this algorithm can reduce by nearly 50% the encoding time with PSNR reduction of about 0.2dB as compared to the standard.

C. Grecos and M. Y. Yang (2007) extend (Lim, 2003) into the error domain. The basic idea can be summarized as follows: Video objects in consecutive frames are not always deformed or divided. They may be still or just change location translationally, especially for slow motion video sequences or for the slow motion frame parts of fast video sequences. In terms of computational speed ups, there is potential mis-prediction of modes in macroblocks of those areas from spatial only mode decision techniques such as (Lim, 2003). This occurs in the cases where these macroblocks have high spatial detail and as such will be assigned smaller size modes, whereas by examining error characteristics these macroblocks can be assigned larger size modes. Similarly with (Lim, 2003), the authors check homogeneity but in the error domain. A novel concept of the "moving average sum of amplitudes of edge error vectors" is exploited for designing adaptive thresholds, which makes (Grecos, 2007) suitable not only for the slow motion video sequences but also for the fast motion video sequences. BDPSNR (Bjontegaard Delta Peak Signal-to-Noise Ratio) and BDBR (Bjontegaard Delta Bit Rate) (Bjontegaard, 2001) recommended by JVT are used as metrics to measure the performance difference between methods. Experimental results show that compared with (Lim, 2003), the algorithm gains an average of 12% time savings for the baseline profile with BDPSNR average reduction of 0.04dB and BDBR average increase of 0.43% for the simple profile. For the main profile, a BDPSNR average reduction of 0.03dB and BDBR average increase of 0.07% are observed depending on motion characteristics.

Cost comparisons based on the SATD metric have been used in (Kim, 2004; Tanizawa, 2004) for fast mode decision. The basic idea comes from experiments showing that there is a strong relationship between the costs of low complexity mode decision and the costs of RD based mode decision. In the above algorithms, three most probable modes with lowest costs in low-complexity mode decision are chosen for high complexity mode decision. The disadvantage of these schemes is that mode candidates are known only after all motion estimation has been performed for the low complexity case, thus only part of the mode decision process can be saved time-wise. About 80% of execution time of RD based mode selection can be saved for an average PSNR loss is 0.07dB as compared to the standard.

A fast inter mode decision algorithm is proposed in (Zhou, 2004) by exploiting the correlation of J costs. The basic idea of this algorithm is that if the cost of larger block-size modes is higher than the cost of the current block-size mode, then the best mode of current macroblock cannot be of a larger block-size. Meanwhile, if the cost of a smaller block-size mode is higher than that of current block-size mode, then best mode of current macroblock cannot be of a smaller block-size. A similar idea has been used in (Yin, 2003) which is based on the monotonicity of the error surface as another way to group the modes. The error surface is built initially by 3 modes: MODE_16*16, INTER_8*8 (the entire macroblock is examined using only 8x8 partitions which means four 8x8 sub-blocks for this macroblock), INTER_4*4 (the entire macroblock is examined using only 4x4 partitions which means the entire macroblock is partitioned equally into sixteen 4x4 sub-blocks). If the error surface is monotonic, that is if $J(16x16) < J(8x8) < J(4x4)$ or $J(16x16) > J(8x8) > J(4x4)$ where J denotes a cost function, only modes (block sizes) between the best two modes are tested. If not, all other modes need to be tested. The order of motion estimation suggested by the H.264/AVC standard is MODE_16*16, MODE_16*8, MODE_8*16 and MODE_8*8. And then in each 8x8 sub-block, motion estimations of INTER_8*8, INTER_8*4, INTER_4*8 and 4x4 are performed one by one. In this structure, best motion vectors of neighbor sub-blocks can be utilized to predict the predicted motion vector (the initial position for motion estimation) of current sub-block. However in (Yin, 2003) the order of motion estimation for different partition block size is changed which will introduce complexity and will affect negatively the RD performance.

B. G. Kim (2008) proposed a fast inter mode decision algorithm based on the temporal correlation of mode information for P slices. In the slow motion video sequences, the mode information in the previous reference slice is highly correlated with the mode decision in the current slice. Experimental test shows that if the co-located macroblock is in skip mode, the probability that the current macroblock will be in skip or MODE_16*16 is greater than 91%. When the co-located macroblock is in MODE_16*16, the probability that the current macroblock will be in skip, MODE_16*16, MODE_16*8 or MODE_8*16 modes is greater than 80%. Even when the co-located macroblock is in MODE_16*8 or MODE_8*16, the probability that the current macroblock will be in skip, MODE_16*16, MODE_16*8 or MODE_8*16 is still greater than 70%. Due to this observation, a simple macroblock mode tracking strategy is devised. Initially, motion estimation for the MODE_16*16 is performed and the best match will intersect at most four macroblocks in the reference slice. The most correlated macroblock of the current one is found in the reference slice and based on its mode type, a sub set of candidate modes is checked initially. If the minimum J cost of candidate modes is less than the cost of the tracked macroblock (the most correlated macroblock in the reference slice), the mode of the current macroblock is the one that has the minimum J cost. Otherwise, the sub set of candidate modes is enlarged with the co-located macroblock mode type. Image intensity analysis is also used to refine the choice of candidate modes. The author compared his algorithm with other three fast mode decision algorithms (Jeon, 2003; Jing, 2004; Salagdo, 2006) and showed that he can achieve a good balance between time savings and RD performance. A speed-up factor of 57% on the average was shown, with a bit rate increment of 0.07% and a loss of 0.05dB as compared to the standard.

C Grecos (2005) presented a layered inter mode prediction scheme for P slices. In the first stage, an enhanced fast skip mode decision technique is used. After a percentage of macroblocks in characterized as skipped, the conditions of (Jeon, 2003) are used to identify even more skipped macroblocks in the second stage. For the remaining macroblocks (which also include a percentage of skipped macroblocks that were not classified with the first two stages), (Jeon, 2003) proposed a set of three smoothness conditions. Firstly, the J_{MODE_16*16} cost of the current macroblock should be less than the average J_{MODE_16*16} cost of the macroblocks in this mode in the reference frame. Secondly the co-located macroblock in the reference frame should be of skip mode or of MODE_16*16. Thirdly, the SAD between the current and the co-located macroblock should be less than the average SAD among the skip mode macroblocks in the previous frame and the collocated macroblocks in the current frame. For RD performance very close to the reference encoder, (Jeon, 2003) achieves 35-58% reduction in run times and 33-55% reduction in CPU cycles for both rate-controlled and non-rate-controlled encoding. Compared to (Jeon, 2003), gains of 9-23% in run times and 7-22% in CPU cycles are reported from the scheme of (Grecos 2005). In order to increase the time savings, C. Grecos and M. Y. Yang proposed another algorithm in (Grecos, 2006) which could be considered as an extension of (Jeon, 2003). Their algorithm devised three heuristics instead of three smoothness conditions to be used for predicting subsets of decidable modes. Firstly, the J_{MODE_16*16} cost of the current macroblock should be less than the average J_{MODE_16*16} cost of the macroblocks in this mode in the previous slice of the same slice type. Secondly, for all macroblocks not satisfying the first heuristic, the J_{INTER_8*8} cost of the current macroblock should be less than the average J_{INTER_8*8} cost of the macroblocks in the previous slice that are neither skipped nor were satisfied with the first heuristic. Thirdly, the J_{MODE_16*16} cost should be less than J_{INTER_8*8} for the current macroblock. If any of these three heuristics is satisfied, a subset of candidate modes {SKIP, MODE_16*16, MODE_16*8, MODE_8*16} is assigned to the current macroblock. For the macroblocks that are neither skipped nor belonging to the above subset of modes, the monotonicity property (Zhou, 2004) is used to predict even more macroblocks. For very similar RD performance, (33%-90%) reduction in run times can be achieved as compared to the standard. Compared to (Jeon, 2003; Lee, 2004) that were used as input to the standard, (Grecos, 2006) is faster by 9-23% for very similar RD performance. The ideas in (Grecos, 2006) can be implemented in both the simple and main profiles.

In (Seok, 2008), the authors propose a fast mode decision algorithm using a filter bank of Kalman filters for the H.264/AVC. The basic idea is similar with (Jeon, 2003; Grecos, 2006). A simplified Kalman filter that evaluates an expected RD cost for current macroblock can be built based on the estimated RD cost for the previous macroblock mode, the real RD cost of previous macroblock mode, and the adaptation gain. In order to classify each category, the authors employ three Kalman filters: EJa (the expected RD cost for all macroblock modes together), EJp (the expected RD cost for all inter macroblock modes) and EJi (the expected RD cost for all intra macroblock modes). Macroblock modes can be categorized into four classes based on these three filters and each class contains a subset of all modes. The algorithm has two steps. Firstly the current macroblock is encoded using some candidate mode types of each class. In the second step, if the minimal RD cost of the current macroblock is less than the average RD cost of the mode types of all previously encoded macroblocks in the current frame, the candidate mode set for the current macroblock is reduced. Otherwise, the candidate mode set for the current macroblock is increased. The authors claimed encoding speed ups of about 30% with small degradation of video quality as compared to the standard. Since this algorithm is designed especially for high definition video encoding, it cannot be used for low bit rate and/or low resolution video sequences for two reasons. Firstly the MODE_8*8 is disabled in the technique due to the insignificant effect of the MODE_8*8 in high definition video in high bit rates. But without using the MODE_8*8, the RD performance will be degraded significantly for the low bit rate cases. Secondly, an initial phase of collecting at least 15 macroblock modes is needed before the estimation can be performed by using Kalman filters. For low resolution video sequences which are typical for video on mobile devices, the collection time for filter evaluation is prohibitive thus both reducing overall time savings but more importantly potentially violating real time constraints for bidirectional communications.

The works in (Choi, 2006; Kuo, 2006; Wang, 2007) propose fast inter mode decisions based on the useful information extracted from the motion estimation step. B. D. Choi (2006) contributed a scheme to jointly optimize inter mode selection and ME using multi-resolution analysis. The multi-resolution motion estimation based on the discrete wavelet transform and modified integer transform is employed in the 4x4 sub macroblock level. Different search patterns are chosen in different bands. Subsequently, edge intensity is calculated in each band for each 4x4 sub macroblock. Homogeneity properties are found using linear or quadrature discriminant functions for both the 8x8 sub-macroblock and 16x16 macroblock levels. Candidate modes for the current macroblock are assigned based on this homogeneity information. Experiments are performed in slow and average motion video sequences. The encoding time is reduced by 60% on average with up to 0.15 dB PSNR decrease as compared to the standard. However, no bit rate information is provided. Instead of using multi-resolution motion estimation, (Kuo, 2006) simply used a diamond search (DS) motion estimation algorithm for each 4x4 block of the current macroblock. Sixteen motion vectors called the seed motion field are collected after DS. The Bhattacharyya distance is calculated to measure the separability of motion field classification. If the seed motion field is separable, maximum likelihood classification based on the motion vectors distribution is employed to find which mode is most likely from the set {MODE_16*16, MODE_16*8, MODE_8*16, MODE_8*8}. If the motion field not separable, the predicted RD costs are calculated using the seed motion vectors and the seed motion vector which gives the minimal cost is set as the search center for motion refinement. If MODE_8*8 is selected as the best mode, the motion estimation of block types in the set {INTER_8*8, INTER_8*4, INTER_4*8, INTER_4*4} can be sped up by using a predicted initial search position and an adaptive search range based on the seed motion vectors information. Compared with (Yin, 2003; Kuo, 2006), this algorithm achieved more time savings with similar RD performance. In (Wang, 2007), motion estimation is implemented in the conventional way from 16x16 to 4x4 block sizes. An all-zero coefficient blocks detection technique similar to (Yu, 2008) is adopted during the motion estimation. If all the 4x4 blocks within the current block are determined as all-zero coefficient blocks, the motion estimation is terminated and the rest of modes are skipped. If there are some none zero coefficients in the current macroblock, spatial and temporal homogeneity information with the help of the fixed thresholds is employed to predict the candidate modes. About 52% (Simple Profile) and 44% (Main Profile) of time on the average can be saved with 0.06 PSNR reduction and 0.80% bit rate increase as compared to the standard.

In (Yu, 2008), the authors proposed a hierarchical structure comprising of three layers for fast inter mode decision. The first layer is a fast skip mode detection algorithm using all-zero coefficient blocks as well as the spatial and temporal skip mode information. The central idea in the second layer is homogeneous contents analysis in the DCT domain. High AC components of the current macroblock after the DCT transform indicate a homogeneous macroblock. So the total energy of the AC coefficients of a macroblock is used for classifying it into two categories (low spatial complexity and high spatial complexity) with the help of a fixed threshold after empirical evaluation. Low spatial complexity macroblocks will be assigned the candidate mode set {SKIP, MODE_16*16}, whereas high complexity macroblocks the set {MODE_16*8, MODE_8*16}. From the above two candidate mode sets, if the lowest RD cost mode is in the low spatial complexity set, MODE_8*8 and its sub-modes will not be examined. Otherwise, the algorithm will go to the third layer. In the third layer, motion estimation will be performed in the partition size of the 8x8 block. If the RD cost of the current 8x8 block is bigger than a quarter of the RD cost of the best mode in the previous layers, modes of smaller partition blocks are ignored. The simulations show that up to 75% time savings is achievable with very similar RD performance as compared to the standard.

In summary, due to the extensive use of inter prediction in coded video sequences and the multitude of INTER coding modes in H.264/AVC, a lot of research efforts target fast INTER mode decision algorithms. Despite the very high diversity in techniques that are applied to achieve a fast mode decision, a number of general classes can be identified: techniques that predict the current mode based on spatio-temporal information (e.g., based on SA(T)D, MAD), techniques that use a prediction model based on mode statistics, probabilities, or mode regions, techniques that also incorporate (fast) motion estimation in the mode decision process, and techniques that efficiently predict the bit rate of certain modes (thus eliminating multiple redundant encoding steps). The reported results indicate that overall, gains in the range of 30% - 80% can be achieved by applying fast INTER mode decision. Due to the high diversity in techniques and the different comparison points for complexity and RD performance, it is impossible to state that one class of techniques performs better than another. However, mode decision techniques that use a statistical model to predict modes tend to report slightly higher gains. Also combining motion estimation and mode decision looks very promising.

3.3 Fast Intra Mode Decision

The total number of candidate intra modes for a macroblock is five hundred and ninety two, thus imposing a high computation load of the encoder and triggering of course a flurry of research efforts for the reduction of this load. In (Meng, 2003; Zhang, 2004; Pan, 2003; Fu, 2004; Tsai, 2008; Yang, 2004; Cheng, 2005) fast intra mode decision is based on pixel domain analysis. B. Meng (2003) proposed a fast intra-prediction algorithm for INTRA4*4 modes. According to this work, pixels in 4x4 blocks are categorized into 4 groups and each group is a “down-sampled” version of the original block. The best prediction mode is chosen in a computationally efficient manner by using both SAD and the quantisation parameter to check some of these groups of blocks. Due to the correlation of intra modes directions, the best prediction mode’s two neighbouring directional modes are also chosen as candidate modes, so finally the candidate mode set has cardinality of three. For improving the speed of intra mode prediction, thresholds for early termination are also used. Computation can be saved by not only examining a small set of INTRA4*4 mode candidates but also by using fewer pixels due to down-sampling of the original block. In order to achieve significant time savings and good RD performance, different thresholds for different video sequences based on different quantization parameters need to be set before encoding. Evidently, the setting of thresholds can be problematic in one pass encoding schemes with video of unknown contents.

Y. Zhang’s (Zhang, 2004) algorithm is based on two observations. One is that the best INTRA4*4 mode in 4x4 blocks is highly likely to be in the dominant direction of the local edges. The other is that the DC mode has higher probability to be the best mode compared to other intra 4x4 modes. According to the algorithm, a 4x4 block is initially divided into four 2x2 blocks and feature analysis is performed to find the local dominant edge direction. The local dominant edge direction can be classified into 7 types namely, no obvious edge, vertical edge, horizontal edge, diagonal down/left edge, diagonal down/right edge, vertical-dominant edge and horizontal-dominant edge. A set of modes based on local edge direction information plus the DC mode are chosen to be the set of candidate modes. However there are two problems in the above algorithm which result in increase of bitrates and loss of PSNR as compared to the standard. Firstly the assumption of the best mode being in the dominant edge direction is not always true and secondly analysis of local edge information extraction based on the 2x2 blocks’ intensity can not be very accurate. According to the authors, 40% to 70% of computational complexity can be saved with less than 5.5% of bit rate increase and not more than 0.05dB PSNR degradation as compared to the standard.

Pan’s algorithm (Pan et. al 2003) is based on local edge directional information in order to reduce the amount of calculations in intra prediction. Firstly, the Sobel edge operators (Gonzalez, 2002) are applied to the current frame to generate the edge map. Then 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. The histogram cell with the maximum amplitude indicates that there is a strong edge presence in that direction, and thus it is the direction of the best prediction mode. For increased accuracy, a small number of the most likely intra prediction modes are also chosen for RDO calculation. The drawbacks are similar with (Zhang, 2004). This method shows average gains of 60% in time savings which can be achieved with average increase of 5.9% in bit-rate and induces negligible loss of PSNR as compared to the standard.

In Pan’s algorithm (2003), an edge map needs to be produced for the whole picture and this needs some computation. F. Fu (2004) proposed a faster algorithm which performs only partial edge detection since he observed that mode decision normally depends on the edge information between the left, top blocks and current block. The set of candidate modes is chosen based on Pan’s intra candidate mode selection but using partial edge information. A most probable intra mode type for current macroblock can also be obtained based on the intra mode types and costs of the left and top macroblocks of the current macroblock, and this most probable intra mode type can be used for an early termination criterion to disable INTRA4*4 or INTRA16*16 mode decision. If the most probable intra mode type is INTRA4*4 and the Rate distortion cost of the best INTRA4*4 mode is significant, the INTRA16*16 will be disabled. If the most probable intra mode type is INTRA16*16 and the Rate distortion cost of the best INTRA16*16 is negligible, the INTRA4*4 modes will not be examined. Compared with Pan (2003) this work reduces computation time by a factor of 2 to the expense of very small RD performance degradation.

In (Tsai, 2008) another intensity gradient filter (Ma, 1998) called texture edge flow is adopted instead of the Sobel edge detector. The difference between these two gradient filters is that the Sobel detector only uses two directional gradients to extract edge information but the texture edge flow uses four. This difference makes texture edge flow more suitable for the H.264/AVC than the Sobel detector since the energy information it extracts from different directions is more accurate. By ordering the energy data of different intra modes, a sub-set of candidate intra modes can be selected. A good balance can be achieved between computation complexity and RD performance by selecting an appropriate number of candidate modes. Based on experiments, four modes for INTRA4*4 and 2 modes for INTRA16*16 were found to be a good option. Compared with the standard, the algorithm can achieve around 76% time savings with an average of 0.17 dB PSNR decrease and 2.83% bit rate increase. Authors also claim that their scheme outperforms the work in (Pan, 2003) in terms of PSNR (0.05dB increase), bit-rate (0.66% decrease) and encoding time savings (around 43% savings).

A fast INTRA16*16/INTRA4*4 mode selection algorithm is proposed in (Yang, 2004) which uses macroblock properties. The main idea is that INTRA16*16 modes are more suitable for predicting smooth areas and INTRA4*4 modes can achieve good prediction in regions with significant detail. Based on this idea, pixel level analysis based on thresholds is performed to classify macroblocks into smooth and non-smooth ones and assign to them the aforementioned mode groups. The proposed algorithm can achieve 10%-40% computation reduction with similar PSNR and bitrate performance compared to the standard.

C. Cheng (2005) assumed that the J costs of INTRA4*4 modes are monotonic and proposed a three-step fast intra mode prediction algorithm at the end of which only six INTRA4*4 modes need to be examined instead of nine in the standard. Experiments show that this algorithm can obtain about 31% time savings on the average for intra mode decision, with similar PSNR and about 1% of bit rate increase compared to the standard.

Instead of using spatial (pixel) domain analysis, the algorithms in (Sarwer, 2008; Wang, 2007; Yu, 2005) develop fast intra mode decision algorithms using transformed domain analysis. The work in (Sarwer, 2008) used the SATD metric for some of the INTRA4*4 modes. Due to the high spatial correlation information in the natural video sequences, the probability that the best mode for the current macroblock is the same as the mode of the upper or left macroblocks is very high. The proposed method reduces the number of candidate modes from nine to one based on the combination of the rank of the SATD values for the subset of the examined INTRA4*4 modes and the most probable mode i.e the one of the top or left macroblocks. Thresholds are also needed in this algorithm. Experimental results show that this scheme saves about 70% of the encoding time with 0.06dB loss and 2.24% bit rate increase as compared to the standard. H. M. Wang et al (Wang 2007) proposed a fast intra 4x4 mode decision algorithm based on a fast SATD computation scheme which reduces the full computation of the metric by about 50% The INTRA 4*4 mode decision is further sped up by a two stage simplified scheme. In the first step, only five out of nine possible modes are examined and an extra mode based on the SAD criterion is further examined for the current block in the second step. The achievable time savings are 70% on the average, with 0.05dB quality loss and 0.51% bit rate increase as compared to the standard. In (Yu, 2005), a fast intra mode prediction algorithm is proposed based on a fast partial DCT transform scheme. The DC coefficient and the low-frequency AC coefficients which contain more energy than the high-frequency ones, are calculated using this fast transform scheme for all intra modes. Between one and four modes with the smallest energy plus the most probable mode are selected as the candidate modes for the current block. Compared with Pan (2003) the average time savings are increased and the RD performance is improved.

Changsung Kim (2004; 2006) proposed a new fast intra prediction mode scheme which is a combination of spatial (pixel) and transformed domain analysis. The proposed algorithm adopts a multi-stage mode decision process which uses a spatial domain feature (SAD) and a transform domain feature (SATD) together to remove unlikely intra modes. In the final step of this scheme, a new Rate Distortion model is used to find the best intra mode from the set of intra modes when the QP is larger than a threshold (sixteen in this case). Since the RD model predicts the rate and distortion instead of actually measuring them, some computation can be further saved in the mode decision step due to avoiding macroblock reconstructions. When the QP is smaller than the threshold, full RD-based mode decision (including reconstructions) will be used. Experiments show reduction of the computational complexity of intra mode decision of up to 90%, with little PSNR degradation and at very similar bitrates compared with the standard.

In summary, there are two main approaches for intra mode decision techniques: based on pixel information (e.g., edge detection) or based on transformed domain analysis (e.g., SATD). Because pixel-domain processing also implies some complexity, the reported gains in complexity (for the same RD performance) for pixel-domain techniques (30% - 70%) tend to be lower than those for the transform-domain techniques (50% - 90%). Of course, these complexity gains only apply for intra-only coding and the impact thereof on the overall mode decision process will be much smaller (since the majority of execution time is spent on inter coding).

3.4 Fast Mode Selection Between Intra and Inter Sets of Modes

The selective intra/inter mode decision technique of (Jeon, 2003) contains a spatial (pixel) domain analysis which aids the decision of whether the intra modes for the current macroblock should be checked or not. This technique uses the average boundary error (ABE) between the pixels on the boundary of the current and its adjacent encoded blocks under the best inter mode as an indicator of the degree of spatial correlation and the average rate (AR), i.e., the average number of bits consumed to encode the motion-compensated residual data under the best inter mode as an indicator of the degree of temporal correlation. Subsequently, the average rate for the best inter mode and the average boundary error for the current block are compared. If AR <k*ABE which k is a certain user-defined positive number, there is no need to consider any intra modes for the current macroblock so only inter mode decisions are performed. Otherwise, both inter and intra modes should be considered. Experiments show that about 20% of time savings on the average can be achieved, at the expense of 0.036dB PSNR reduction and 0.7% bitrates increase as compared to the standard.

A feature-based fast intra/inter mode decision method is proposed by Changsung Kim (Kim 2007) to reduce the encoder complexity of the H.264 video coding standard. Three extracted features are considered: the spatial domain correlation found by applying SATD on the intra prediction residual, the temporal domain correlation found by applying SATD on the difference between the current macroblock and the best matched one in the reference slice, and finally the motion vector length. These three features are extracted from the current macroblock to form a 3D (three-Dimension) feature vector. Subsequently, based on the location of feature vector in the feature space, the video slices are partitioned into three regions: risk-free, risk-tolerable and risk-intolerable regions. Depending on the small cells into which the risk region is quantized, the risk minimizing mode can be found. Finally, if the feature vector lies in the risk-free region, the mode decision is made based on simple feature comparison. If the feature vector lies in the risk-tolerable region, the risk minimizing mode is selected. If the feature vector is in the risk-intolerable region, a full RD based mode decision process is performed. Experiments show reductions of about 19-25% of the total encoding time at the expense of 4.1% average rate increase and 0.27% average PSNR loss as compared to the H.264/AVC standard.

4. CONCLUSION AND FUTURE RESEARCH DIRECTIONS

The H.264/AVC standard has shown significant Rate Distortion improvements as compared to other standards for video compression. Special design of the network abstraction layer also makes it more flexible for application to a wide variety of network environments. Considering that transmission bandwidth is still a valuable commodity, H.264/AVC becomes a very promising standard in applications ranging from television broadcast to video for mobile devices. However, high coding performance comes with high computation and power consuming cost, which makes this standard problematic in its use for low delay applications such as video conferencing. In this context, fast mode decision techniques are very important since they enable meeting these low delay requirements. This chapter gives an overview of the state of the art in fast mode decision algorithms. Some of our most important findings are firstly that motion estimation and mode decision are inter-wined and the majority of time savings in mode decision occurs due to the avoidance of motion estimations, secondly that reported gains in speed are often hard to interpret since they depend on the experimental conditions/content and thirdly that the literature rarely compares mode decision methods and whenever this occurs there is not a clear winner,

We would like to conclude by stressing two facts. Firstly, in this chapter we concentrated on algorithmic rather than micro architectural optimizations and added speed-ups and power savings are possible if we take the latter class of optimizations into account. Classic techniques for optimizations at the micro level include module and functional unit level parallelism and clock gating. In module and functional unit level parallelism, different parts of an algorithm and different operations in each module are executed concurrently in a pipelined manner, thus improving the speed significantly. In clock gating, clocks can be deactivated for functions when they are not required, thereby reducing the chip power consumption. Secondly that fast mode decision is still an on-going research area and as such there is a lot of room for improvement but also for extensions in the recent H264 derivatives. A typical example is the lack of much research work in the contexts of H264/SVC (Schwarz, 2007) and H264/MVC (Vetro, 2006), the scalable and multi-view extensions of the H.264/AVC. We hope that this chapter will ignite more research efforts in these directions.

REFERENCES

"Coding of Audio Visual Objects Part 2: Visual", ISO/IEC JTC1, ISO/IEC 14496-2 (MPEG-4 Visual version 1), 1999.

"Generic Coding of Moving Pictures and associated Audio Information. Part 2: Video", ITU-T and ISO/IEC JTC1, ITU-T Recommendation ISO/IEC 13818-2 (MPEG-2) 1994.

"Video Coding for Low Bit Rate Communication," ITU-T Recommendation H263, version 2, January 1998.

Ahmad A., N. Khan, S. Masud & M.A. Maud, (2004, January) "Efficient Block Size Selection in H.264 Video Coding Standard," Electronics letters, vol 40, issue 1.

Bjontegaard G. (2001, April), "Calculation of Average PSNR Differences Between RD-Curves," ITU-T Q6/SG16, Doc. VCEG-M33.

Bystrom M., I. Richardson & Y. Zhao, (2008, February) "Efficient Mode Selection for H.264 Complexity Reduction in a Bayesian Framework," Signal Processing: Image Communication, vol. 23, issue. 2, pp. 71-86.

Chang A., O.C. Au & Y.M. Yeung, (2003, July) "A Novel Approach to Fast Multi-block Motion Estimation for H.264 Video Coding," International Conference on Multimedia and Expo, 2003. ICME'03. Vol 1, 6-9.

Cheng C. & T. Chang, (2005, May) "Fast Three Step Intra Prediction Algorithm for 4x4 Blocks in H.264," IEEE International Symposium on Circuits and System, 2005, ISCAS 2005.

Chia A., Woo Yu, Graham R. Martin & Heechan Park, (2008, February) “Fast Inter-Mode Selection in the H.264/AVC Standard Using a Hierarchical Decision Process,” IEEE Transaction on Circuits and System for Video Technology, Volume 18, No. 2.

Choi B. D., J. H. Nam, M. C. Hwang & S. J. Ko, (2006) “Fast motion estimation and intermode selection for H.264,” EURASIP Journal on Applied Signal Processing, Vol. 2006, pp 1-8.

Fu F., X, Lin & L. Xu, (2004, August) “Fast Intra Prediction Algorithm in H.264/AVC,” 7th International Conference On Signal Processing, Proceedings, ICSP’04, Vol 2, 31Aug-4.

Gonzalez R. C. & Richard E. Woods, (2002) “Digital Image Processing,” Prentice Hall.

Grecos C. & M. Y. Yang, (2007) “Exploiting temporal information and adaptive thresholding for fast mode decision in H264 video coding standard,” in Multidim Syst Sign Process, 18:309-316.

Grecos C. & Mingyuan Yang, (2005, June) “Fast inter mode prediction for P slices in the H264 video coding standard,” IEEE Transaction on Broadcasting, Volume 51, Issue 2.

Grecos C. & Mingyuan Yang, (2006, December) “Fast mode prediction for the Baseline and Main profiles in the H.264 video coding standard,” IEEE Transaction on Multimedia, Vol.8, No. 6.

Haan D., G., Biezen, P.W.A.C., Ojo, O.A., & Huijgen, H., (1993), ‘True motion estimation with 3-D recursive search block matching’, IEEE Trans. Circuits Syst. Video Technol., 3, pp. 368–379, 388

Hsu C.Y , A. Ortega and A. R. Reibman (1997), "Joint Selection of Source and Channel Rate for VBR Video Transmission under ATM policing constraints" IEEE Journal on Sel. Areas in Communications, Special Issue on Real-Time Video Services in Multimedia Networks, Vol. 15, No. 6, pp. 1016-1028.

Source code link: http://iphome.hhi.de/suehring/tml/download/old_jm/ jm81a.zip

Jeon B. & J. Lee, (2003, December) “Fast Mode Decision for H264”, ISO/IEC JTC1/SC29/WG11 and ITU-T SG16, Input Document JVT-J033.

Jing X. & L.P. Chau, (2004, August) “Fast Approach for H.264 Inter Mode Decision,” Electronics letters, Vol. 40, Issue 17.

Jing X. & L.-P. Chau, (2004, September) “Fast approach for H.264 inter-mode decision”, Electron. Lett., Vol. 40, No. 17, pp. 1050-1052.

Kannangara C.S., I.E.G. Richardson, M. Bystrom, J. Solera, Y Zhao, A. MacLennan & R. Cooney, (2006, February) "Low Complexity Skip Prediction for H.264 Through Lagrangian Cost Estimation", IEEE Transactions on Circuits and Systems for Video Technology, vol. 16, no. 2, pp. 202-208.

Kim B. G., (2008, February) “Novel Inter-Mode Decision Algorithm Based on Macroblock Tracking for the P-Slice in H.264/AVC Video Coding”, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 18, No. 2.

Kim C. & C. C. J. Kuo, (2007, April) “Feature-Based Intra/Inter Coding Mode Selection for H.264/AVC,” IEEE Transactions on Circuits and Systems for Video Technology, Vol. 17, Issue 4, pp: 441-453.

Kim C., Hsuan-Huei Shih & C. C. Jay Kuo, (2004, January) “Multistage Mode Decision for Intra Prediction in H.264 Codec,” IS&T/SPIE 16th Annual Symposium EI, Visual Communications and Image Processing, Orlando, Florida.

Kim C., Hsuan-Huei Shih & C. C. Jay Kuo, (2004, October) ”Feature-Based Intra- Prediction Mode Decision for H.264”, IEEE Proceedings of International Conference Image Processing, submitted, Singapole.

Kim C., Hsuan-Huei Shih & C.-C. Jay Kuo, (2006, April) “Fast H.264 Intra-Prediction Mode Selection Using Joint Spatial and Transform Domain Features,” Journal of Visual Communication and Image Representation, ELSEVIER.

Kim H. & Y. Altunbasak, (2004, October) “Low-complexity Macroblock Mode Selection for H.264/AVC Encoders,” IEEE International Conference on Image Processing, 2004, ICIP’04. vol 2, 24-27.

Kuo T. Y. & C. H. Chan, (2006, October) “Fast Variable Block Size Motion Estimation for H.264 Using Likelihood and Correlation of Motion Field,” IEEE Transactions on Circuits and Systems for Video Technology, Vol. 16, No. 10.

Lee J., I.Choi, W.Choi & B.Jeon, (2004, March) “Fast Mode Decision for B slice”, ISO/IEC JTC1/SC29/WG11 and ITU-T SG16, Input Document JVT-K021.

Lim K. P., S.Wu, D.J.Wu, S.Rahardja, X.Lin, F.Pan & Z.G.Li, (2003, September) “Fast Inter Mode Selection”, ISO/IEC JTC1/SC29/WG11 and ITU-T SG16, Input Document JVT-I020.

Ma W.Y. & B. S. Manjunath, (1998, May) “A Texture Thesaurus for Browsing Large Aerial Photographs,” J. Amer. Soc. Inf. Sci, Vol . 49, no.7, pp. 633-648.

Malvar H., M. Karczewicz, & L. Kerofsky, (2003, July) “Low complexity transform and quantization in H.264/AVC,” IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 7.

Meng B., O.C. Au, C. Wong, & H. Lam, (2003, July) “Efficient intra-prediction mode selection for 4x4 blocks in H.264”, 2003 IEEE lot. Conf. Multimedia &Expo (ICME2003). Baltimore. Maryland, USA.

Ostermann J., J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockammer & T. Wedi, (2004) “Video Coding with H.264/AVC: Tools, Performance, and Complexity”, IEEE Circuits and Systems, Vol. 4, No. 1.

Pan F., X.Lin, R.Susanto, K.P.Lim, Z.G.Li, G.N.Feng, D.J.Wu & S.Wu, “Fast Mode Decision for Intra Prediction”, ISO/IEC JTC1/SC29/WG11 and ITU-T SG16, Input Document JVT-G013, March 2003.

Salagdo L. & M. Nieto, (2006, October) “Sequence independent very fast mode decision algorithm on H.264/AVC baseline profile,” in Proc. Int. Conf. Image Process., Atlanta, GA, USA, pp. 41-44.

Sarwer M. G., L. M. Po & Q. M. Wu, (2008) “Fast sum of absolute transformed difference based 4x4 intra-mode decision of H.264/AVC video coding standard,” Signal Processing: Image Communication 23, pp 571-580.

Schwarz H., D. Marpe & T. Wiegand, (2007, September) “Overview of the Scalable Video Coding Extension of the H.264/AVC Standard,” IEEE Transactions on Circuits and Systems for Video Technology, Vol. 17, No.9.

Seok J., J. W. Lee & C. S. Cho, (2008, February) “Fast Block Mode Decision Algorithm in H.264/AVC using a Filter Bank of Kalman Filters for High Definition Encoding,” in Multimedia Systems, Vol. 13, No. 5-6.

Tanizawa A., S. Koto, T, Chujoh, & Y. Kikuchi, (2004, October) “A Study on Fast Rate-distortion Optimized Coding Mode Decision for H.264,” IEEE International Conference on Image Processing, 2004, ICIP’04. vol 2, 24-27.

Tsai A. C., A. P. Paul & J. C. Wang, (2008 May) “Intensity Gradient Technique for Efficient Intra-Prediction in H.264/AVC”, IEEE Transactions on Circuits and Systems for Video Technique, Vol. 18, No. 5.

Vetro A., Y. Su, H. Kimata, & A. Smolic, (2006, October) “Joint Draft 1.0 on Multiview Video Coding,” Doc. JVT-U209 Joint Video Team, Hangzhou, China.

Wang H. M., C. H. Tseng & J. F. Yang, (2007, September) “Computation Reduction for Intra 4x4 Mode Decision with SATD Criterion in H.264/AVC,” Signal Processing, IET, Vol. 1, Issue: 3, pp 121-127.

Wang H., S. Kwong, & C. W. Kok, (2007, June) “An Efficient Mode Decision Algorithm for H.264/AVC Encoding Optimization,” IEEE Transactions on Multimedia, Vol. 9, No. 4.

Wiegan T. & Girod B. (2001, September), “Lagrange Multiplier Selection in Hybrid Video Coder Control,” IEEE International Conference on Image Processing (ICIP’01), Thessaloniki, Greece.

Wiegand T., G.J. Sullivan, G. Bjntegaard & A. Luthra, (2003, July) “Overview of the H.264/AVC video coding standard,” IEEE Transactions on Circuits and Systems for Video Technology, vol 13, no 7.

Yang C., L. PO & W. Lam, (2004, October) “A Fast H.264 Intra Prediction Algorithm using Macroblock Properties,” International Conference on Image Processing, ICIP’04, vol 1, 24-27.

Yin P., H.Y.Cheong, A.Tourapis & J.Boyce, (2003) “Fast Mode Decision and Motion Estimation for JVT/H.264”,IEEE International Conference in Image Processing.

Yu A. C., G. Martin & H. Park, (2005, September) “A Frequency Domain Approach to Intra Mode Selection in H.264/AVC,” in Proc. of 13th European Signal Processing Conference (EUSIPCO) 05, 4PP., Antalya, Turkey.

Zhang Y., F. Dai & S. Lin, (2004, June) “Fast 4x4 Intra-prediction Mode Selection for H.264,” 2004 IEEE International Conference on Multimedia and Expo (ICME2004), Vol 2, 27-30.

Zhou Z. & M.T. Sun, (2004, October) “Fast Macroblock Inter Mode Decision and Motion Estimation for H.264/MPEG-4 AVC,” IEEE International Conference on Image Processing, ICIP’04. vol 2, 24-27.

Zhu D., Q. Dai & R. Ding, (2004, June) “Fast Inter Prediction Mode Decision for H.264,” IEEE International Conference on Multimedia and Expo, ICME '04. vol 2, 27-30.