

## Research Article

# An Improved Fast Mode Decision Method for H.264/AVC Intracoding

**Abderrahmane Elyousfi**

*Computer Science Department, National Engineering School of Applied Sciences, University Ibn Zohr, 80000 Agadir, Morocco*

Correspondence should be addressed to Abderrahmane Elyousfi; [elyousfiabdo@yahoo.fr](mailto:elyousfiabdo@yahoo.fr)

Received 9 January 2014; Accepted 22 April 2014; Published 20 May 2014

Academic Editor: Constantine Kotropoulos

Copyright © 2014 Abderrahmane Elyousfi. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An improved fast and efficient mode decision method for H.264/AVC intracoding is proposed, which is based on the analysis of the gravity center method and more efficient mode selection. In contrast to the fast mode decision method where the intramodes are determined by the gravity center of the block, the mass center vector is computed for the block and the subblocks formed by the proposed subsampling techniques. This method is able to determine all correlation directions of the block that correspond to the intraprediction mode directions of the H.264/AVC. On this basis, only a small number of intraprediction modes are chosen as the best modes for rate-distortion optimization (RDO) calculation. Different video sequences are used to test the performance of the proposed method. Experimental results reveal the significant computational savings achieved with slight peak signal-to-noise ratio (PSNR) degradation and bit-rate increase.

## 1. Introduction

The H.264/AVC video coding standard supports intraprediction for various block sizes. For the luma samples, H.264/AVC supports three block-size types in high profile: luma  $4 \times 4$ , luma  $8 \times 8$ , and luma  $16 \times 16$ . Intraluma  $4 \times 4$  supports eight directional modes and DC mode. Figure 1 illustrates the  $4 \times 4$  luma block and intraprediction directions. In Figure 1(b), eight out of the nine different intraprediction modes of luma  $4 \times 4$  are shown (except DC mode). In each of the modes, the prediction pixels of the current block a-p can be obtained by the neighboring pixels A-M with a certain extrapolating operation. And the pixels A-M in Figure 1(a) belong to the reconstructed image (not the original lossless image). Except the DC mode (mean value prediction), the eight modes can be specified by their local texture direction [1]. Intraluma  $8 \times 8$  shares the same prediction modes with intraluma  $4 \times 4$  except that the coding block size is  $8 \times 8$ . Intraluma  $16 \times 16$  contains three modes with directions: vertical, horizontal, and plane. For the chroma intraprediction, only  $8 \times 8$  block is supported and the prediction modes are the same as those of intraluma

$16 \times 16$  [1–6]. To select the optimal encoding mode for an MB, H.264/AVC video encoder calculates the rate distortion cost (denoted as RDcost) of every possible mode and chooses the mode having the minimum value, and this process is repeatedly carried out for all the possible modes for a given MB [7, 8]. Unfortunately, the computational burden of this type of exhaustively full searching algorithm is far more demanding than any other existing video coding standards.

Several fast intraprediction mode selection algorithms have been proposed [9–22]. In [15], hybrid method is used to achieve better performance than conventional edge detection methods by adapting these filters. Plus, several other improvements are also investigated. First, the sort operations at the stage of individual pixel processing are time consuming and summing method is used instead. Second, as the adaptation of the computation efficient summing method is used, more pixels can be applied to improve the filtering results with negligible time increase. Finally, local information is exploited to enhance the mode selection accuracy. Elyousfi [16] proposed a fast intramode selection algorithm for H.264/AVC that uses the gravity center vector of the block to determine the best intraprediction mode.

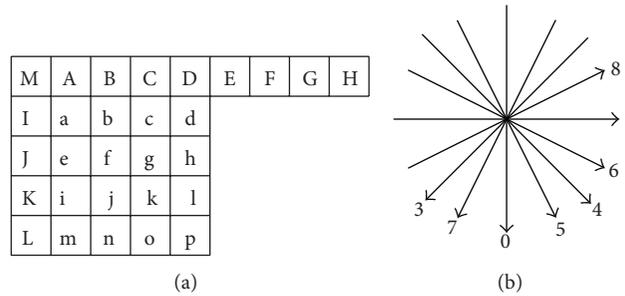


FIGURE 1: The  $4 \times 4$  luma block and prediction directions. (a) For a  $4 \times 4$  luma block, a to p are the pixels to be predicted, and A to M are the neighboring pixels that are available of prediction. (b) Eight prediction directions for intraprediction of  $4 \times 4$  luma block.

In this paper, we present an improved gravity center method proposed by Elyousfi [16]. In [16], the authors use the idea that the direction of the gravity center vector of the block is perpendicular with the direction of the correlation of this block. However, the prediction precision of the above algorithm is hampered by the limitation that this algorithm is not applicable for all correlation directions of the block that correspond to intraprediction mode directions of H.264/AVC. In case of the blocks having a directional correlation such as horizontal-up, horizontal-down, vertical-left, and vertical-right direction, the gravity center vector direction of these blocks is not identified. From this, the gravity center method cannot determine all correlation directions of the block that correspond to intraprediction mode direction of H.264/AVC. Hence, as the best intraprediction mode direction is this that corresponds to the block correlation direction, the previous algorithm [16] cannot determine the best intraprediction mode candidates for RDO computation in intraprediction. Consequently, the previous algorithm [16] may increase the bit rate and/or may affect heavily the PSNR degradation.

This work proposes a novel adaptive fast and efficient intraprediction in H.264/AVC algorithm based on center of mass and two subsampling techniques. This method is able to determine all correlation directions of the block that correspond to the intraprediction mode directions of the H.264/AVC. The technique of the mass center used in the previous work is also applied in this paper. That is, the block mass center direction is perpendicular to the block correlation direction. However, in this paper we indicate the block correlation directions that are perpendicular to the mass center direction of the block. We note that these correlation directions are horizontal, vertical, diagonal-right, and diagonal-left. Furthermore, after the analysis of the characteristics of the blocks having these correlation directions, we observed that these blocks are symmetric.

For the case of the blocks that their correlation directions are not perpendicular with the direction of the mass center vector of these blocks, the previous work [16] is not capable to determine these correlation directions. We note that these directions are horizontal-up, horizontal-down, vertical-left, and vertical-right. In this paper, in order to determine the correlation direction of these blocks with mass center direction

we formed their corresponding symmetric subblock. These subblocks are formed by subsampling the pixels of the block.

Two subsampling techniques are proposed in this paper. The first method named by ILHC (impar lines and half columns) and applied to vertical-left and vertical-right directional correlation blocks. The square subblock formed by this method (ILHC) is composed by the impar lines and the half columns in the middle of the block. The second method named by ICHL (impar columns and half lines) and applied to horizontal-up and horizontal-down correlation direction of the blocks. The square subblock formed by this method (ICHL) is composed by the impar columns and half lines in the middle of the block.

The mass center direction computed from the subblock formed by the ILHC method can determine if the block has vertical-left or vertical-right correlation direction. Also, the mass center direction computed from the subblock formed by the ICHL method can determine if the block has horizontal-up or horizontal-down correlation direction.

Based on the block correlation direction determined by the mass center method and subsampling techniques, the best intraprediction candidates are chosen for RDO calculation during intraprediction. The experiment results reveal that the proposed correlation direction detection algorithm can provide better coding performance and time reduction comparing to the previous algorithms.

This paper is organized as follows. The proposed fast and efficient intraprediction algorithm will be illustrated in Section 2. In Section 3, we present the experimental results and discuss the performance of the proposed algorithm, and, then, we conclude this paper in Section 4.

## 2. The Proposed Algorithm

In this section, the observations which motivate the basic idea of the proposed subsampling techniques will be discussed first. Then the proposed method will be introduced in detail, including the two proposed subsampling techniques and block correlation direction determination. Finally, to further reduce the computational complexity for intracoding in H.264/AVC, a fast intramode decision algorithm is introduced, while maintaining the original coding performance very well.

### 2.1. Observations and the Two Proposed Subsampling Techniques

**2.1.1. Observations.** We observed that when the block has a correlation direction, such as horizontal, vertical, diagonal-right, and diagonal-left, this block has a symmetry axis through the center of this block. The direction of this axis is perpendicular to the direction of the homogeneous pixels of the block. Figure 2 shows these four correlation directions and their corresponding symmetry axes for  $5 \times 5$  blocks. Also, we observed that when the block has a correlation direction, such as horizontal-up, horizontal-down, vertical-left, and vertical-right, this block does not have a symmetry axis through the center of this block. We note that this is valid for all block sizes.

**2.1.2. The Proposed ILHC Subsampling Method.** The proposed ILHC subsampling method forms square subblocks by subsampling the pixels of the block with two main steps. In the first, it selects the impair lines of the block. In the second, it selects the half columns in the middle of the block.

Figure 3(a) shows the two  $5 \times 5$  blocks and their corresponding square subblocks formed by ILHC subsampling method. We observed, from this figure, that the subblock formed by the proposed ILHC subsampling method for the block having, respectively, the vertical-left and vertical-right directional homogeneous pixels has a diagonal-left and diagonal-right directional homogeneous pixels. Hence, each subblock of these subblocks has a symmetry axis through the center of this subblock.

**2.1.3. The Proposed ICHL Subsampling Method.** The proposed ICHL subsampling method forms square subblocks by subsampling the pixels of the block with two main steps. In the first, it selects the impair columns of the block. In the second, it selects the half lines in the middle of the block.

Figure 3(b) shows the two  $5 \times 5$  blocks and their corresponding square subblocks formed by ICHL subsampling method. From this figure we observed that the subblock formed by the proposed ICHL subsampling method for the block having, respectively, the horizontal-up and horizontal-down directional homogeneous pixels has diagonal-left and diagonal-right directional homogeneous pixels. Hence, each subblock of these subblocks has a symmetry axis through the center of this sub-block.

We conclude that the eight correlation directions of the block can be determined if we determine the vector direction of the symmetry axis of the block (or of the subblock formed by the subsampling techniques). In this paper we prove that this direction can be determined by the mass center method. In the next section, we detail the relationship between the symmetry axes direction and mass center direction of the block.

**2.2. Block Symmetry Axes Direction Determination.** The principle of the mass center scheme is applied to the determination of the block symmetry axes. In this section, we prove

that the symmetry axes direction of the block is parallel to the mass center direction of the block.

**2.2.1. Mass Center Theory.** In this study, gray levels are regarded as the pixel mass. For a block, in a luma (or chroma) picture, we define the corresponding mass center vector,  $\vec{G} = \{G_x, G_y\}$ , as

$$\begin{aligned} G_x &= \frac{1}{S_I} \sum_{x=x_0}^{x_0+N} \sum_{y=y_0}^{y_0+M} x \times I_{x,y} \\ G_y &= \frac{1}{S_I} \sum_{x=x_0}^{x_0+N} \sum_{y=y_0}^{y_0+M} y \times I_{x,y}, \end{aligned} \quad (1)$$

where  $I_{x,y}$  is the intensity of the pixel of location  $(x, y)$  of a block,  $(G_x, G_y)$  is the coordinate of the mass center of a block,  $(x_0, y_0)$  is the coordinate of the pixel up-left of a block, and  $(M+1, N+1)$  is the block dimension.  $S_I$  represents the sum of block pixels intensity values; the formula of the equation used for computation of this sum is defined as follows:

$$S_I = \sum_{x=x_0}^{x_0+N} \sum_{y=y_0}^{y_0+M} I_{x,y}. \quad (2)$$

The direction of the block mass center vector ( $\vec{G}$ ) is computed by

$$\text{Ang}(\vec{G}) = \frac{180}{\pi} \arctan\left(\frac{G_y}{G_x}\right). \quad (3)$$

**2.2.2. Block Symmetry Axes Direction Determination with Mass Center Direction.** A symmetry axes direction of the block can be obtained by using the mass center method. The following simple proof verifies that the mass center vector direction represents the symmetry axis direction of the block and is taken as the basis of the proposed algorithm.

Suppose that in an orthonormal coordinate system  $(O, \vec{i}, \vec{j})$ , the origin is chosen to be the block's center,  $i$  represents the horizontal direction to the right and  $j$  the vertical direction to the bottom, and each block  $N \times N$ ,  $N$  is an impair value (i.e.,  $N = 2m + 1$  which  $m = \{2, 4, 8\}$ ), and the possible correlation directions of this block are horizontal, vertical, diagonal-right, and diagonal-left direction; then the mass center vector direction is parallel to the symmetry axis direction of this block.

*Proof.* Vertical direction correlation block: we let  $L_v$  be the line through the origin  $O$  and parallel to the vector  $V_{L_v} = (x, 0)$ , where  $x$  is an integer number except zero. This line  $L_v$  is the symmetry axis of the vertical direction correlation block (see observation 1). Hence, all pixels in this block that are symmetrical with this line  $L_v$  have the same intensity and can be written as

$$I_{x,y} = I_{x,-y}, \quad -m \leq x, y \leq m. \quad (4)$$

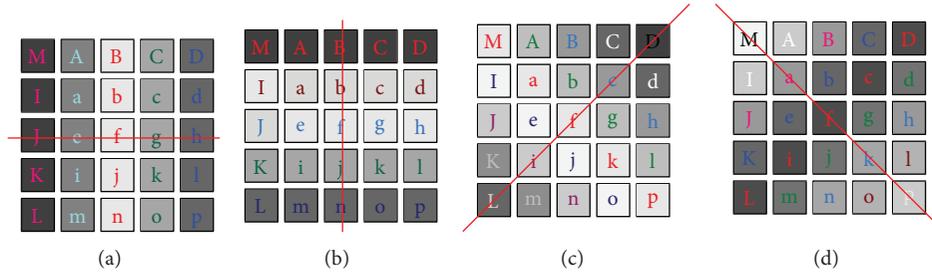


FIGURE 2: (a) The  $5 \times 5$  block horizontal homogeneous pixels and their corresponding symmetry axes. (b) The  $5 \times 5$  block vertical homogeneous pixels and their corresponding symmetry axes. (c) The  $5 \times 5$  block diagonal-right homogeneous pixels and their corresponding symmetry axes. (d) The  $5 \times 5$  block diagonal-left homogeneous pixels and their corresponding symmetry axes.

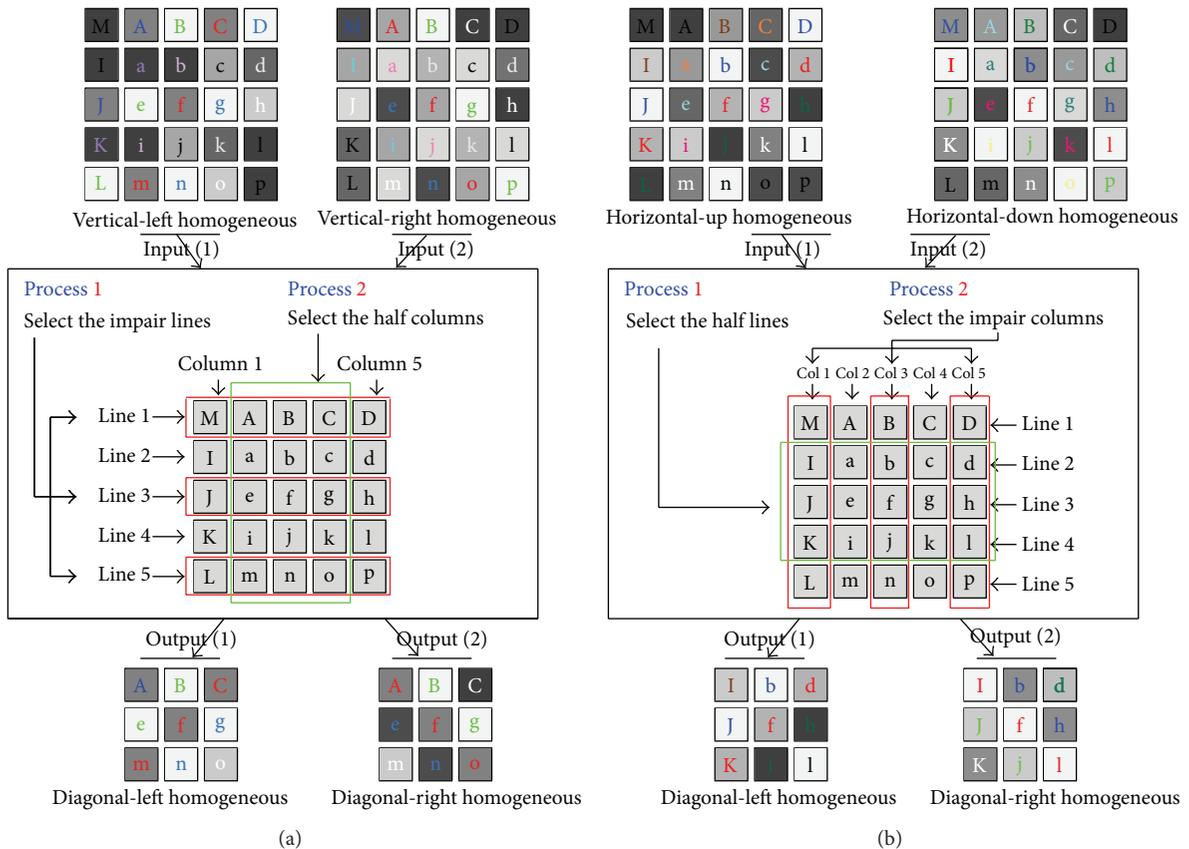


FIGURE 3: (a) The two  $5 \times 5$  blocks and their corresponding square subblocks formed by ILHC method. (b) The two  $5 \times 5$  blocks and their corresponding square subblocks formed by ICHL method.

The coordinates of the mass center of this block can be written as

$$\begin{aligned}
 G_x &= \frac{1}{S_I} \sum_{x=-m}^m \sum_{y=-m}^m I_{x,y} \times x \\
 &= \frac{1}{S_I} \sum_{x=-m}^m N \times I_{x,0} \times x, \\
 G_y &= \frac{1}{S_I} \sum_{y=-m}^m \sum_{x=-m}^m I_{x,y} \times y
 \end{aligned}$$

$$\begin{aligned}
 &= \frac{1}{S_I} \sum_{y=1}^m \left[ \sum_{x=-m}^m I_{x,y} \times y + \sum_{x=-m}^m I_{x,-y} \times (-y) \right] \\
 &= 0.
 \end{aligned}$$

(5)

Hence, the mass center vector of this block is  $\mathbf{G} = (\kappa, 0)$ , where  $\kappa$  is a constant value. From these computations, we found that the direction of the mass center vector is parallel to the symmetry axis direction of this block.

Horizontal direction correlation block: we let  $L_h$  be the line through the origin  $O$  and parallel to the vector

$V_h = (0, y)$ , where  $y$  is an integer number except zero. This line  $L_h$  is the symmetry line of the horizontal direction correlation block (see observation 1). The pixels of this block that are symmetrical with this line  $L_h$  have the same intensity. This later expression can be written by this equality

$$I_{x,y} = I_{-x,y}, \quad -m \leq x, y \leq m. \quad (6)$$

The coordinates of the mass center of this block can be written as

$$\begin{aligned} G_x &= \frac{1}{S_I} \sum_{x=1}^m \left[ \sum_{y=-m}^m I_{x,y} \times x + \sum_{y=-m}^m I_{-x,y} \times (-x) \right] = 0 \\ G_y &= \frac{1}{S_I} \sum_{y=-m}^m \sum_{x=-m}^m I_{x,y} \times y. \\ &= \frac{1}{S_I} \sum_{y=-m}^m N \times I_{0,y} \times y. \end{aligned} \quad (7)$$

From these computations, the mass center vector of horizontal direction correlation block is written by  $\mathbf{G} = (0, \kappa)$ , where  $\kappa$  is a constant value. Then the direction of this vector  $\mathbf{G}$  is parallel to the symmetry axis of this block.  $\square$

**2.3. Mass Center Direction and Intraprediction Mode Candidates in H.264/AVC.** In H.264/AVC, intraprediction uses different block sizes, and each block has a limited number of the intraprediction direction. Hence, we use the angle of the mass center vector of the block to determine the intraprediction mode of this block.

**2.3.1.  $4 \times 4$  Luma Block Directional Correlation.** The  $4 \times 4$  luma blocks are more suitable to predict the pictures with significant details. There are nine prediction modes, the DC prediction mode and eight directional prediction modes. We classified these directional prediction modes into three classes: the first class, named by class 1, contains the mode 0: the vertical prediction mode, mode 1: the horizontal prediction mode, mode 3: the diagonal-left mode, and mode 4: the diagonal-right mode. The second class, named by class 2, contains the mode 5: the vertical-right mode and mode 7: the vertical-left mode. The third class, named by class 3, contain mode 6: the horizontal-down mode and mode 8: the horizontal-up mode. In the following, the corresponding

mass center direction of each mode of each classes modes prediction of the  $4 \times 4$  blocks is determined:

$$I_{4 \times 4} = \begin{cases} \text{Mode 3} & \beta_1 \in \left[ \frac{3\pi}{16} + k\pi, \frac{5\pi}{16} + k\pi \right] \\ \text{Mode 0} & \beta_1 \in \left[ \frac{7\pi}{16} + k\pi, \frac{9\pi}{16} + k\pi \right] \\ \text{Mode 4} & \beta_1 \in \left[ \frac{11\pi}{16} + k\pi, \frac{13\pi}{16} + k\pi \right] \\ \text{Mode 1} & \beta_1 \in \left[ -\frac{\pi}{16} + k\pi, \frac{\pi}{16} + k\pi \right] \\ \text{Mode 5} & \beta_2 \in \left[ \frac{11\pi}{16} + k\pi, \frac{13\pi}{16} + k\pi \right] \\ \text{Mode 7} & \beta_2 \in \left[ \frac{3\pi}{16} + k\pi, \frac{5\pi}{16} + k\pi \right] \\ \text{Mode 6} & \beta_3 \in \left[ \frac{11\pi}{16} + k\pi, \frac{13\pi}{16} + k\pi \right] \\ \text{Mode 8} & \beta_3 \in \left[ \frac{3\pi}{16} + k\pi, \frac{5\pi}{16} + k\pi \right] \end{cases} \quad k = \{0, 1\}. \quad (8)$$

The angle  $\beta_1$  is computed as  $\beta_1 = \theta_1 - \pi/2$ , where  $\theta_1$  is the angle of the  $4 \times 4$  block mass center computed by (3). The angle  $\beta_2$  is computed as  $\beta_2 = \theta_2 - \pi/2$ , where  $\theta_2$  is the mass center angle, computed by (3), for the subblocks formed by ICHL method. The angle  $\beta_3$  is computed as  $\beta_3 = \theta_3 - \pi/2$ , where  $\theta_3$  is the mass center angle, computed by (3), for the subblocks formed by ILHC method.

**2.3.2.  $16 \times 16$  Luma Block Directional Correlation.** In the case of  $16 \times 16$  luma blocks, there are only horizontal and vertical prediction modes, plus a plane prediction and a DC prediction mode. So, to determine the prediction mode candidate, we associate the directional correlation for horizontal and vertical prediction modes to their corresponding areas of the block's mass center direction and the rest of this areas is associated with the plane mode.

Therefore, for each  $16 \times 16$  luma block, the mass center direction of this block and their corresponding prediction modes are represented as follows. For each  $16 \times 16$  luma block, let  $\theta_4$  be the mass center angle and let  $\beta_4 = \theta_4 - \pi/2$ ; then

$$I_{16 \times 16} = \begin{cases} \text{Mode 1} & \beta_4 \in \left[ -\frac{\pi}{8}, \frac{\pi}{8} \right] \cup \left[ -\frac{\pi}{8} + \pi, \frac{\pi}{8} + \pi \right] \\ \text{Mode 0} & \beta_4 \in \left[ \frac{3\pi}{8}, \frac{5\pi}{8} \right] \cup \left[ \frac{3\pi}{8} + \pi, \frac{5\pi}{8} + \pi \right]. \end{cases} \quad (9)$$

**2.4. Mode Decision Algorithm for Intraprediction.** Based on the prediction mode determined by the mass center and subsampling techniques, the efficient and fast mode decision algorithms for intraprediction select a small number of the prediction modes as the best candidates to be used in RDO computation. So, we can determine the candidate mode for intracoding block size by the following rules.

*Step 1.* For each chroma block, two mass center directions are computed by using (3), one from component  $U$  and the other from component  $V$ . According to (9), the intraprediction mode for each component is selected.

*Step 2.* If these two intraprediction modes of the two components are identical; we chose this mode as the candidate intraprediction modes for RDO calculation; otherwise, the DC mode is the mode used in the RDO calculation.

*Step 3.* The direction of the mass center vector is computed for the  $16 \times 16$  block by using (3). According to (9), the mode that corresponds to this direction is chosen as a candidate intraprediction mode for RDO computation. In addition to this mode, the DC mode is also chosen as the another intraprediction mode candidate for RDO computation.

*Step 4.* For each of the  $4 \times 4$  luma blocks (total number is sixteen for a MB), the directions of the mass center vector of the  $4 \times 4$  block, of the subblock formed by IHCL technique, and of the subblock formed by ICHL technique are computed by using (3).

*Step 5.* According to (8), the modes that correspond to these directions are chosen as a candidate intraprediction modes for RDO computation. In addition to these modes, the DC mode is also chosen as the other intraprediction mode candidate for RDO computation.

*Step 6.* For each of the  $8 \times 8$  luma blocks (total number is four for a MB), the directions of the mass center vector of the  $8 \times 8$  block, of the subblock formed by IHCL technique, and of the subblock formed by ICHL technique are computed by using (3).

*Step 7.* According to (8), the modes that correspond to these directions are chosen as a candidate intraprediction modes for RDO computation. In addition to these modes, the DC mode is also chosen as the another intraprediction mode candidate for RDO computation.

*2.5. Computational Complexity Assessment.* Table 1 summarizes the number of candidates selected for RDO computation by the fast intraprediction methods. As can be seen from Table 1, the encoder with the proposed algorithm would need to perform only  $1 \times (1 \times 16 + 1 \times 4 + 2) = 22$  if one intraprediction mode is selected by the mass center method for each  $4 \times 4$  luma blocks and one for each  $8 \times 8$  luma block. The upper limit of RDO calculations in our algorithm is  $1 \times (4 \times 16 + 4 \times 4 + 2) = 82$  if three intraprediction modes are selected by the mass center method for each  $4 \times 4$  luma block and three for each  $8 \times 8$  luma blocks.

From these data, our proposed algorithm significantly reduces the number of RDO calculation, compared with the method presented in [15] (between  $1 \times (2 \times 16 + 2 \times 4 + 2) = 42$  and  $2 \times (5 \times 16 + 5 \times 4 + 2) = 204$ ) and the method presented in [16] (between  $1 \times (2 \times 16 + 1 \times 4 + 2) = 38$  and  $2 \times (2 \times 16 + 5 \times 4 + 4) = 112$ ).

### 3. Experimental Results and Discussion

As the most previous methods do not support intra  $8 \times 8$  block, this section presents simulation results based on Chen et al.'s algorithm [15], Elyousfi's algorithm [16],

and the proposed fast and efficient intraprediction algorithm in H.264/AVC.

*3.1. Coding Conditions.* All the algorithms were implemented into H.264/AVC reference software JM18.0 [23]. The system platform is the Intel (R) Core (TM)2 Duo CPU Processor of speed 3.4 GHz, 4.00 Gbytes RAM, and Microsoft Windows Vista. The simulation condition is as RD optimization is enabled, CABAC is enabled, GOP structure is full I, the number of frames of each sequence is 150, and FREXT Profile: high profile. The group of experiments were carried out on the test sequences with the 4 quantization parameters; that is, QP = 28, 32, 36, and 40. The averaged PSNR values of luma (Y) and chroma (U, V) are used and are based on the following equations:

$$\overline{\text{PSNR}} = 10 \times \log_{10} \left( \frac{255^2}{\overline{\text{MSE}}} \right), \quad (10)$$

where  $\overline{\text{MSE}}$  is the average mean square error [16].

The comparisons with the case of exhaustive search were performed with respect to the PSNR difference ( $\Delta\text{PSNR}$ ), the data bits rate difference ( $\Delta\text{Bit}$ ), and the difference of coding time ( $\Delta\text{Time}$ ).

In order to evaluate the time saving of the fast prediction algorithm, the following calculation is defined to find the time differences. Let  $T_{\text{JM}}$  denote the coding time used by full search intraprediction algorithm of JM18.0 encoder and let  $T_{\text{FI}}$  be the time taken by the fast mode decision algorithm; the time difference is defined as

$$\Delta\text{Time} = \frac{T_{\text{FI}} - T_{\text{JM}}}{T_{\text{JM}}} \times 100\%. \quad (11)$$

PSNR and bit-rate differences are calculated according to the numerical averages between the RD-curves derived from JM18.0 encoder and the fast algorithm, respectively. The detailed procedures in calculating these differences can be found from a JVT document authored by Bjontegaard [24].

*3.2. Coding Performances.* In this experiment, a total number of 150 frames are used for each sequence, and the period of I-frames is set to 1; that is, all the frames in the sequence are intracoded. In all the I-frames structure encoding, for each MB in each frame of this sequence, intracoding is chosen as the possible coding modes in RDO operation; thus great time saving is expected by using the fast intracoding algorithms for this structure encoding. Table 2 shows the simulation results of the proposed algorithm and Chen et al.'s [15] and Elyousfi's [16] algorithms for various sequences with all intraframes type. Notice that in the table, positive values mean increments and negative values mean decrements.

The results show that the proposed schemes reduced execution time greater than 87.893% with only an average of 0.046 dB losses in PSNR and 0.616% increments in bitrate only. These results are compared to those obtained with Chen et al.'s [15] and Elyousfi's [16] algorithms where the time coding is reduced in average, respectively, to 63.183% and 79.689% with an average of 0.089 dB and 0.134 dB losses in

TABLE 1: Comparison of the number of candidate modes.

Algorithms	Luma $4 \times 4$	Luma $8 \times 8$	Luma $16 \times 16$	Chroma $8 \times 8$	Min and max number of RDO computation
Reference [15]	2-5	2-5	2	1-2	042-204
Reference [16]	2	1-5	2-4	1-2	038-112
Proposed	1-4	1-4	2	1	022-082

TABLE 2: Simulation results for all intraframes sequences.

Sequences		$\Delta Time$			$\Delta PSNR$			$\Delta Bitrate$		
Format	Name	Reference [15]	Reference [16]	Proposed	Reference [15]	Reference [16]	Proposed	Reference [15]	Reference [16]	Proposed
QCIF	Silent	-64.855	-87.711	-92.890	-0.088	-0.134	-0.044	0.994	1.219	0.327
	Container	-64.309	-80.062	-89.733	-0.106	-0.148	-0.050	1.020	1.369	0.488
	Foreman	-60.426	-74.167	-84.592	-0.119	-0.153	-0.048	1.952	2.682	1.203
	Hall	-68.100	-87.697	-93.747	-0.056	-0.129	-0.033	1.003	1.205	0.340
CIF	Foreman	-63.571	-75.347	-83.283	-0.103	-0.187	-0.069	2.884	3.110	1.084
	Coastguard	-61.330	-75.066	-82.891	-0.147	-0.159	-0.053	2.922	3.790	1.347
	Paris	-59.846	-76.884	-84.694	-0.085	-0.094	-0.031	1.470	2.178	0.728
720 p	News	-66.109	-87.650	-93.455	-0.047	-0.127	-0.024	0.825	1.090	0.390
	Crew	-67.466	-82.388	-90.176	-0.089	-0.107	-0.055	1.077	1.146	0.366
	Parkrun	-54.992	-71.105	-83.800	-0.072	-0.122	-0.058	0.962	1.130	0.288
	Shields	-64.004	-78.503	-87.561	-0.069	-0.113	-0.041	0.883	1.193	0.211
Average		-63.183	-79.689	-87.893	-0.089	-0.134	-0.046	1.454	1.828	0.616

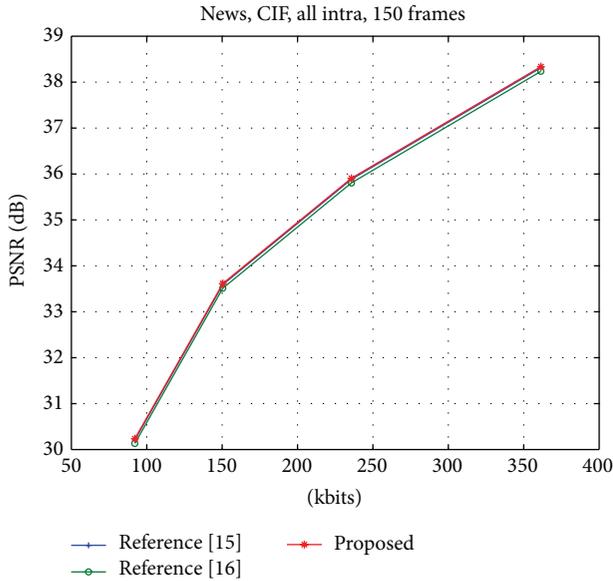


FIGURE 4: Comparison of PSNR for the all intrasequences of news.

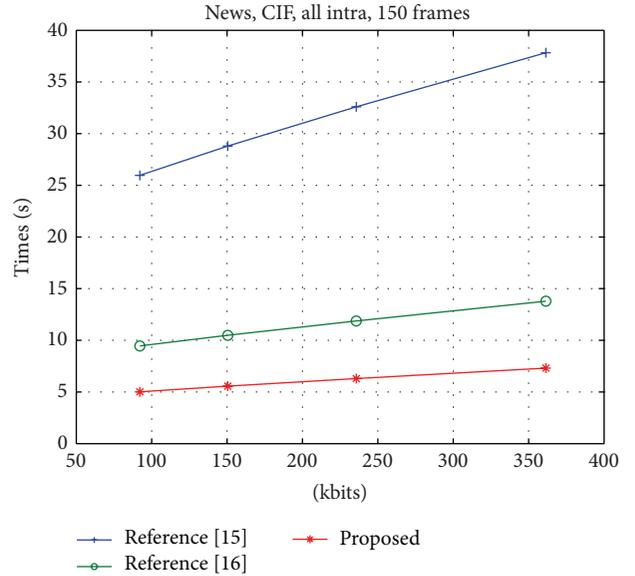


FIGURE 5: The computational time comparison of news all intrasequence.

PSNR and, respectively, 1.454% and 1.828% increments in bitrate.

The proposed schemes achieve faster encoding in intraprediction compared to the Chen et al.'s [15] and Elyousfi's [16], with little RD performance enhancement. Figures 4 and 5 show the RD performance and the computation time for the all I-frames sequence "News," respectively. In Figure 4, three RD curves resulting from the Chen et al.'s algorithm [15], Elyousfi's algorithm [16], and the proposed

schemes are nearly overlapping each other that our proposed algorithm has greater performances as compared to the other approaches in terms of PSNR and data bits but offers higher computation time saving as shown in Figure 5.

#### 4. Conclusion

This paper presented an improved fast intraprediction in H.264/AVC algorithm based on mass center method and two

subsampling techniques. Extensive experimental results show that the proposed method can achieve 87.893% total encoding time reduction with only 0.046 dB PSNR degradation and 0.616% bit rate increase on average. This performance is more efficient than most of the well-known fast intramode decision algorithms for H.264/AVC.

## Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

## References

- [1] G. J. Sullivan, P. Topiwala, and A. Luthra, "The H.264/AVC advanced video coding standard: overview and introduction to the fidelity range extensions," in *Applications of Digital Image Processing XXVII*, vol. 5558 of *Proceedings Of SPIE*, pp. 454–474, Denver, Colo, USA, August 2004.
- [2] ITU-T Recommendation H. 264 and ISO/IEC, 14496-10 (MPEG-4) AVC, "Advanced Video Coding for Generic Audio-visual Services," (version 1: 2003, version 2: 2004) version 3: 2005.
- [3] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.
- [4] A. Puri, X. Chen, and A. Luthra, "Video coding using the H.264/MPEG-4 AVC compression standard," *Signal Processing: Image Communication*, vol. 19, no. 9, pp. 793–849, 2004.
- [5] I. E. G. Richardson, *H. 264 and MPEG4 Video Compression: Video Coding for Next Generation Multimedia*, John Wiley & Sons, 2003.
- [6] ISO/IEC, "Report of The Formal Verification Tests on AVC, (ISO/IEC, 14496-10—ITU-T Rec.H.264)," Tech. Rep. ISO/IEC JTC1/SC29/WG11 MPEG2003/N6231, ISO/IEC, Waikoloa, Hawaii, USA, 2003.
- [7] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for: video compression," *IEEE Signal Processing Magazine*, vol. 15, no. 6, pp. 74–90, 1998.
- [8] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 688–703, 2003.
- [9] H. Li, K. N. Ngan, and Z. Wei, "Fast and efficient method for block edge classification and its application in H.264/AVC video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 6, pp. 756–768, 2008.
- [10] A.-C. Tsai, A. Paul, J.-C. Wang, and J.-F. Wang, "Intensity gradient technique for efficient intra-prediction in H.264/AVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 5, pp. 694–698, 2008.
- [11] A.-C. Tsai, J.-F. Wang, J.-F. Yang, and W.-G. Lin, "Effective subblock-based and pixel-based fast direction detections for H.264 intra prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 7, pp. 975–982, 2008.
- [12] K. Bharanitharan, B.-D. Liu, J.-F. Yang, and W.-C. Tsai, "A low complexity detection of discrete cross differences for fast H.264/AVC intra prediction," *IEEE Transactions on Multimedia*, vol. 10, no. 7, pp. 1250–1260, 2008.
- [13] Y.-H. Huang, T.-S. Ou, and H. H. Chen, "Fast decision of block size, prediction mode, and intra block for H.264 intra prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 8, pp. 1122–1132, 2010.
- [14] R. Su, G. Liu, and T. Zhang, "Fast mode decision algorithm for intra prediction in H.264/AVC with integer transform and adaptive threshold," *Signal, Image and Video Processing*, vol. 1, no. 1, pp. 11–27, 2007.
- [15] C. Chen, J. Chen, T. Xia, Z. Ju, and L. Po, "An improved hybrid fast mode decision method for H. 264/AVC intra coding with local information," *Multimed Tools and Applications*, 2013.
- [16] A. Elyousfi, "Gravity direction-based ultra-fast intraprediction algorithm for H.264/AVC video coding," *Journal Signal, Image and Video Processing*, vol. 7, no. 1, pp. 53–65, 2013.
- [17] D. Quan and Y.-S. Ho, "Categorization for fast intra prediction mode decision in H.264/AVC," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, pp. 1049–1056, 2010.
- [18] K. Lim, S. Kim, J. Lee, D. Pak, and S. Lee, "Fast block size and mode decision algorithm for intra prediction in H.264/AVC," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 2, pp. 654–660, 2012.
- [19] J. W. Chen, C. H. Chang, C. C. Lin, Y. H. Yang, J. I. Guo, and J. S. Wang, "A condition-based intra prediction algorithm for H.264/AVC," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '06)*, pp. 1077–1080, Ontario, Canada, July 2006.
- [20] F. Fu, X. Lin, and L. Xu, "Fast intra prediction algorithm in H.264/AVC," in *Proceedings of the 7th International Conference on Signal Processing Proceedings (ICSP '04)*, pp. 1191–1194, Beijing, China, September 2004.
- [21] B.-G. Kim, "Fast selective intra-mode search algorithm based on adaptive thresholding scheme for H.264/AVC encoding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 1, pp. 127–133, 2008.
- [22] J. Kim and J. Jeong, "Fast intra-mode decision in H.264 video coding using simple directional masks," in *Visual Communications and Image Processing*, vol. 5960 of *Proceedings Of SPIE*, pp. 1071–1079, Beijing China, July 2005.
- [23] H. 264/AVC, "JM Reference Software Version 18.0," <http://iphome.hhi.de/suehring/tml/download/>.
- [24] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," in *Proceedings of the 13-th VCEG Meeting*, Austin, Tex, USA, April 2001, Document VCEG-M33.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

