

A Survey of Fast Inter Mode Decision Algorithms for H.264/AVC Video Encoding System

Badrul Hilmi, Chan-Seob Park and Byung-Gyu Kim
Department of Computer Science and Engineering,
Sun Moon University, Republic of Korea

Abstract: The MPEG-4 Part-10 AVC/H.264 standard employs several powerful coding methods to obtain high compression efficiency. To reduce the temporal and spatial redundancy more effectively, motion compensation uses variable block sizes and directional inter prediction investigates all available coding modes to decide the best one. But when we perform all of variable block sizes, it means very high complexity due to the large number of combinations of coding modes. Thus the decision process requires extremely high computational load. The goal of this study is to review the fast mode decision methods and classify them into different categories. In this survey, we explain the methods in terms of how to determine decision of Macroblock (MB) to reduce its complexity and compare the performance of the well-known algorithms in terms of quality, bitrate and speed of encoding system. We verify several proposed algorithms to achieve speed up factor of with minimal loss image quality and negligible bitrate increment.

Key words: H.264/AVC video, inter mode, fast mode decision, variable block sizes, encoding system, negligible

INTRODUCTION

Video communication is one of the hottest topics in telecommunication and broadcasting industry nowadays. However, raw video streams are difficult to transmit without compression or modification because the limited bandwidth is still the most important bottleneck of data communication. Former video compression standards, such as H.261 and H.263 have to decrease the video quality in order to satisfy the real-time requirement. H.264/AVC (also known as MPEG-4 Part 10) is the latest video compression standard as single-layer video coding which is jointly developed by ISO and IEC (Wiegand *et al.*, 2003). H.264/AVC achieves extremely well encoding performance in terms of video quality and compression ratio than other standards by adopting a number of new techniques: multiple directions of intra-prediction, flexible block size based Motion Estimation (ME) in inter mode prediction, quarter-pel accuracy and Motion Estimation (ME) using multi-references, weighted prediction, Rate Distortion Optimization (RDO). Here, we would like to describe some major features shortly as followings:

Variable block-size motion compensation with small block sizes: This standard supports more flexibility in the selection of motion compensation block sizes and shapes than any previous standard with a minimum luma motion compensation block size as small as 4×4 .

Quarter-sample-accurate motion compensation: Most prior standards enable half sample motion vector accuracy at most. The new design improves up on this by adding quarter sample motion vector accuracy as first found in an advanced profile of the MPEG-4 Visual (part 2) standard but further reduces the complexity of the interpolation processing compared to the prior design.

Multiple reference picture motion compensation: Predictively coded pictures (called P pictures) in MPEG-2 and its predecessors used only one previous picture to predict the values in an incoming picture. The new design extends upon the enhanced reference picture selection technique found in H.263++ to enable efficient coding by allowing an encoder to select for motion compensation purposes, among a larger number of pictures that have been decoded and stored in the decoder.

Weighted prediction: A new innovation in H.264/AVC allows the motion-compensated prediction signal to be weighted and offset by amounts specified by the encoder. This can dramatically improve coding efficiency for scenes containing fades and can be used flexibly for other purposes as well.

Directional spatial prediction for intra coding: A new technique of extrapolating the edges of the previously-

decoded parts of the current picture is applied in regions of pictures that are coded as intra (i.e., coded without reference to the content of some other picture). This improves the quality of the prediction signal and also allows prediction from neighboring areas that were not coded using intra coding (something not enabled when using the transform-domain prediction method found in H.263+ and MPEG-4 Visual). Except the above items, there are more techniques to improve coding efficiency. For more detailed coding tools, you are able to get from (Wiegand *et al.*, 2003). Although, the encoding performance is enhanced, the incurred computational complexity is the new challenge for real-time applications with negligible bit increment and loss image quality. There are many algorithms for reducing the incurred computational complexity of H.264/AVC encoder. In this study, we will review some effective methods for fast inter mode decision scheme and analyze experimental results of well known algorithms.

INTERMODE DECISION PROCESS IN H.264/AVC VIDEO STANDARD

In H.264/AVC, there are in total 7 different block sizes (16×16 , 16×8 , 8×16 , 8×8 , 8×4 , 4×8 and 4×4) that can be used in inter frame motion estimation/compensation (Wiegand *et al.*, 2003). These different block sizes actually form a two-level hierarchy inside a MB. The first level comprises block size of 16×16 , 16×8 or 8×16 . In the second level, the MB is specified as P8x8 type of which each 8×8 block can be one of the subtypes such as 8×8 , 8×4 , 4×8 or 4×4 . The relationship between these different block sizes is shown in Fig. 1.

For each MB, all the sizes are tried and the one that leads to the least RD cost is selected. This try all and select the best philosophy is optimal in deciding the block size for final encoding. Nevertheless, this optimal decision is achieved at the expense of high computational

complexity. However, during experiments on various video sequences using H.264/AVC video coding observed that homogeneous regions prevail in nature video sequences. In addition, many natural video sequences contain stationary regions.

These two types of regions namely spatial homogeneous and temporal stationary are mostly encoded in bigger block sizes such as 16×16 or SKIP mode. Therefore, if we can decide that a MB is homogeneous and/or temporal stationary before encoding, we can safely skip all the other modes and encode this MB using the larger block size such as 16×16 or 8×8 . It can be seen from Fig. 2a that the homogenous areas such as the background, black suit of the man are coded using 16×16 block size.

Although, the boundary area of the lady's suit is non-homogenous with strong edges, the object remains still during some time interval, thus that area is also coded using 16×16 block size because of its temporal stationarity. On the other hand, the dancers in the upper part of the image are relatively smaller and contain much motion, thus they are coded in small size blocks.

Similarly, in Fig. 2b, the homogeneous regions such as the suit, the table, hairs of the man and the woman are coded using large blocks such as 16×16 while the motion regions or motion boundaries such as the hands, the edges of human heads and shoulders are coded in small block sizes. Although the background contains bookshelf that contains much edge information, they are still coded in large block size due to temporal stationarity.

A region is homogeneous if the textures in the region have similar spatial property. There exist many techniques for detecting homogeneous regions in an image. One of those is to use edge information as video object boundary usually exhibits strong edges. The edge detection has already been performed in the fast intra mode decision algorithm (Wu *et al.*, 2005), therefore we can make use of this existing information and to avoid

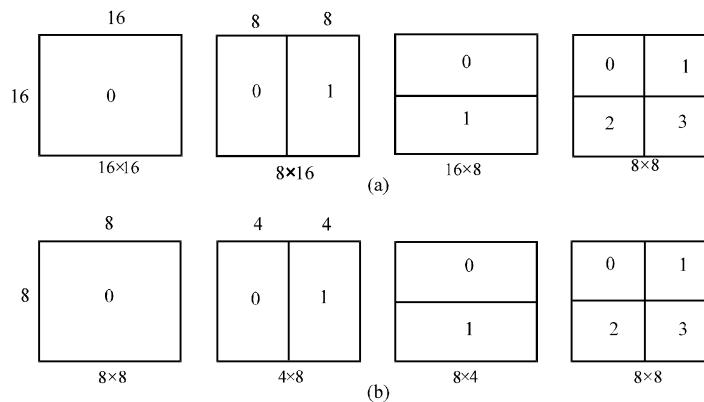


Fig. 1: Different partition in MB: (a) Macroblock partition and (b) Sub-macroblock partition

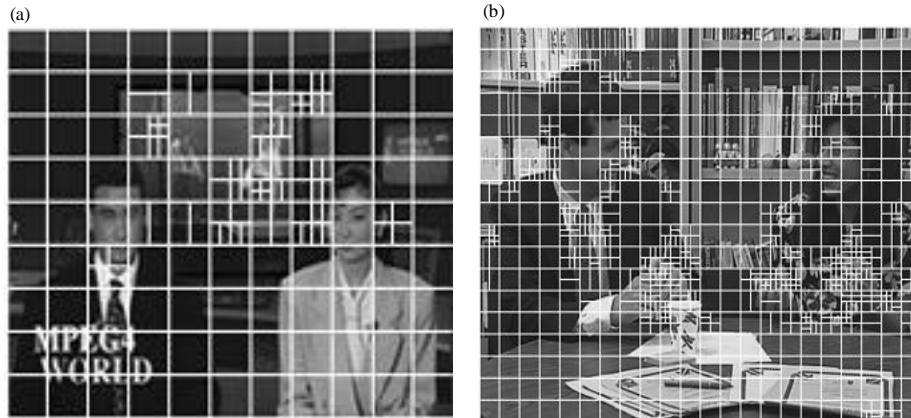


Fig. 2: Examples of block sizes chosen after inter mode search in H.264/AVC: (a) News and (b) paris sequences

extra computation. An edge map is created for each frame in using Sobel operator. For a pixel, in a luminance picture, we define the corresponding edge vector. Whereas homogeneity refers to texture similarities inside a single video frame, stationarity refers to the stillness between consecutive frames in the temporal dimension. In natural video sequences, many image regions especially in the background area, exhibit similar motion even if not still and are thus considered stationary temporally. These stationary regions are normally coded using 16×16 mode after RDO computations.

In the H.264/AVC encoder, the RD cost to decide the best inter-prediction mode is computed as follows (Sullivan and Wiegand, 1998):

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

Where:

- QP = The quantization parameter
- λ_{MODE} = The Lagrangian multiplier
- SSD = The sum of the squared differences between the original block and its reconstruction

$R(s, c, \text{MODE} | \text{QP})$ = The number of bits associated with the mode currently selected for the MB

To find the best coding parameters for each macroblock, H.264/AVC reference software encodes all possible combination of parameters and calculates the rate and distortion of a given macroblock for each combination. This means that the encoder computes the RD costs of all possible coding options and chooses the coding mode of a given macroblock which has the minimum RDcost. However, such mode decision method has critical problem from practical point of view. When a macroblock is given, encoder must have information about required bits and the resulting distortion of the

current coding mode to choose its best coding mode, the information is available only after finishing the encoding process. Therefore, the current H.264/AVC reference software does the complex process for only finding the best coding mode.

H.264/AVC employs a 4×4 integer transform, the RD cost of a macroblock is computed 16 times under the assumption that RD cost is computed on 4×4 block units. In 8×8 sub-partitions, each sub-block can be motion-compensated independently with variable block size such as 8×8, 8×4, 4×8 and 4×4. Thus, each 8×8 sub-block computes its RD cost to decide its own best block mode. To choose the minimum RD cost of an 8×8 sub-partition, the RD cost of each sub-partition is calculated 16 times. Thus, the RD cost is computed 64 times in total for an 8×8 sub-partition mode for a given macroblock. In total, the RD cost is calculated 128 times for all variable block size modes of inter macroblocks.

Here, the crucial thing for applying mode prediction to fast mode decision is to make sure that the predicted mode has the smallest RD cost for a given MB. So far, there have been several ways to decide whether the predicted mode can be assumed to have the smallest RD cost or not. The most common method is to adopt a threshold value derived from the RD costs which are already calculated. The threshold is set to the average of the RD costs of neighboring MBs with identical modes and it is compared with the RD cost of MB with the predicted mode to evaluate if it is the best mode or not. Another method adopts the square of the quantization parameter as a threshold for the RD cost to decide whether the predicted mode is to be used.

EARLY SKIP MODE DECISION IN P AND B SLICES

After computing the RD costs of all coding modes, many macroblocks finally end up with being decided as

Table 1: Probability of SKIP macroblok in P and B slices

Test sequence	P slice	B slice
Container	0.60	0.98
Foreman	0.13	0.55
Mobile	0.59	0.87
News	0.05	0.56
Paris	0.56	0.85
Silent	0.61	0.86
Tempete	0.09	0.64
Average	0.38	0.76

Search range = ±32; Number of reference frames = 5; QP = 28

skipped since they belong to background or a motionless object. Furthermore, many real-world video sequences indeed contain substantial amounts of background and motionless objects.

Table 1 shows that probability of the SKIP macroblock type occurring in P and B slices. This result shows that in the real world, most of macroblocks in P and B slices have SKIP mode as the best block mode: the average probability of SKIP mode is about 0.76 (maximum of 0.98 in Container and minimum of 0.55 in Foreman).

When the SKIP mode is selected, the encoder sends just a skip indicator without transmitting such data as motion vectors, reference frame number, segmented block information and so on. Moreover, if we can decide in advance whether the mode of a macroblock is SKIP mode or not, the wasteful process of computing the RD cost can be omitted so that a huge amount of computation can be saved.

H.264/AVC employs motion compensation using multiple reference frames. An inter-coded macroblock can choose reference frame among predefined numbers of previously encoded frames. The multiple-reference-frame technique provides not only coding efficiency especially when motion is periodic over frames or two different scenes are alternating but also error resilience in error-prone environments. Note that H.264/AVC employs bi-prediction slices (B slice) in the main profile. As in a traditional video coding algorithm, it uses two reference pictures for motion compensation. The two reference pictures can be a pair of backward and forward two forward or two backward pictures. This multiple references frames increases the computational complexity as well, since the motion search is to be performed against all reference frames stored in the picture buffer. The condition of a macroblock to be a skip mode depends on the slice type. In a P slice, the skipped macroblock should satisfy the following conditions:

- The best motion compensation block mode is MB 16×16
- The reference frame is the one closest in the reference frame memory to the current picture
- The motion vector is the same as its Predicted Motion Vector (PMV)
- Its transform coefficients are all quantized to zero

To check these conditions in the P slice, the encoder should know motion vector, reference frame and coded block pattern information for 16×16 block mode. First of all to check conditions, the encoder estimates the motion vector and reference frame for a 16×16 block mode. Subsequently, the RD cost of a 16×16 macroblock is computed.

The calculation of the RD cost includes transform, quantization and entropy coding. After computing the RD cost, the encoder can check the coded block pattern (cbp) information. Therefore, the encoder can check condition. If the SKIP conditions are all satisfied, the other coding modes are simply put aside. In the case of B slices, the skip conditions are as follows:

- The best motion compensation block mode is direct 16×16
- Its transform coefficients are all quantized to zero

In the same way as for the P slice, the proposed method checks whether the mode of the current macroblock can be decided as SKIP in a B slice. To do this, the macroblock is first encoded in the direct mode. The encoder calculates the direct motion vector for the given macroblock. In a direct 16×16 mode of B slice, an 8×8 block size is used in the derivation process for motion vectors and reference frame. Subsequently, the RD cost of the macroblock is computed and checked for the SKIP conditions. After computing the RD cost, the encoder can check the cbp information. If the cbp is zero, its transform coefficients are all quantized as zero. After these processes, the encoder checks SKIP condition.

If the conditions are satisfied, the mode of this block is decided as the SKIP. Even if the conditions of P or B slices are not satisfied, the calculated RD cost of the SKIP mode (P slice) or direct mode (B slice) can be used in a later ordinary selection process of the best block mode. Thus, this proposed method is not an extra burden.

FEATURES IN FAST MODE DECISION

Fast mode decision by grouping candidate mode macroblock: In Fast mode decision, Early decision Macroblock type is very important because reduce the number of computational which caused by un necessary candidate mode decision, Generally several method is using some properties to devide Large macroblock type and Sub-macroblock type, we know that large macroblock type is the highest probability occurrence in encoding process.

For the example, in Jing's algorithm (Jing and Chau, 2004), they divide macroblock types using frame absolute difference and mean absolute difference. Generally, the absolute frame difference contains lots of information about the motions in successive frames. Large amplitudes will appear on the moving edges or boundaries of moving objects while small amplitudes in homogeneous areas. Therefore, if the amplitudes in a MB are small, it is most likely that this MB belongs to a homogeneous region and using only larger block sizes in motion estimation will be accurate. Otherwise, this MB may contain complex motions and using more block types can achieve better rate distortion performance.

In this algorithm proposed fast inter mode decision method depends only the absolute differences between consecutive frames without using multi-stage classifier or edge detection. In the proposed fast inter mode decision algorithm the Mean Absolute Frame Difference (MAFD) of the current frame and Mean Absolute Difference (MAD) of the current MB are used to determine whether the current MB belongs to homogeneous regions or not. They can be calculated using equation below, respectively:

$$\begin{aligned} \text{MAFD} &= \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N |x_{i,j} - y_{i,j}| \\ \text{MAD} &= \frac{1}{256} \sum_{i=1}^{16} \sum_{j=1}^{16} |x_{i,j} - y_{i,j}| \end{aligned} \quad (2)$$

where, $x_{i,j}$ and $y_{i,j}$ denote the intensity levels of pels at location (i, j) of the current frame and its previous frame and M, N are the horizontal and vertical dimensions of the frame.

Weighting factor for determined mode decision based on the Quantization Parameters (QP). Normally smaller QP will result in using smaller block types (8×8 - 4×4 mode) while larger QP will lead to use of larger block types (16×16 - 16×8 mode). Therefore w is used in order to add a bias for using small block types when the QP is small and vice versa. Currently, weighted factor (w) is set as 1.2, 1.0, 0.8, 0.6 for QP = 28, 32, 36, 40, respectively. If the MAD satisfies the following condition:

$$\text{MAD} \cdot w \leq \text{MAFD} \quad (3)$$

This MB is considered as in a homogeneous region and only large block types (16×16 , 16×8 , 8×16) will be used in its motion estimation. Otherwise it belongs to a moving edge region and additional four block types (8×8 , 8×4 , 4×8 and 4×4) will also be enabled to find the best mode.

Other method for determine large macroblock type and sub macroblock type is by detecting macroblock type is a homogenous or not such as in Andi Chia's algorithm (Yu *et al.*, 2008) on his algorithm divide 3 step method in order for fast mode decision with one of his level has target to encoded macroblock by inter modes with a large partition size (8×16 , 16×8 and 16×16 pels). The general tendency that inter modes with large partition sizes are more suitable for the encoding of homogeneous content has been verified by a number of researchers. The reasons given are that:

- Homogeneous macroblocks tend to contain fewer moving features requiring multiple motion descriptors
- Owing to the homogeneous content, the distortion costs arising from incorrect predictions are often insignificant

Consequently, a spatial complexity measurement is developed to determine the content of the macroblocks being considered. It is clear that a non-homogeneous macroblock features significant intensity dissimilarity in the pel domain. This is equivalent to the high-frequency (AC) energy reflected in the DCT-domain. By definition, the total energy of the AC coefficients of a macroblock can be represented by the variance of the macroblock:

$$E_{AC} = \sum_{x=0}^{15} \sum_{y=0}^{15} (w_t(x,y))^2 - \frac{1}{256} \left(\sum_{x=0}^{15} \sum_{y=0}^{15} w_t(x,y) \right)^2 \quad (4)$$

Note that is obtained from a block comprising a checkerboard pattern in which each adjacent pel is the permissible maximum and minimum value alternatively. By using equation above, the value of for a macroblock of size 16×16 is determined to be 15.249. According to empirical evaluation, a macroblock with is considered to be a highly detailed block. Thus, the spatial complexity decision becomes:

$$\text{Complexity} = \begin{cases} \text{high,} & \text{if } E_{AC} > 92735 \\ \text{low,} & \text{otherwise} \end{cases} \quad (5)$$

Low spatial complexity indicates that the current macroblock requires examination by inter-modes with large partition size. However, complexity excludes the case of high detailed macroblocks that can be adequately described by large partitions. Furthermore, it is observed that the mode decision for these macroblocks varies according to the level of compression that is applied, i.e., the large partition inter-modes are to be favoured for coding at high settings. Since the mode decision for a macroblock is determined by the lowest Lagrangian (RD)

Table 2: Classification candidate type macroblock.

Computational complexity	Candidate type	Inclusive mode
Low	SG1	SKIP, 16×16 16×8, 8×16
Medium	SG2	8×8, 8×4, 4×8, 4×4
High	SG3	Intra 4×4, Intra 16×16

cost, suggest computing the RD cost of a few inter-modes before the entire search process is performed. If the best mode for a high-detailed macroblock in this level requires a partition size of 16×8 and 8×16, a more thorough search in the next level is required. Otherwise, the mode decision for the current macroblock is made.

Another example are using Macroblock Detection Level (Huang and Hu, 2009). The main idea of this algorithm is to reduce the number of candidate modes by using early mode detection in four different levels which are SKIP mode detection, type detection at macroblock-level, type detection at sub-macroblock level and intra modes detection. In addition, the correlation of cost of current MB and the one of the previous frame as well as the cost monotonous property are both used to accelerate the encoding process. In this proposed algorithm based on the computational complexity of cost evaluation of each mode can be classified into three candidate types as shown in Table 2.

In this algorithm, SG1 classified as a simple type because the computational complexity of its mode is much lower than SG2's and SG3's. In general, the correlation of the costs of a specific mode for current MB with the average costs of the same mode in the previous frame is high. Based on this property, we compare J 16×16 of current MB with the average Javer 16×16 of this mode in the previous frame. If J 16×16 is less than Javer 16×16, we consider that the best mode of current MB is in the simple type SG1. If J 16×16 = Javer 16×16, steps are carried out to get the Local Motion Activity (LMA) of neighboring Mbs from the Region Of Support (ROS) as shown in Fig. 3, so that further decision can be made for 16×16 mode:

$$\begin{aligned}
 LMV_{i,j} &= |X_{i,j}| + |Y_{i,j}| \\
 LMA &= \max\{LMV_{i,j}, LMV_{i-1,j}, LMV_{i-1,j-1}, LMV_{i+1,j-1}, LMV'_{i,j}\} \\
 \varphi &= \{MV_{i,j}, MV_{i-1,j}, MV_{i-1,j-1}, MV_{i+1,j-1}, MV'_{i,j}\}
 \end{aligned} \tag{6}$$

The type of current MB's best mode still may be simple if it was not detected at macroblock-level, so we detect it at sub-macroblock-level. Firstly, get motion vector of each 8×8 block of current MB with 8×8 mode motion searching. The four motion vectors are mv0, mv1, mv2 and mv3 as shown in Fig. 4. Then we sum up the

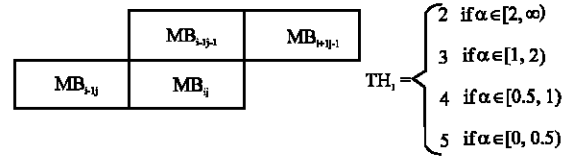


Fig. 3: Part of the region of support of current MB

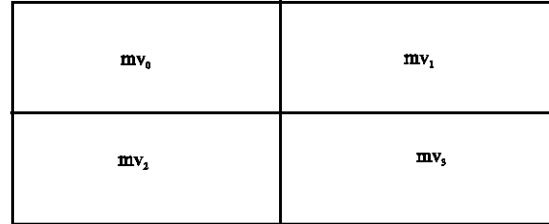


Fig. 4: MV of 8×8 block type

horizontal and vertical distance of adjacent 8×8 block, which are labeled as L1 and L2. The process of calculating L1 and L2 is presented as follows:

$$\begin{aligned}
 mvd_0 &= |mv_{0x} - mv_{1x}| + |mv_{0y} - mv_{1y}| \\
 mvd_1 &= |mv_{2x} - mv_{3x}| + |mv_{2y} - mv_{3y}| \\
 mvd_2 &= |mv_{0x} - mv_{2x}| + |mv_{0y} - mv_{2y}| \\
 mvd_3 &= |mv_{1x} - mv_{3x}| + |mv_{1y} - mv_{3y}| \\
 L_1 &= mvd_0 + mvd_1 \\
 L_2 &= mvd_2 + mvd_3
 \end{aligned} \tag{7}$$

If L1+L2 = TH2 and max (L1, L2) = TH3, the best mode of current MB is in SG1. TH2 and TH3 are empirical thresholds which obtained from a lot of statistical experiments. If the type of current MB is not determined by the methods described above, evaluate the RD costs of 8×8 mode and 4×4 mode (labeled as J 8×8 and J 4×4) and use the monotonous property of the cost functions. According to this property, if J 16×16 < J8×8 < J4×4, the best mode of current MB is in SG1.

Another example of grouping candidate mode using SAD value (Shen *et al.*, 2008) from the observation the SAD value can represent the block mode characteristic. Generally, it can confirm that SKIP mode or large block mode has smaller SAD value than P8×8 sub macroblock modes in most cases. The final block mode may be selected as one of the P8×8 sub-block mode, when the motion or boundary of the block is complex in most cases. Therefore, the SAD value between current block and reference block increases. If there are some specified ranges of SAD values for each block mode, we can easily

determine the proper candidate search modes. Plot the average SAD value of each block mode for many sequences to prove whether each mode has its own SAD value range. The average SAD value of SKIP mode is smallest while the average SAD value of the p8×8 sub block mode is largest. The average SAD values of the 16×16, 16×8 and 8×16 are similar. This means that each block mode can be selected as the best mode more frequently when the SAD value of the block is near the mean of each SAD distribution (Table 3).

Experiments with many sequences and various quantization parameters (QP) show that if the SAD value after motion estimation is within any region, the final mode is highly correlated with the block mode of the boundary of each region. Therefore in this algorithm design adaptive ranges with the average SAD values for block mode decision by grouping the modes which have similar average SAD values. Some types of average SAD values are defined as follows as shown in Table 4.

In B picture coding, the block motion estimation procedure is performed twice for B-slices. The list 0 is generated by the forward motion estimation and the list 1 generated by the backward motion estimation. Because forward and backward motion estimation its very exhaustively complexity to reduce the complexity of the B-slice encoding using a differential mode allocation method using the better list information after 16×16 block motion estimation of the B-slice. After 16×16 block mode motion estimation is performed, get the SAD values of 16×16 modes for list 0 and list 1. If the probability that the best mode exists in the list which has smaller SAD value is high, we can reduce the complexity by allocating the more candidate modes for the better list.

Some of algorithm proposed method of grouping macroblock in encoding process not only divide large macroblock and sub macroblock in inter-mode decision, but also for Intra process like Grecos's algorithm with his Heuristic method (Grecos and Yang, 2007). In this algorithm, introduce two heuristics for predicting a small set of decidable modes, thus achieving significant computational savings due to the avoidance of exhaustive evaluation of all the RD costs. The set of decidable modes in our scheme includes two subsets {SKIP, 16×16, 16×8, 8×16} which requires 4 RD evaluations {INTRA 16×16, INTRA 4×4} which requires 148 RD evaluations (9 INTRA 4×4 modes +16×16 blocks +4 INTRA 16×16 modes) estimation algorithms (FS and FME).

The distinction between the intra and inter modes subsets (including the skip mode) is based on relation of the Average Boundary Error (ABE) with the Average Rate

Table 3: Type of average SAD values

Index	Meaning (N-1: Number of MB)
SAD ^{N-1} _{Mean_SKIP}	The average SAD of the SKIP mode
SAD ^{N-1} _{Mean_Largeblock}	The average SAD of the large block type
SAD ^{N-1} _{Mean_P8x8sub}	The average SAD of the P 8×8 sub-block mode

Table 4: Adaptive range set-up with the average SAD values

Index	Range
Range 1	0 ~ SAD ^{N-1} _{Mean_SKIP}
Range 2	SAD ^{N-1} _{Mean_SKIP} ~ SAD ^{N-1} _{Mean_Largeblock}
Range 3	SAD ^{N-1} _{Mean_Largeblock} ~ SAD ^{N-1} _{Mean_P8x8sub}
Range 4	Over SAD ^{N-1} _{Mean_P8x8sub}

(AR). The selection of inter only modes (including skip) saves 164 RD evaluations while the selection of intra modes after the best inter mode is found saves 16 RD evaluations. The proposed heuristics for the decidable modes are based on the relation between costs of specific modes for the current macroblock with average costs of modes in the previous and previous of the same type frames.

They are applicable in a sequential fashion in both the simple and main profile of the standard (P and B slices). The heuristics used are:

- $J_{MODE_16 \times 16}$ cost of the current macroblock should be less than the average $J_{MODE_16 \times 16}$ cost of the macroblocks in this mode in the previous frame of the same type
- If the current macroblock is not satisfying heuristic, the $J_{INTER_8 \times 8}$ cost of the current macroblock should be less than the average $J_{INTER_8 \times 8}$ cost of the macroblocks in the previous frame that are neither skipped nor were they caught by heuristic. Furthermore, the $J_{MODE_16 \times 16}$ cost should be less than $J_{INTER_8 \times 8}$ for the current macroblock

For the first heuristic, need to stress two points. First, that the RD cost of the MODE_16×16 is known (for the P slice macroblocks) from the skip mode decision part of this scheme, so no extra computation is needed. A single RD evaluation is obviously needed for B slice macroblocks and for both slice types a simple mean cost calculation is also required.

Second, temporal rather than spatial information was chosen for designing this adaptive cost inequality, since although MODE_16×16 is highly likely for smooth areas in the current frame, it is the temporal rather than the spatial information that is more relevant to the J costs.

For the second heuristic, we only need to perform a single RD cost evaluation for 8×8 block size for both slice types, plus to find the average of the $J_{INTER_8 \times 8}$ cost for the

macroblocks of the previous frame that were neither skipped nor caught by heuristic. For the macroblocks that are neither skipped nor they belong to the set of predictable modes, finally evaluate the $J_{\text{INTER}_{4 \times 4 \text{ cost}}}$ and use the property of the cost functions. According to this property, if $J_{\text{MODE}_{16 \times 16}} < J_{\text{INTER}_{8 \times 8}} < J_{\text{INTER}_{4 \times 4}}$ or $J_{\text{MODE}_{16 \times 16}} > J_{\text{INTER}_{8 \times 8}} > J_{\text{INTER}_{4 \times 4}}$. This scheme only need to examine a subset of inter modes plus we need to distinguish between inter and intra modes.

EARLY SKIP MODE IN FAST MODE DECISION

SKIP mode is the large probability mode decision, so how to decide a macroblock become a SKIP mode is very necessary to get the early decision in SKIP mode and also still have good quality in encoding result. Wu *et al.* (2005) proposed a method based on the analysis of the edge map of the entire frame to decide a macroblock is SKIP mode or not. The edge-map information is then used to decide the best edge direction furthermore, they also exploited the edge-map information to determine whether an MB are homogeneous or stationary region for finding the best inter mode. If it is homogenous it not need to process at smaller block size. The edge detection is using Sobel operator and determined the homogeneity by using the amplitude of the edge vector in the block using.

If the sum of the magnitude of the edge vectors at all pel locations in the block is less than Threshold, it is classified as homogeneous block. Otherwise, it is non-homogeneous. Whereas homogeneity refers to texture similarities inside a single video frame, stationarity refers to the stillness between consecutive frames in the temporal dimension. In natural video sequences, many image regions especially in the background area, exhibit similar motion even if not still and are thus considered stationary temporally. These stationary regions are normally coded using 16×16 mode after RDO computations. Thus, we can use the sum of absolute difference to check if this MB changes or not. The difference is defined as follows:

$$\text{Diff} = \sum_{i=1, j=1}^{16 \times 16} \text{abs}(M[i, j] - N[i, j]) \quad (8)$$

where, $M [I, j]$ and $N [I, j]$ represent pel intensities in the previous MB and the present MB, respectively. If the change between the two MBs is less than a threshold, the MB is classified as stationary and 16×16 mode is used for motion estimation, thus all the other modes can be skipped. Based on the experimental results on all the test

sequences, setting for the threshold to 200 achieves good and consistent results for all the test sequences.

Other method of Early SKIP decision is Jeon Algorithm (Choi *et al.*, 2006), this algorithm use ABE and AR properties (will explain below) to determine the SKIP decision. In this proposed fast mode decision algorithm involving two stages of early terminations. In the first stage, the 16×16 block size mode checking is conducted for possible early termination. If such early termination condition is satisfied, then the SKIP mode is set otherwise, the remaining inter modes are checked to find the best inter mode. The second early termination checking is then performed for possibly skipping all the intra modes by evaluating the average boundary errors measured between the pels located at the boundary of the current MB and that of its adjacent upper-left encoded MB.

The small probability of intra mode in real video sequences suggests that the practice of deciding the best intra mode first and subsequent decision of the inter mode may have a certain limit in reducing the computational complexity. Therefore, this scheme propose a more efficient mode decision method, the selective intra coding, that checks intra modes only when it is required after deciding the best inter mode.

The decision method compares temporal correlation with spatial correlation of current macroblocks and investigates various intra modes only when such an investigation is believed to be certainly worthwhile. In this way, the encoder can avoid a number of unnecessary calculations of RD cost values. Note that the underlying principle of inter mode is to make use of the temporal correlation between current and reference pictures and the intra mode is to utilize spatial correlation between the current and adjacent blocks. Therefore, if spatial correlation of a current block is higher than the temporal correlation, the block has a higher probability of being an intra block. A proper decision between the intra and inter modes need some objective measure of spatial and temporal correlation.

To avoid additional computational requirements, the measure will be desirable if it can utilize already available intermediate results. In this case, we can use the Average sum of Boundary Error (ABE) between pels at a boundary of the current and its adjacent upper and left encoded blocks under the best inter mode as indicative of the degree of spatial correlation (Fig. 5). Also, it used 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 indicative of degree of temporal correlation.

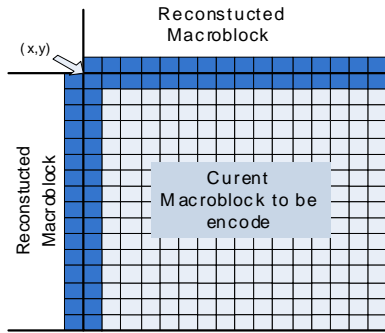


Fig. 5: Pixels involved in calculating ABE

$$AR = \frac{\lambda}{384} \times (\text{number of texture data bits})$$

$$\lambda = 0.85 \times 2^{QP/3} [4], [5], 384 = (16 \times 16)_{LUMA} + (8 \times 8)_{chroma} \times 2$$

This ABE and AR also use at other method of fast mode decision (Liu and Jia, 2009). Other method to make Early decision is by using residual of motion estimation properties (Yang and Chen, 2009). By Using residual motion estimation properties we can also decide mode for sub-macroblock. The residual of motion estimation is the difference of current block and the most matching block in the reference frame. The smaller the residual is the better current block and reference block match. And a small residual implies that current MB is likely to adopt the mode of matching MB's mode, ignoring the other modes.

On the contrary, a bigger residual shows mismatching of current MB and reference MB and implies that the current MB needs more candidate mode(s). On the other hand, variance reflects the coarseness of a texture. A small variance of residual implies a big MB mode (i.e., SKIP, 16x16, 16x8 and 8x16) while a big variance of residual implies a small MB mode (i.e., P8x8). The relationship between Sum of Absolute Difference (SAD) and MB's best mode has been reported by Yang and Chen (2009). Residual's variance and MB's best mode are shown by Yang and Chen (2009). From this result, we can see that there is a close relationship between SAD and MB's best mode. When the SAD is within a small boundary the best mode is only depended on the collocated MB in reference frame. With the increasing of SAD, this relationship becomes weak and the current MB needs neighboring MBs' modes for candidate modes. In comparison with SAD, the variance of residual reveals weaker relationship with MB's best mode. But on this algorithm find that if the variance is very low, the MB's best mode is nearly either SKIP or 16x16. So, utilized that in this algorithm H.264 employs 8x8, 8x4, 4x8 and 4x4 modes in sub-MB coding if the MB adopts P8x8 mode. Previous fast inter-mode decision

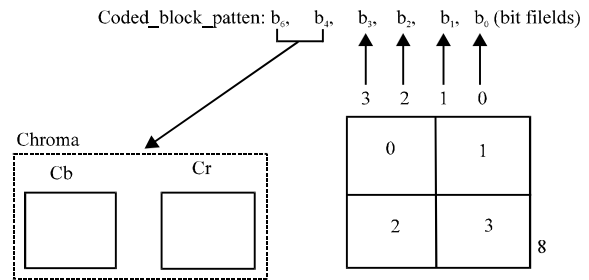


Fig. 6: Bit assignment of coded block patterns

algorithms didn't chose the modes for each sub-MB. Instead, they choose a mode from 8x8, 8x4, 4x8 and 4x4 mode for the whole MB. This processing method may cause more mismatch of mode decision when the motion in the video sequence is high. In order to solve this problem, we suggest a novel adaptive P8x8 mode decision method. If we divide the residual block into 8x8 sub-blocks, the sub-MB is inclined to be coded in 8x8 mode if the SAD of sub-block's residual is small while in 4x4 mode if the SAD of sub-block's residual is big.

To adaptively decide the sub-MB's mode, we noted the 4 sub-blocks of the residual of 16x16 ME as Ri. Then we calculated SAD of Ri, recorded them as SADi and recorded the whole residual block's SAD as SADw. And we used the ratio (λ) of SADi and SADw to decide a sub-MB's mode.

FAST MODE DECISION USING CODED BLOCK PATTERN PROPERTIES

Another fast inter mode decision is use Coded Block Pattern (CBP) Criterion. The Coded Block Pattern (CBP) is a syntax element in the H.264 macroblock layer which specifies which of the six 8x8 blocks (for the Baseline profile) may contain non-zero transform coefficient levels. CBP based criterion method has the following three merits:

- No extra computation is needed (for obtaining CBP)
- No ad-hoc threshold is involved in the algorithm
- The proposed algorithm is simple but efficient and it is highly reliable for various video characteristics

As proposed on Fast inter mode based on Coded Block Pattern (Yang and Wang, 2009), A CBP consists of 6 bits b5b4b3b2b1b0 shown in Fig.6 where b3b2b1b0 (Coded Block Pattern Luma) individually specifies the 4 luma blocks and b5b4 (Code Block Pattern Chroma) jointly specifies the 2 chroma blocks for a 16x16 macroblock with predicted CBP values in the set {0, 1, 2, 4, 8, 16, 32}.

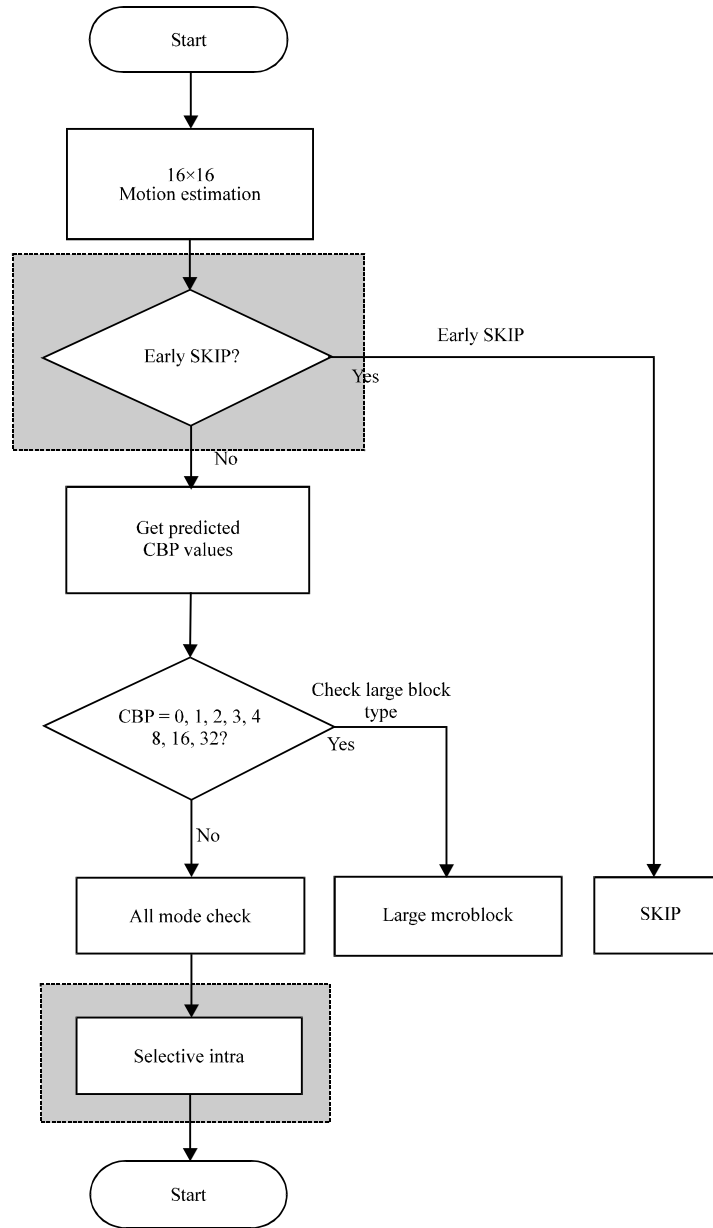


Fig. 7: Flowchart of proposed algorithm MDCBP

This can further integrate this criterion with (Choi *et al.*, 2006). Early-SKIP and Selective-Intra conditions.

The combined algorithm, called the MDCBP (Mode Decision based on Coded Block Patterns). The proposed algorithm shows at flowchart in Fig. 7.

First, check Early Skip (Choi *et al.*, 2006) by motion estimation in 16x16 macroblock, if SKIP mode go to the next macroblock, otherwise get predicted CBP value of current macroblock. If CBP value are 0, 1, 2, 3, 4,

8, 16, 32, process just larger block type, otherwise process all type macroblock. After that, it continues the process with Selective Intra coding (Choi *et al.*, 2006).

FAST BLOCK MODE DECISION BY UTILIZING PROPERTIES OF CO-LOCATED MACROBLOCK

Many algorithm proposed method that use information for current macroblock candidate mode from it co-located Macroblock, it's because many experiment

show that co-located macroblock highly has similarity with current macroblock. Based on that several method to decide candidate mode same type with mode from co-located macroblock. Such as Kim's algorithm with his Direct prediction and Early termination mode (Kim and Kim, 2008), this algorithm uses same mode with co-located macroblock at first decision on detail this Algorithm divided into two steps including direct prediction to determine the initial check mode and early termination to increase the speed. The direct prediction algorithm has two steps. First, determine the initial search mode with correlated MB mode information if the correlated MB mode is included in the large size block mode (Modes 0, 1, 2, 3). Otherwise (sub-macroblock), we use a full mode search. The direct prediction algorithm for the initial search can be described as follows:

- If the co-located MB is SKIP mode, Set the initial MB_{current}: SKIP mode
- If the co-located MB is 16×16 mode, Set the initial MB_{current}: SKIP and 16×16 mode
- If the co-located MB is 16×8 mode, Set the initial MB_{current}: SKIP, 16×16 and 16×8 mode
- If the co-located MB is 8×16 mode, Set the initial MB_{current}: SKIP, 16×16 and 8×16 mode
- If the co-located MB is sub macroblock or intra mode, Set the initial MB_{current}: Fullsearch mode

After this process continue with early termination method for increasing the speed of the mode search procedure. When use only the initial search mode, errors are large. To correct these errors, use a threshold to determine whether to terminate early with the initial search mode or add additional search modes. The threshold used is the rate-distortion cost of the correlated MB. The bottom and right side blocks relative to the correlated MB should be encoded as one of the sub-block modes and the top and left side blocks relative to the current MB should be encoded as one of the sub-block mode.

In other early termination algorithms this operation is performed sequentially from SKIP to sub-block modes using a threshold such as the average rate distortion cost or the Sum of the Absolute Differences (SAD) of the reference frame. But this proposed algorithm checks the mode of the correlated block or a similar block size mode group in the time-successive frame based first on statistical analysis and then checks the other modes. Using this method, the time required for mode checking caused by sequential mode checks is reduced. From the experiment, observed that the inter-best mode block type and the final intra-mode are highly correlated, so the direction of the local texture edge of objects and the inter-mode type usually have a similar direction. If we know the block direction, it is not necessary to search

other directions of the intra-mode prediction except for a dominant direction of the object edge. On the experiment most tested sequences indicate that the DC mode is best for the intra-mode. Thus, use the DC mode as the default primary mode. The horizontal mode uses added rectangular blocks of 16×8 and 8×4 and a vertical mode of 8×16 and 4×8 blocks is also added. Because the inter-mode block type is usually related to the direction of the edge or an object boundary, we can use inter-mode information for the best intra-mode prediction.

Another algorithm that use properties of co-located macroblock is MB tracking in P slice (Byung-Gyu, 2008). This algorithm has proposed an MB tracking strategy scheme which using temporal correlation for all motion cases. The candidate modes of the current MB were selected as modes that are using the mode of the co-located MB. For example, if the mode of the co-located MB is 16×16 candidate modes of the current MB will be SKIP and P16×16. Also, P16×8 if the mode of the co-located MB is 16×8 candidate modes of the current MB will be SKIP, P16×16 and when the mode of the co-located MB is a P8×8 subtype (sub-partition macroblock), it considers to use all eight possible modes as candidate modes. To apply track an object is co-locate the current object region in an adjacent frame this tracking scheme to block-based video coding, need to consider each MB as a desired object in a mode decision procedure.

As shown in Fig. 8, a P16×16 MB type is used to locate the region in the previous frame that has the highest correlation. This is an integer pel motion estimation procedure for the current MB. Once the best motion vector and the most highly correlated region are

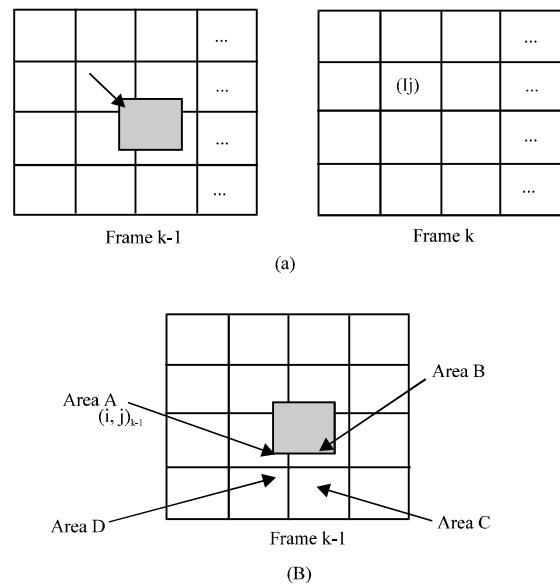


Fig. 8: MB tracking scheme using P16×16 block motion estimation

obtained, determine the most highly correlated MB of the current MB in the previous frame as follows:

$$\text{MAX}_{\text{MB}(k,l)} \{ \text{AreaA}, \text{AreaB}, \text{AreaC}, \text{AreaD} \} \quad (9)$$

where, (k, l) denotes an index of MBs that contain correlated region of the current MB and use the above Eq. 9 to determine the MB that has the maximum correlation with the current MB. Next process is need to define a threshold value (τ) to guarantee a sufficient correlation between the two MBs, this threshold value is desirable to determine the MB when the correlation ratio of the acquired MB to the current MB is greater than the predefined threshold τ . In this scheme, he used threshold $\tau = 0.85\text{--}0.9$. After decide the most correlated MB in the previous frame, the RD cost value of this MB is used for an adaptive decision of early termination for the current MB in the mode search procedure.

In this scheme also used scheme which they called Refinement process, if co-located macroblock is 16×16 the candidate mode setting is SKIP and 16×16 , when RD cost is higher than tracked MB, they compute binary pattern for current macroblock using spatial intensity values. This process is to determine the candidate mode is necessary to use $P8 \times 8$ or not. As shown in Fig. 9a, the refinement process are: set average intensity of each 8×8 block as μ_1 and average value of intensity for the current MB (16×16) as μ_T . If the average intensity of any 8×8 block is larger than μ_T , then set 1. Otherwise set 0. If texture of refinement process like Fig. 9b, the remaining modes for the candidate modes set as $P16 \times 8$, $P8 \times 16$ and $P8 \times 8$ sub types. Otherwise, the candidate modes are $P16 \times 8$ and $P8 \times 16$.

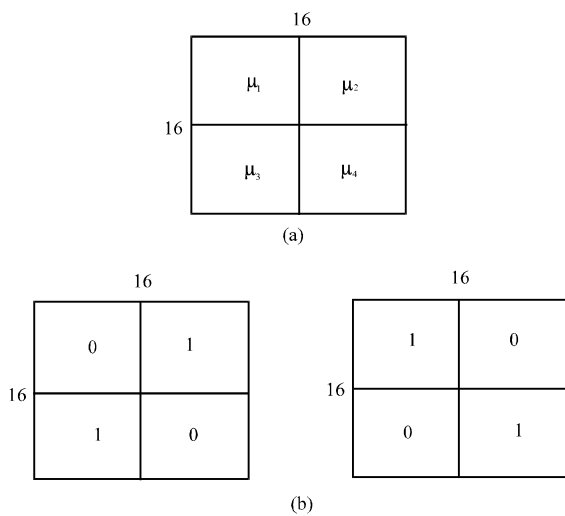


Fig. 9: Binary pattern for the refinement stage of the P- 16×16 mode. (a) 8×8 blocks (b) Binary patterns for searching all modes

Another method that use RD cost to decide mode decision is proposed by (Salgado and Nieto, 2006). This proposed algorithm aims to appropriately decide which MBs need not to be computed with the complete analysis of modes. The ME process sequentially tests modes and the FMD algorithm decides if more modes need to be computed.

The decisions are taken based on the RD results obtained for the previous analyzed modes and the RD results of the co-located MBs, those placed at the same position at the previous encoded frames, the process are firstly computes the SKIP Mode with this mode, motion is directly obtained from previously encoded information, available at the decoder side, without performing any search at the ME process and therefore, without sending motion information or residual data to the decoder. The algorithm decides to stop computing modes if the RD cost of the SKIP Mode satisfy the following condition:

$$J_{\text{SKIP}} < J_{\text{Co-located}} \quad (10)$$

Where:

- J_{SKIP} = The RD cost of the SKIP Mode computed as in Eq. 1
- $J_{\text{co-located}}$ = The lower RD cost among the collocated MBs of the previous encoded frames, updated at each IDR (Instantaneous Decoder Refresh) frame

This condition ensures that SKIP Mode obtains a RD cost lower than the best RD cost of the previous encoded Mbs. If SKIP Mode does not satisfy this condition, Mode 1 is computed and J_{SKIP} is tested against J_1 :

$$J_{\text{SKIP}} < J_1 \quad (11)$$

Where J_1 is the RD cost of the Mode 1. If this condition is satisfied, SKIP Mode is selected as the best mode and the ME process ends.

If Mode 1 is better than SKIP Mode, then its RD cost is compared with the lower RD cost of the co-located Mbs:

$$J_1 < J_{\text{Co-located}} \quad (12)$$

Analogously to Eq. 2, if this condition is satisfied, Mode 1 is selected as the best mode and no more modes are analyzed. If $J > J_{\text{co-located}}$, Modes 2 and 3 are computed and its RD costs are compared with the RD cost of Mode 1:

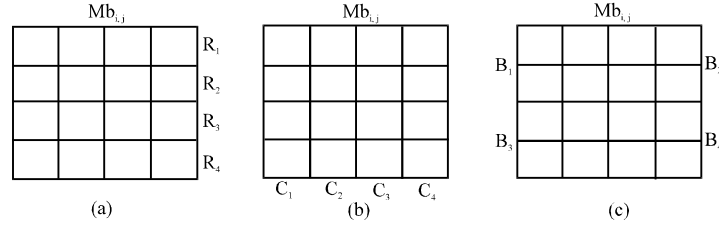


Fig. 10: Partitions used in calculating three directional motion homogeneity measures

$$J_1 < \{J_2, J_3\} \quad (13)$$

where, J_2 and J_3 are the RD costs of Mode 2 and 3, respectively. If this condition is satisfied, Mode 1 is chosen as the best mode and the ME process ends. After testing all these conditions, if none is satisfied, the FMD algorithm ends and the ME process analyzes the rest of modes. The value of $J_{co-located}$ is updated only if the RD cost of the best mode is lower than the previous value of $J_{co-located}$.

MODE DECISION BASED ON MOTION OF MACROBLOCK

Generally motion of a macroblock can represent the characteristic of candidate mode in mode decision. Some algorithm proposed by evaluate the motion homogeneity of a macroblock (Liu *et al.*, 2009). This method proposed because there exists a higher correlation between the motion homogeneity exhibited in the MB and its optimal inter mode selected using full inter mode decision. Inspired by the tree-structured block sizes defined in H.264/AVC, this algorithm propose three directional motion homogeneity measures namely, horizontal, vertical and quartered which are exploited for MB classification.

Motion estimation is first performed on the block size of 4×4 and then a normalized MV field at 4×4 block level is generated for calculating the motion homogeneity of each MB. For each 4×4 block in the current frame (frame number), the normalized MV is calculated based on the temporal distance and the direction indicated by the reference frame index from and so as to make the equivalent reference frame for each 4×4 block is the previous frame in the resultant normalized MV field. For a 4×4 block in the current frame, assume that the MVs from and are denoted as and (only for B-frames), respectively. The normalized MV for is then defined as where and are the reference frame index, respectively.

Assume that a MB located at the i th row and the j th column is denoted as $MB_{i,j}$ and the normalized MVs of its covered 4×4 blocks are thus denoted as $NMV_{m,n} = \{mvx_{m,n}, mvy_{m,n}\}$, $m \in [4i, 4i+3]$, $n \in [4j, 4j+3]$ shown in Fig. 10, the

mean deviation of MVs in each partition which may represent each row in Fig. 10a or each column in Fig. 10b or each 8×8 block in Fig. 10c is defined as:

$$MD(P_k) = \frac{1}{4} \sum_{R_k} \{ |mvx_{m,n} - \frac{1}{4} \sum_{R_k} mvx_{m,n} + |mvy_{m,n} - \frac{1}{4} \sum_{R_k} mvy_{m,n} | \} \quad (14)$$

Then the horizontal, vertical and quartered motion homogeneity of are defined as:

$$H_{i,j} = \frac{1}{4} \sum_{k=1}^4 MD(R_k), \quad V_{i,j} = \frac{1}{4} \sum_{k=1}^4 MD(C_k), \quad (15)$$

$$Q_{i,j} = \frac{1}{4} \sum_{k=1}^4 MD(B_k)$$

Based on the above three directional motion homogeneity measures, each MB is classified into one of the following five classes when the specified condition is satisfied.

Class A: The motion is completely homogenous in each direction:

$$H_{i,j} < T_s \text{ and } V_{i,j} < T_s \text{ and } Q_{i,j} < T_s \quad (16)$$

Class B: The motion is complex and exhibits no obvious homogeneity in any direction:

$$H_{i,j} > T_s \text{ and } V_{i,j} > T_s \text{ and } Q_{i,j} > T_s \quad (17)$$

If does not satisfy the above two conditions, it is further classified into one of the following three classes:

Class C: The motion is more likely to be homogenous in the horizontal direction:

$$H_{i,j} < V_{i,j} \text{ and } H_{i,j} < Q_{i,j} \quad (18)$$

Class D: The motion is more likely to be homogenous in the vertical direction:

$$V_{i,j} < H_{i,j} \text{ and } V_{i,j} < Q_{i,j} \quad (19)$$

Class E: The motion is likely to be homogenous in the horizontal or/and vertical direction when one of the following two conditions is satisfied:

$$\begin{aligned} Q_{i,j} < H_{i,j} \text{ and } Q_{i,j} < V_{i,j} \\ H_{i,j} = V_{i,j} \text{ or } H_{i,j} = Q_{i,j} \text{ and } V_{i,j} < Q_{i,j} \end{aligned} \quad (20)$$

The threshold is set to 0.1 in order to tolerate one noisy MV in the motion homogenous MB. The noisy MV is different from the other 15 MVs by only one 1/4-pel in either horizontal or vertical direction and is selected to strictly satisfy the above limitation is set to 0.5 by extensive experiments and this value achieves a good and consistent performance on a variety of video sequences with different motion activities.

At different case, a motion of macroblock also can be use to decide macroblock type by evaluate the RD cost (Zeng *et al.*, 2009). In general the motion activity of an MB is intimately related to its RD cost and this relation can be exploited on the design of the mode decision algorithm as follows. If the RD cost computed at the SKIP mode is small enough (next called threshold, T_{low}), then the current MB is probably motionless. Thus, the SKIP mode should be selected as the best mode and the mode decision process proceeds to the next MB. In fact, this is equivalent to the so-called early termination method often being implemented in fast motion estimation process.

On the other external end, if the RD cost computed at the SKIP mode results in a fairly large value (next called threshold, T_{high}) this situation signifies that the current MB quite likely involves a highly-textured region and even with a fast motion or at a scene cut. Therefore, the two intra modes from Class 5 are the most likely candidate modes and should be further checked by computing their RD costs to see which one yields the least cost as the best mode to be assigned as shown in Table 5.

How to determine these two thresholds, T_{low} and T_{high} which are investigated as follows: these two thresholds must be QP dependent. As specified in H.264, the range of QP values is from 0-51 and only the integer values from this range can be used in the JM reference software. For the delivery of robust fast mode decision performance, the threshold values of and play a crucial role on the entire mode decision process.

For that, all the MBs from a set of commonly used test sequences are employed to empirically determine the reliable threshold values for T_{low} and T_{high} with a goal of achieving 90% degree of confidence. In other words, if the RDcost (SKIP) of the current MB yielded is smaller than

Table 5: The motion activity classes and their involved modes

Class	Motion activity	Involved modes
1	Motionless	SKIP
2	Slow motion (homogenous region)	SKIP, 16×16
3	Moderate motion	16×8, 8×16
4	Fast motion	P8×8 (8×8, 8×4, 4×8, 4×4)
5	Highly-textured region in fast motion or with scene changes	I4MB, I16MB

T_{low} , the SKIP mode is assigned to the current MB as its best mode and the probability that the SKIP mode is indeed the best mode determined by the exhaustive mode decision is 0.9.

On the contrary, if the RDcost (SKIP) of a MB is larger than T_{high} , either I4M or I16M whichever yielding a smaller cost will be assigned to the current MB as its best mode in this case, the probability that either I4 or I16M is indeed the best mode is 0.9. By calculating the threshold values of T_{low} and T_{high} versus the various QP, the result of both threshold are:

$$\begin{aligned} T_{low} &= 34e^{0.1759 \times QP} \\ T_{high} &= 24215e^{0.00675 \times QP} \end{aligned} \quad (21)$$

Another case of motion activity of a macroblock is by evaluating the spatial property of motion field (Shen *et al.*, 2008). One of the reasons for adopting variable size motion estimation in H.264/AVC inter prediction is to capture the true motion in nature video and represent object movement more accurately so as to reduce the residual energy when the fixed block size prediction is used.

Usually choosing a large block size means that a small number of bits are required to signal the choice of motion vector(s) and the type of prediction mode but the motion compensated residual may contain a significant amount of energy in region with motion spatial edge or discontinuities.

Choosing a small block size may give a lower energy residual after motion compensation but requires a larger number of bits to signal the motion vectors and type of prediction.

There is much redundancy to static and motion spatial continuity regions. Ideally, it would like to represent motion on a coarse basis and in static and smooth regions. Otherwise, we would like to represent motion in a finer level in regions of motion edges or discontinuities.

Region with motion continuity is more likely to select large block size and the region with discontinuities is more likely to select small size. Thus, motion continuity is good indication in choosing the best inter mode size and thus can be used to skip unnecessary mode size so as to speed

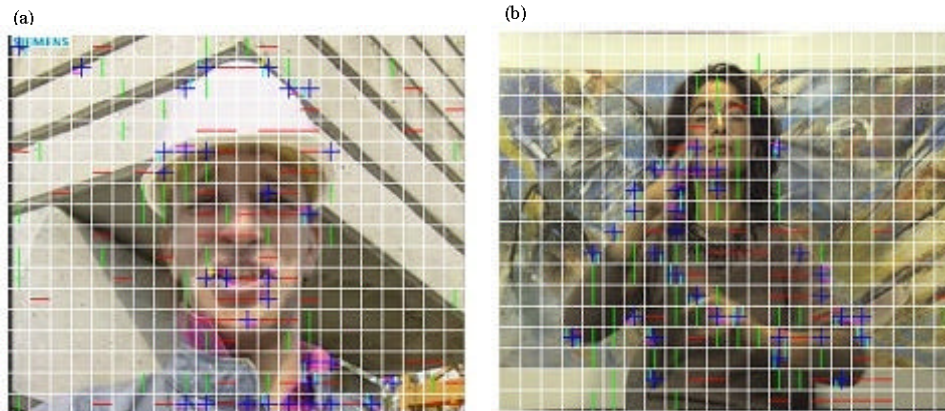


Fig. 11: (a) 54th frame foreman CIF and (b) 205th frame silent CIF

the procedure of motion estimation and RD cost computation. Two example frames are shown in Fig. 11 in which the block size selected using full mode decision is represented by different sized boxes overlaid on the corresponding MB.

Figure 11a shows the 54th frame from the CIF sequence foreman in which the background is moving due to the camera panning operations. The background region exhibits a continuous movement and thus most MBs in the still wall region select 16×16 . Besides, most MBs in the interior regions of the face and body also select 16×16 since they have continuous motion. On the contrary, most MBs in the boundary regions of face, mouth and body are more likely to select smaller block sizes. Figure 11b is the 205th frame from the CIF sequences Silent which is a standard head and shoulder sequence with static background. It can be seen that nearly all MBs in the static background are coded using 16×16 and most MBs in the interior regions of face and body with slight homogenous motion are coded using block sizes $> 8 \times 8$.

However, MBs located at the boundary between objects with different motion activities such as the hand and arm regions (fast motion) and the head boundary (slow motion) are more likely to select the smaller block size 8×8 . No matter whether the sequence is captured by a static or moving camera, we found that MBs in the motion continuous regions have a higher probability to be coded using larger block sizes including 16×16 , 16×8 and 8×16 . The above observation is also validated on a variety of video sequences with different motion activities and image contents.

It is further observed that many real-world video sequences indeed contain substantial amounts area of static or continuous motion. After computing the RD

Table 6: Probability of each block size chosen in P slices

Test sequence	Frame rate (Hz)	16×16 (%)	16×8 (%)	8×16 (%)	8×8 (%)
Miss-am (QCIF)	30	87.5	5.2	6.5	0.9
Grandma (QCIF)	30	89.0	4.7	5.4	0.8
Football (QCIF)	30	53.3	8.0	11.2	24.5
Foreman (CIF)	15	68.1	12.8	13.3	5.8
Costguard (CIF)	15	59.2	15.2	15.0	10.6
Tempete (CIF)	15	49.1	16.6	14.7	19.7
Silent (CIF)	15	82.3	5.9	7.3	4.5
Average	-	69.8	9.8	10.5	9.9

costs of all coding modes, most MBs in these areas finally end up with being decided as large block size. Table 6 shows the probability of each block size chosen in P slices. The result shows that most of MBs have chosen 16×16 as the best block size.

The average chosen probability of size 16×16 is about 69.8% (maximum of 89% in Grandma and minimum of 49.1% in Tempete) and 90% MBs select larger block sizes (16×6 , 16×8 and 8×16). Based on the above analysis if it can exploit MB motion continuity to determine those MBs that have a large probability to be coded using larger block sizes, we can skip RD cost computation and motion estimation on smaller size and thus significantly reduce the computation complexity of the encoding process.

A region with motion continuity means that the motions in the region have homogenous spatial property. There exist many techniques for detecting motion homogeneity. The simple statistical measurement such as standard deviation and variance is a good way of determining homogeneity. But, these techniques cannot distinguish the real edge of motions and the random motion caused by the limitation of the block motion estimation. In this algorithm, motion estimation is first performed on size 4×4 and motion vectors from inter 4×4 compose motion field.

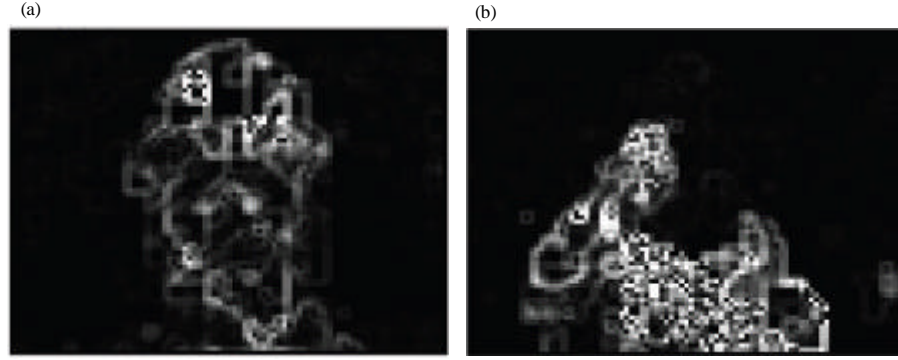


Fig. 12: Magnitudes of the motion edge vectors (a) 54th frame foremen and (b) 205th frame of the sequence client

The motion edge map is created for each frame by applying the Sobel operator to the horizontal and vertical components of the motion field. Assume that a MB located at the m th row and n th column is denoted as MB_m ; n and the Motion Vectors (MVs) of its covered 4×4 blocks are thus denoted as $MV_{i,j} = \{mvx_{i,j}; mvy_{i,j}\}$; $I \in [4m; 4m+3]; j \in [4n; 4n+3]$. We define the 4×4 block $(i; j)$ motion edge vector $D_{i,j} = \{dxi_{i,j}; dyi_{i,j}\}$ as:

$$\begin{aligned} dxi_{i,j} &= |mvx_{i-1,j+1} + 2 \cdot mvx_{i,j+1} + mvx_{i+1,j+1} \\ &\quad - mvx_{i-1,j-1} - 2 \cdot mvx_{i,j-1} - mvx_{i+1,j-1}| \\ &\quad + |mvy_{i-1,j+1} + 2 \cdot mvy_{i,j+1} + mvy_{i+1,j+1} \\ &\quad - mvy_{i-1,j-1} + 2 \cdot mvy_{i,j-1} + mvy_{i+1,j-1}| \\ dyi_{i,j} &= |mvx_{i+1,j-1} + 2 \cdot mvx_{i+1,j} + mvx_{i+1,j+1} \\ &\quad - mvx_{i-1,j-1} - 2 \cdot mvx_{i-1,j} - mvx_{i-1,j+1}| \\ &\quad + |mvy_{i+1,j-1} + 2 \cdot mvy_{i+1,j} + mvy_{i+1,j+1} \\ &\quad - mvy_{i-1,j-1} - 2 \cdot mvy_{i-1,j} - mvy_{i-1,j+1}| \end{aligned} \quad (22)$$

where, $dxi; j$ and $dyi; j$ represent the degrees of motion differences in vertical and horizontal directions, respectively. Small $dxi; j$ and $dyi; j$ mean that the 4×4 block has the consistence motion with the contiguous blocks and locates in the motion continuous region. Large $dxi; j$ and $dyi; j$ mean it locates in the disorder region. It should be noted that Sobel operator is applied to every 4×4 block except those block on borders of pictures.

This is because the operator cannot apply on those blocks without eight surrounding blocks. Since the 4×4 blocks on borders of picture mostly are located in background and their motion are continuous, we define the motion edge vectors $\{dxi; j; dyi; j\}$ of these blocks with $\{0; 0\}$. The edge vector of $MB_m; n\{DX; DY\}$ is determined by edge vector of the 4×4 blocks in the MB and it is computed as follows:

$$\begin{aligned} DX_{m,n} &= \frac{1}{16} \sum_{i=4m}^{i=4m+3} \sum_{j=4n}^{j=4n+3} dx_{i,j} \\ DY_{m,n} &= \frac{1}{16} \sum_{i=4m}^{i=4m+3} \sum_{j=4n}^{j=4n+3} dy_{i,j} \end{aligned} \quad (23)$$

Figure 12 shows the amplitudes of MB motion edge vectors detected by sobel operator. One can see from Fig. 11 that MBs in the edge of a moving object have larger magnitude while the magnitudes of MBs in the interior of a moving object or the background are smaller.

When DX is smaller than $ThdL$, the MB is considered with continuous motion in vertical direction and when DX larger than $ThdH$; it is considered with complex motion. Otherwise, it is considered as the medium continuous motion. The same classification is also performed on DY . $ThdL$ and $ThdH$ are motion edge detector-dependent thresholds. The thresholds are determined by analyzing the motion continuity in a frame. That is $ThdH$ value is set to the average motion edge value (D_{ave}) and $ThdL$ value is set to $0.25D_{ave}$. D_{ave} is given by:

$$D_{ave} = \frac{1}{M \times N} \cdot \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} (DX_{m,n} + DY_{m,n}) / 2 \quad (24)$$

where, $M \times N$ is the total number of MBs in a frame and M and N denote the number of the row and column in term of MB, respectively.

The block size of 16×16 is more suitable for encoding the areas with continuous motion in both horizontal and vertical directions and areas not containing multiple objects or lying on the boundary of a moving object. The block size of 16×8 is more suitable for areas belonging to one object in horizontal direction and the areas are with continuous horizontal motion and complex vertical motion. The block size of 8×16 is more suitable for areas belonging to one object in vertical direction and the areas are with continuous vertical motion and complex horizontal motion.

However, the block size of 8×8 is more suitable for areas containing multiple objects or lying on the boundary of a moving object and the areas are likely with complex motion in both horizontal and vertical directions. When MBs are with continuous motion in both horizontal and vertical directions, these MBs are suitable for 16×16 and motion estimation on small block sizes could be avoided. When MBs are with continuous motion in horizontal direction, these MBs are suitable for 16×16 and 16×8 and motion estimations on 8×16 and 8×8 could be avoided. When MBs are with continuous motion in vertical direction, these MBs are suitable for 16×16 and 8×16 and motion estimations on 16×8 and 8×8 could be avoided. When MBs are with medium continuous motion, size 8×8 is chosen with very small percentage, thus motion estimation on size 8×8 could be avoided. When MBs are with complex motion, size 8×8 is chosen with large percentage while the percentages of 16×8 and 8×16 selected cannot be negligible.

FAST MODE DECISION BY REDUCE TRANSFORM PROCESS

As usual in many method of algorithm in Fast mode decision, the first process in mode decision is to identify the macroblocks to be encoded with SKIP and the threshold methods achieve fast detection for skipped-macroblocks. Considering that skipped macroblocks tend to occur in clusters such as in a patch of static background, the temporal similarity examination is applied if one of the following conditions is valid:

- The collocated macroblock in the reference frame is encoded with SKIP mode
- At least one of two possible valid skipped macroblocks is found above or to the left of the current macroblock

The decision process for the temporal similarity detection is defined as:

$$\text{Decision} = \begin{cases} 1, & \text{SAD}(w_t^{(m,n)}) \leq \text{Thd}_A \\ 0, & \text{otherwise} \end{cases} \quad (25)$$

Where represents the Sum of Absolute Difference (SAD) reflects the residue macroblock generated by the current macroblock and its collocated macroblock in the reference frame is an adaptive threshold obtained from the average SAD values of the available skipped neighbors in the current frame and the reference frame. The corresponding locations of the neighbors are shown in Fig. 13. A nonzero outcome in indicates that the current block is a potential skipped-macroblock. A further

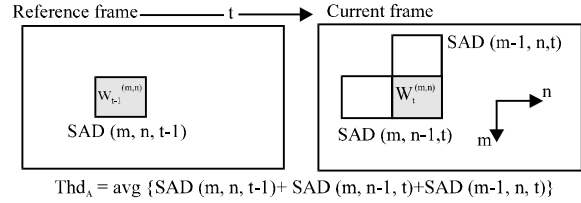


Fig. 13: Neighboring macroblocks located in current frame and reference frame used for computation of Thd_A

examination of its transform-quantized coefficients is required. This transform process can be reduce by SKIP and Pick strategy to reduce it time examination as proposed in Andy Chia algorithm (Yu *et al.*, 2008). As the residue blocks have previously passed the threshold examination, they are unlikely to possess high-frequency energy. Thus, the proposed fast implementation is achieved by examining a limited number of low-frequency coefficients for instance, the first three coefficients in a zigzag scanned manner:

$$\text{abs} \left(\begin{bmatrix} W_{\text{DCT}}(0,0) \\ W_{\text{DCT}}(0,1) \\ W_{\text{DCT}}(1,0) \end{bmatrix} \right) < \begin{bmatrix} T_{\text{Qz},0} \\ T_{\text{Qz},1} \\ T_{\text{Qz},1} \end{bmatrix} \quad (26)$$

Where is the absolute operator and is a transform quantized threshold fixed for a specific factor. Table 7 shows the values of and with respect to several values commonly employed in video compression. The values of the transform coefficients:

$$\begin{aligned} W_{\text{DCT}}(0,0) &= \sum_{x=0}^3 \sum_{y=0}^3 w_t(x,y) \\ W_{\text{DCT}}(0,1) &= \sum_{x=0}^3 (w_t(x,1) - w_t(x,2)) + \left\{ \sum_{x=0}^3 (w_t(x,0) - w_t(x,3)) \right\} \ll 1 \\ W_{\text{DCT}}(1,0) &= \sum_{y=0}^3 (w_t(1,y) - w_t(2,y)) + \left\{ \sum_{x=0}^3 (w_t(0,y) - w_t(3,y)) \right\} \ll 1 \end{aligned} \quad (27)$$

Note that the transform-quantized scheme in the H.264/AVC standard works on the basis of 4×4 pels. For a macroblock (of size 16×16), the fast transform-quantized evaluation has to be repeated 16 times. Since pel information has a strong correlation with its neighbors, a skip-and-pick strategy depicted in Fig. 14 transform above is incorporated. By selecting one of four adjacent pels, the encoder collates 16 pel representatives required for the transform-quantized evaluation from an 8 by 8 block. This reduces the procedure to only four examinations. The efficiency of the skip and pick strategy has been studied extensively in many literature.

Table 7: Values of $T_{Qz,0}$ and $T_{Qz,1}$ with respect to QP factor

QP	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
$T_{Qz,0}$	21.7	23.3	26.7	30.0	33.3	36.7	43.3	46.7	53.3	60.0	66.7	73.3	86.7	93.3	106.7
$T_{Qz,1}$	33.3	37.5	41.7	47.9	54.2	58.3	66.7	75.0	83.3	95.8	108.3	116.7	133.3	150.0	166.7

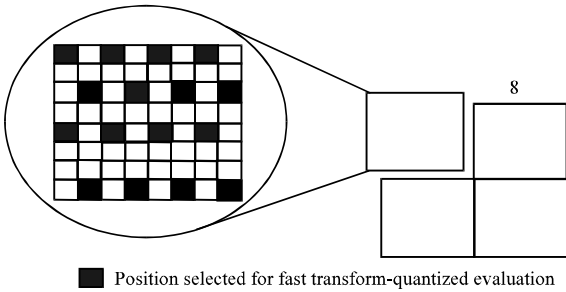


Fig. 14: Skip and pick strategy to reduce time for examination of transform-quantized coefficients in macroblock

EXPERIMENT RESULT

To verify the several proposed scheme, a comprehensive set of experiments for a variety of video sequences with different motion characteristics was performed. We used six methods for an objective comparison of the encoding performance. These are's (Jing and Chau, 2004; Choi *et al.*, 2006; Salgado and Nieto, 2006; Grecos and Yang, 2007; Byung-Gyu, 2008; Kim and Kim, 2008) methods which are usually used for comparison for new proposed fast mode decision algorithms and give good performances.

All the test video sequences used in this experiment are very common in video compression test various MPEG standard sequences were used with CIF and QCIF sizes. Analyses were performed with encoding frames = V 100, RD optimization enabled, QP = 24, 28 and 32, sequence types of IPPP in the Main profile, using CAVLC, with a search range of $MV = \pm 16$ and the number of reference frame = 1, FME is used as default and Hadamard transform is turn on.

JM 11.0 reference software of the JVT (joint video team) was used as a reference code for evaluation of the encoding performance. We defined three measures for evaluating the encoding performance including average Δ PSNR, average Δ Bits and an encoding-time saving factor Δ T. The average Δ PSNR is the difference in decibels between the average PSNR of the proposed method and the corresponding value of another method. As performance improves, this criterion becomes smaller.

$$\Delta\text{PSNR} = \text{PSNR}_{\text{Proposed}} - \text{PSNR}_{\text{Original}} \quad (28)$$

The average Δ Bits is the bit-rate difference as a percentage between the compared methods and the encoding-time saving factor is defined for a complexity comparison as:

$$\Delta\text{Bitrate} = \frac{\Delta\text{Bitrate}_{\text{Proposed}} - \Delta\text{Bitrate}_{\text{Original}}}{\Delta\text{Bitrate}_{\text{Original}}} \times 100\% \quad (29)$$

$$\Delta\text{Time} = \frac{\Delta\text{Time}_{\text{Proposed}} - \Delta\text{Time}_{\text{Original}}}{\Delta\text{Time}_{\text{Original}}} \times 100\% \quad (30)$$

Figure 15 shown the RD curves for several sequences in several algorithm. All six method or algorithm have a RDO performance similar to the JM Reference software with Full inter-mode search method. Among the six kind of method Kim's algorithm has the most similar performance with Full inter-mode search with a large speed up factor and negligible bitrate increment. Jing's method also has similar performance compare with Full inter-mode search but it also has low speed up factor for encoding time. Jeon's, Grecos's and Macroblock tracking also gives similar performance with Full inter-mode search which are less degradation of image quality and also negligible bitrate increment among three of those method macroblock tracking gives the largest speed-up factor for encoding time which is almost 82% in average. The last method is Salgado's, this method gives large bit increment with similar image quality, although it is not fairly enough in bitrate, this method result speed up factor in encoding time for about 70% saving encoding time.

Table 8 shows the results of 6 algorithms for the IPPP sequence type. This six proposed algorithms achieve a speed-up factor with a minimal loss of image quality and a minimal bit increment.

In case of image quality, most of this method yield loss image quality < 0.6 dB in average, just Kim's method has loss of image quality 0.039 dB in average, this is because the information for encode current macroblocks are just from co-located macroblock, this result is very good due to this method gives large saving speed-up factor. For the other method MB tracking has loss of image quality very similar with Full inter-mode search without degradation performance in speedup factor and bitrate, this method has loss PSNR 0.056 dB in average. Jeon's and Grecos's algorithm also have good performance due to image quality in average they can achieve minimum loss of image quality < 0.03 dB

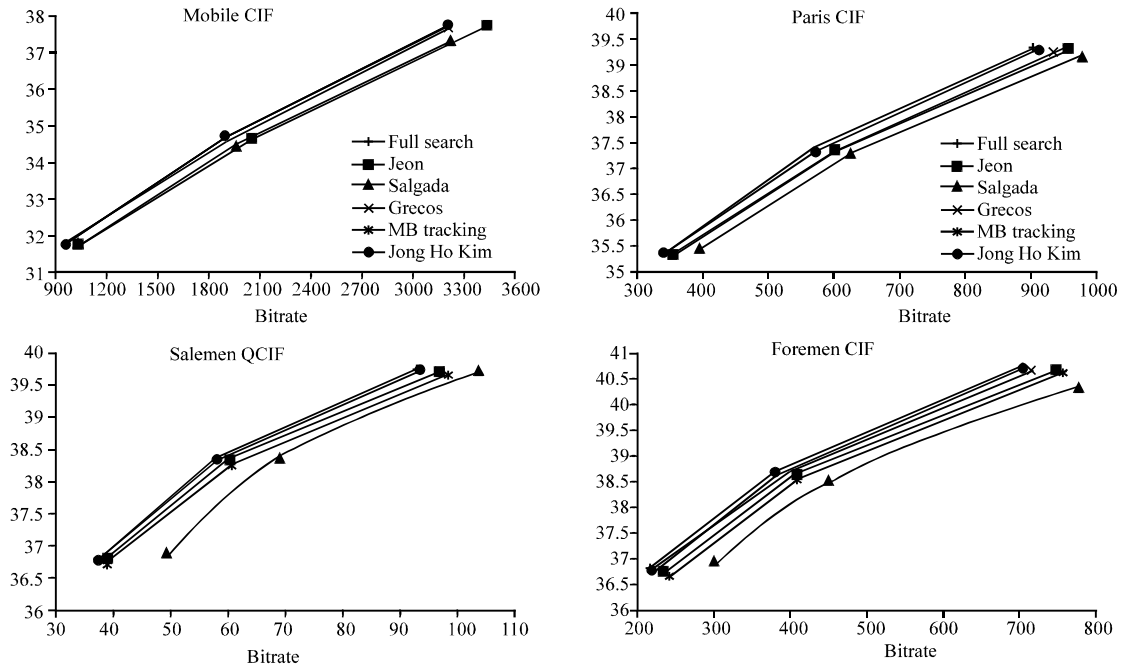


Fig. 15: The RD performance for IPPP sequences of the tested algorithms

although the speed-up factor just 40.9 and 52.12%. Jing’s method although it has low speed-up factor than other method but it gives good performance in prevent loss of image quality which is just 0.051 dB. and Salgado’s method achieve loss of image quality about 0.107 but it provide good speed up factor.

According to bitrate increment in all several proposed method that we review the performance, only Grecos’s method that gives saving bit rate, the other method all of the give bitrate increment in their computation. Among the others Kim’s method yield the smallest bit increment in his computation, it just 0.037% following with MB tracking that yield bit increment about 0.37% in average. Jeon’s and Jing’s method also give good performance due to bit increment, both of them have similar bit increment in average of several sequence, Jeon’s has 0.45% and Jing’s has 0.56%. For all six method, Salgado’s gives the worst bit increment, it yield 13.1% bit increment eventhough it can achieve good speed-up factor. At video sequence which have less motion the bitrate are more higher than video sequence that have large motion. Grecos’s method gives different result, if the other method have bit increment in their encoding process in Greco’s method gives bitrate saving 4.743% in average. This bitrate saving occur in all type of video sequence and video sequence which has less motion tent to gives larger bitrate than large motion.

The last performance evaluation in fast mode decision is speed-up factor or time saving for encoding process. Good performance in speed up factor should be not affect

to other quality qualification, it means the method still have minimum less image quality and negligible bitrate increment. Jing’s method with it fast approach algorithm can maintain the lost image quality and has minimum bitrate increment but the time saving just 12.3% in average. Grecos’s method can achieve 52.12% but in Grecos’s method it’s yield bitrate increment 4.74%, this not quite enough for good performance in compression.

Jeon’s method gives speed-up factor for 40.95% with minimum loss image quality and negligible bitrate increment, it’s because encoding process of this method is Early decision to avoiding Intra process without this process the time saving may less than 40.95% because intra process is 5 time than Inter process. Salgado’s method gives speed-up factor of up to 70.79% in average, this result is fairly enough in encoding performance but Salgado’s method yield very high bitrate increment although it has less image quality. The bitrate increment 13.16% in average is not good due to good performance in encoding process Kim’s method gives very good performance regarding it can maintain very minimum bitrate increment and still has very good image quality, Kim’s method achieve speed factor 74.1%.

Among six methods that we have evaluate the performance, MB tracking method shows the best performance in speed-up factor and also have minimum loss image quality and negligible bitrate increment. MB tracking scheme achieves 82.72% of time saving in average and it is shown in Table 8. MB tracking method also gives result similar time saving for several types of

Table 8: Performance comparison of the tested algorithms on the JM.11.0

Algorithm	Jing			Jeon			Salgado			
Test sequence	QP	Δ PSNR (dB)	Δ Bitrate (%)	Δ Time (%)	Δ PSNR (dB)	Δ Bitrate (%)	Δ Time (%)	Δ PSNR (dB)	Δ Bitrate (%)	Δ Time (%)
Mobile CIF	24	-1.026	0.051	-9.529	-0.002	0.006	-27.187	-0.426	-5.999	-72.529
	28	-0.740	0.563	-11.712	-0.008	-0.094	-18.049	-0.140	-4.277	-64.712
	32	-0.538	1.200	-12.410	-0.011	0.165	-20.988	0.062	2.051	-63.410
Paris CIF	24	-0.745	0.125	-9.691	-0.004	0.088	-45.416	-0.145	2.046	-75.691
	28	-0.632	0.579	-11.356	-0.012	0.175	-43.179	-0.032	4.039	-66.356
	32	-0.519	0.613	-13.683	-0.045	0.022	-43.686	0.081	10.772	-61.683
Foreman CIF	24	-0.915	0.339	-9.449	-0.016	0.024	-34.899	-0.315	3.659	-73.449
	28	-0.632	0.606	-10.623	-0.020	0.279	-33.567	-0.032	12.741	-71.623
	32	-0.480	0.651	-15.018	-0.017	0.015	-36.212	0.121	27.000	-70.958
Mother-daughter CIF	24	-0.698	0.050	-10.264	-0.021	0.377	-46.875	-0.098	8.994	-81.264
	28	-0.494	0.354	-14.628	0.002	0.282	-46.591	0.106	32.939	-74.628
	32	-0.483	0.442	-16.658	-0.027	0.353	-47.594	0.117	58.483	-66.658
Hall CIF	24	-1.062	0.147	-7.044	-0.035	0.538	-36.351	-0.462	-19.107	-82.044
	28	-0.568	0.241	-12.205	-0.038	1.383	-44.390	0.032	13.160	-78.205
	32	-0.531	0.360	-13.981	-0.019	2.703	-49.438	0.069	43.353	-68.981
Akiyo CIF	24	-0.070	0.055	-10.397	-0.008	0.285	-51.366	0.100	15.728	-80.397
	28	-0.177	0.350	-11.448	-0.013	0.867	-52.942	0.123	42.688	-62.448
	32	-0.192	0.640	-10.874	-0.014	1.750	-53.042	-1.192	28.542	-56.874
Stefan QCIF	24	-0.064	0.893	-11.082	-0.013	0.047	-32.676	-0.664	-7.577	-76.082
	28	-0.292	0.951	-12.139	-0.014	0.381	-26.220	-0.092	-2.953	-67.139
	32	-0.322	1.102	-12.480	-0.016	-0.123	-27.250	0.078	2.853	-66.080
Salesman QCIF	24	-0.582	0.453	-12.363	-0.017	0.359	-55.931	0.018	6.506	-80.363
	28	-0.5310	0.480	-18.210	0.004	0.353	-54.652	0.069	14.622	-73.210
	32	-0.534	2.236	-19.293	-0.015	0.481	-53.396	0.066	25.444	-64.293
Average		-0.529	0.562	-12.356	-0.016	0.453	-40.954	-0.107	13.163	-70.795

Algorithm	Grecos			MB tracking			Kim			
Test sequence	QP	Δ PSNR (dB)	Δ Bitrate (%)	Δ Time (%)	Δ PSNR (dB)	Δ Bitrate (%)	Δ Time (%)	Δ PSNR (dB)	Δ Bitrate (%)	Δ Time (%)
Mobile CIF	24	-0.093	-6.351	-46.009	-0.046	0.025	-79.686	-0.0341	0.316	-68.161
	28	-0.063	-6.963	-43.699	-0.049	0.053	-79.288	-0.045	0.793	-71.139
	32	-0.017	-6.300	-46.068	-0.057	-0.242	-79.113	-0.070	0.972	-73.883
Paris CIF	24	-0.084	-2.225	-44.789	-0.045	0.307	-83.212	-0.051	0.093	-64.935
	28	-0.106	-2.079	-46.119	-0.059	0.453	-83.421	-0.050	-0.112	-65.692
	32	-0.132	-1.513	-42.508	-0.082	0.476	-83.413	-0.060	-0.668	-67.588
Foreman CIF	24	-0.040	-4.639	-53.225	-0.050	1.439	-78.161	-0.036	0.747	-74.544
	28	-0.016	-4.906	54.853	-0.056	1.524	-79.291	-0.045	0.752	-75.644
	32	-0.024	-5.451	-56.408	-0.075	2.151	-80.092	-0.039	0.719	-76.231
Mother-daughter CIF	24	-0.028	-6.300	-55.312	-0.042	1.483	-83.113	-0.019	-0.119	-74.738
	28	0.010	-6.454	-56.249	-0.031	1.567	-83.733	-0.034	0.653	-77.252
	32	0.000	-7.742	-57.344	-0.050	1.959	-84.126	-0.027	0.315	-78.951
Hall CIF	24	0.023	-5.747	-52.725	-0.098	-1.828	-81.953	-0.048	-1.067	-74.313
	28	0.008	-4.341	-54.186	-0.081	-1.746	-84.497	-0.057	-1.062	-77.867
	32	-0.039	-6.560	-55.801	-0.060	-0.132	-85.837	-0.016	-2.179	-79.020
Akiyo CIF	24	-0.007	-4.755	-55.363	-0.033	-0.327	-86.322	-0.017	0.059	-77.450
	28	0.008	-5.450	-56.949	-0.029	-0.057	-86.763	-0.014	0.355	-78.913
	32	-0.005	-8.140	-56.688	-0.055	-0.374	-86.642	-0.060	-0.850	-79.593
Stefan QCIF	24	-0.057	-4.493	-48.008	-0.058	-0.008	-80.326	-0.041	0.081	-68.580
	28	-0.045	-4.251	-49.732	-0.061	-0.258	-79.953	-0.057	-0.029	-68.917
	32	-0.034	-3.020	-51.664	-0.052	0.03	-79.767	-0.054	-0.604	-70.156
Salesman QCIF	24	-0.039	-1.047	-51.200	-0.047	1.299	-85.693	-0.019	0.869	-78.436
	28	-0.078	-1.580	-52.012	-0.067	0.708	-85.542	-0.028	1.291	-78.126
	32	-0.080	-3.336	-54.174	-0.051	0.517	-85.399	-0.013	-0.434	-78.505
Average		0.039	-4.743	-52.128	-0.056	0.376	-82.723	-0.039	0.037	-74.110

video sequences. This means that this can give a stable performance for different characteristic of video sequence in H.264/AVC video encoding system.

CONCLUSION

Since fast mode decision scheme is very important in encoding process in H.264 AVC, several algorithms had been proposed to improve performance of fast mode decision with several different methods in inter frame

coding. In this study, we have surveyed the recent algorithms for fast inter mode decision process and categorized them as the used feature. To measure performance of fast mode decision as usual look for loss degradation of image quality, less bitrate increment and speed-up factor.

We have evaluated six methods for an objective comparison of the encoding performance. These are (Jing and Chau, 2004; Choi *et al.*, 2006; Salgado and Nieto, 2006; Grecos and Yang, 2007; Byung-Gyu, 2008).

Kim and Kim, 2008 methods which are famous and usually used for reference comparison for new proposed fast mode decision algorithms.

Jing's method had less speed-up factor just 12.35%, but still had minimum loss image quality 0.529 dB and less bitrate increment 0.562%. Jeon's Algorithm with his selective intra process yielded 40.95% of speed up factor and also less image quality loss 0.016 dB and less bitrate increment 0.453%. Salgado's (Salgado and Nieto, 2006) and Grecos's algorithms (Grecos and Yang, 2007) achieved 70.79 and 52.12% speed-up factor but Salgado's algorithm had large bitrate increment for about 13.16%, different with Grecos's which has bitrate saving for 4.74%. This two algorithm also had less image quality.

Kim and Kim (2008) method achieved very good speed-up factor for 74.11% and just have 0.039 dB loss image quality and 0.037% bitrate increment. The last is MB tracking algorithm (Byung-Gyu, 2008), this method gave very good performance in all term measurement quality, it achieved 82.72% saving with 0.056 dB loss image quality and 0.37% bitrate increment.

REFERENCES

- Byung-Gyu, K., 2008. Novel inter-mode decision algorithm based on macroblock (MB) tracking for the P-slice in H.264/AVC video coding. *IEEE Trans. Circuits Syst. Video Technol.*, 18: 273-279.
- Choi, I., J. Lee and B Jeon, 2006. Fast coding mode selection with rate-distortion optimization for MPEG-4 part-10 AVC/H.264. *IEEE Trans. Circuits Syst. Video Technol.*, 16: 1557-1561.
- Grecos, C. and M. Yang, 2007. A framework for fast mode decision in the H264 video coding standard. *Digital Sig. Process.*, 17: 652-664.
- Huang, H. and D. Hu, 2009. An efficient fast inter mode decision algorithm for H.264/AVC. *Int. Conf. Commun. Mobile Comput.*, 1: 161-165.
- Jing, X. and L.P. Chau, 2004. Fast approach for H.264 inter mode decision. *Elect. Lett.*, 40: 1050-1052.
- Kim, J.H. and B.G. Kim, 2008. Fast block mode decision algorithm in H.264/AVC video coding. *J. Vis. Commun. Image. Representation*, 19: 175-183.
- Liu, P. and K. Jia, 2009. A fast and novel intra and inter modes decision prediction algorithm for H.264/AVC based on the characteristics of macro-block. *Proceedings of the 5th International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Sept. 12-14, Kyoto, Japan, pp: 286-289.
- Liu, Z., L. Shen and Z. Zhang, 2009. An efficient intermode decision algorithm based on motion homogeneity for H.264/AVC. *IEEE Trans. Circuits Syst. Video Technol.*, 19: 128-132.
- Salgado, L. and M. Nieto, 2006. Sequence independent very fast mode decision algorithm on H.264/AVC baseline profile. *Proceedings of the International Conference on Image and Processing*, Oct. 8-11, Atlanta, GA, pp: 41-44.
- Shen, L., Z. Liu, Z. Zhang and X. Shi, 2008. Fast inter mode decision using spatial property of motion field. *IEEE Trans. Multimedia*, 10: 1208-1214.
- Sullivan, G. and T. Wiegand, 1998. Joint video team, rate-distortion optimization for video compression. *IEEE Sig. Process. Mag.*, 11: 74-90.
- Wiegand, T., G.J. Sullivan, G. Bjontegard and A. Luthra, 2003. Overview of the H.264/AVC video coding standard. *IEEE Trans. Circuits Syst. Video Technol.*, 13: 560-576.
- Wu, D., F. Pan, K.P. Lim, S. Wu and Z.G. Li *et al.*, 2005. Fast intermode decision in H.264/AVC video coding. *IEEE Trans. Circuits Syst. Video Technol.*, 15: 953-958.
- Yang, J. and Y. Chen, 2009. A novel fast inter-mode decision algorithm for H.264/AVC based on motion estimation residual. *WASE Int. Conf. Inform. Eng.*, 1: 153-156.
- Yang, S.H. and K.H. Wang, 2009. H.264 fast inter-mode selection based on coded block pattern. *IEICE Trans. Inform. Syst.*, E92.D: 1324-1327.
- Yu, A.C.W., G.R. Martin and H. Park, 2008. Fast inter-mode selection in the H.264/AVC standard using a hierarchical decision process. *IEEE Trans. Circuits Syst. Video Technol.*, 18: 186-195.
- Zeng, H., C. Cai and K.K. Ma, 2009. Fast mode decision for H.264/AVC based on macroblock motion activity. *IEEE Trans. Circuits Syst. Video Technol.*, 19: 491-499.