A Highly Efficient Mode Decision Algorithm and Architecture for AVS HD Video Encoder

Shuai Li, Chuang Zhu, Fei Wang, Hui-zhu Jia, Xiao-dong Xie, Wen Gao
National Engineering Laboratory for Video Technology
Peking university
Beijing, China
{sli, czhu, fwang, hzjia, xdxie, wgao}@jdl.ac.cn

***Abstract*—In Advanced Audio Video coding Standard (AVS), the utilization of variable block size ranging from 16x16 to 8x8 in inter frame encoding improves the coding efficiency significantly compared with a fixed MB partition. Rate distortion optimization (RDO) is the best known mode decision method, but the corresponding extremely high computational complexity limits its application. This paper proposes an algorithm based on the visual perception model and Sobel operator edge detection model to quickly select the best inter mode from 16x16, 16x8, 8x16 and 8x8 just by using the original pixels. We further analyze and redesign the MB level pipeline structure, and give the optimized hardware structure of the encoder. We tested different sequences including cif, 720p and 1080p, and the experimental results show that the coding efficiency is comparable with the traditional RDO method. The proposed hardware structure saves fractional motion estimation (FME) by 60% in areas and reduces the processing time by 200 cycles. Our proposed mode decision architecture can support the real time processing of 1080P@30fps.**

Keywords- inter mode decision, AVS, Sobel operator, visual perception determining model, hardware structure

I. INTRODUCTION

Advanced Audio Video coding Standard (AVS) is established by China AVS Working Group, it has been accepted as an option by ITU-TFGIPTY for IPTV applications.

In AVS, there are four types of partitions in a macroblock (MB): 16x16, 16x8, 8x16 and 8x8 as shown in Fig. 1[1]. To achieve the highest coding efficiency, some previous works like [2-3] used rate distortion optimization (RDO) technique to select the best mode from all the candidate modes of AVS standard. In RDO based mode decision (MD), all the RD costs of different MB modes are calculated via RDO technique, and the mode with minimum RD cost value is chosen as the best mode. The RD cost is computed using (1) for each candidate mode:

$$J = D + \lambda_{mode} \times R \tag{1}$$

D is described as sum of squared differences (SSD) for AVS RDO based mode decision. It represents the distortion between the original picture and the reconstructed picture. λ_{mode} is a weight parameter. R is the coding bits for each mode. The reconstructed pixels are needed to yield SSD. Generally, it needs the motion estimation, discrete cosine transform (DCT), quantization, inverse quantization, inverse DCT and entropy coding to get the reconstructed pixels and the real coding bits, and the whole process is of great computational complexity. To address the high computational complexity problem, a wide range of fast algorithms for inter mode selection have been developed.

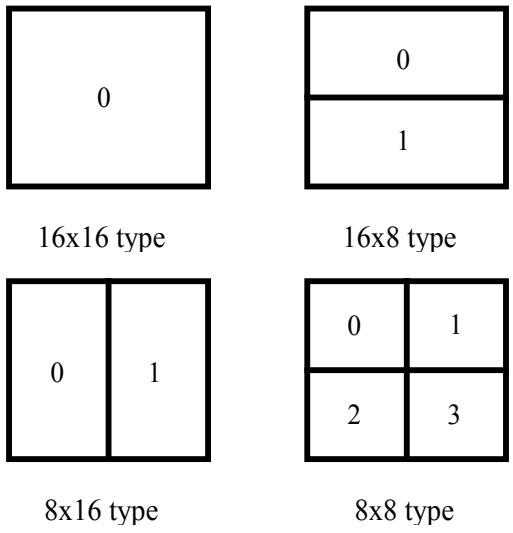

Figure 1. Different partition in a MB

[2] provides a fast mode decision method which select the best mode based on the spatial homogeneity and the temporal stationarity characteristics of video objects. Based on the algorithm, only a small number of inter modes are selected in RDO process. In [3] [4], the authors judge the different modes by using a threshold technique. However, the threshold is not adaptive, and it is just obtained from a lot of statistical experiments. Thus the judgement is not very accurate and need to be adjusted in different sequences. [5] proposes a fast inter mode decision method based on the residual of motion estimation using the sum of absolute difference (SAD) to decide the best mode of current MB. However, this method has significant PSNR degradation [6].

In this paper, we propose a fast inter mode decision algorithm based on the visual perception model [7] [8] and the Sobel operator edge detection model. Specifically, we use visual perception model to compute the visual perception threshold and Sobel operator to extract the image edge information. Based on the visual perception threshold and

This work was supported by CNNSF under contract No.61171139 and No. 61035001, and NBRPC under contract No.2009CB320906.

978-1-4577-1775-8/12/$26.00 ©2012 IEEE

image edge information, we choose the best inter mode (BIM) directly. The rest of the paper is organized as follows. Section II describes the proposed inter mode decision algorithm. In section III, we firstly analyze the MB level pipeline structure, and then give our proposed high efficiency encoder structure. Finally, we analyze its performance and the utilization of hardware resources. The experimental results of our proposed method are given in Section IV. Finally, Section V concludes this paper.

II. THE PROPOSED METHOD

This section we first introduce the visual perception determining model [7][9], and simplify the model for hardware implementation. Then we present the Sobel operator and the gradient vector computation by using Sobel operator [10-12]. At last we give our fast inter mode decision method based on section A and section B.

A. Visual Perception Determining Model

In [8][9][13], the authors suggest a model of visual threshold sensitivity and the model shows that the feelings of people about the object brightness is related with the brightness difference between the object and the background. Assume that there is a spot with the brightness I+ΔI, and the brightness of the surrounding pixels is I. According to [13], there is a function to describe the relationship between ΔI and I, and we call the relationship as threshold versus intensity (TVI) and call ΔI as just-noticeable difference (JND). According to [9] [13] [14], we know that $\Delta I / I$ (Weber fraction curve) is of the high (low contrast) in low intensity background and is also of the high in high intensity background, as shown as a concave shaped distribution in Weber fraction curve [13].

In this paper, we give a simplified Weber fraction curve model as shown in Fig. 2.

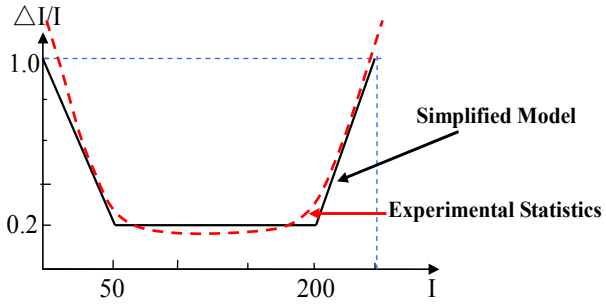

Figure 2. A simplified Weber fraction curve model

According to this simplified model, we can describe the Weber fraction function as below:

$$\Delta I / I = \begin{cases} -0.016I + 1 & I \le 50 \\ 1/5 & 50 < I \le 200 \\ 0.016I - 3 & I > 200 \end{cases} \quad (2)$$

Thus, we can get the function as shown in (3).

$$\Delta I = \begin{cases} -0.016I^2 + I & I \le 50 \quad (a) \\ 0.2I & 50 < I \le 200 \quad (b) \\ 0.016I^2 - 3I & I > 200 \quad (c) \end{cases} \quad (3)$$

This function is somewhat complicated because it includes quadratic computation. To reduce the complexity, we simplify (3) as shown below.

In (3) (a),

$$\Delta I = -0.016I^2 + I$$

$$= -0.016(I - 31.25)^2 + 15.625 \quad (4)$$

When I=31, ΔI achieves the maximum value 15.624; when I=0, ΔI achieves the minimum value 0.

Between the interval [0, 31], (3) (a) is an increasing function. We use (5) as linearly fitting the quadratic function.

$$\Delta I = 0.5I + 2.5 \quad (5)$$

Between the interval [32, 50], (3) (a) is a decreasing function. We use (6) as linearly fitting the quadratic function.

$$\Delta I = -0.3I + 26 \quad (6)$$

The simplify of (3)(a) as shown in Fig.3.

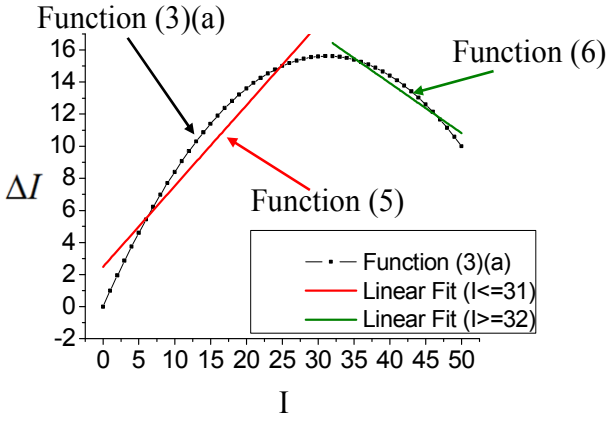

Figure 3. The simplify of Function (3)(a)

In (3) (c),

$$\Delta I = 0.016I^2 - 3I$$

$$= 0.016(I - 93.75)^2 - 140.625 \quad (7)$$

Between the interval $94, +\infty)$, (3)(c) is an increasing function. We use (8) as linearly fitting the quadratic function.

$$\Delta I = 4.25I - 815 \quad (8)$$

Finally, we simplify the function as (9) because its complexity is much lower and convenient to be implemented in hardware.

$$\Delta I = \begin{cases} \beta_1 I + \gamma_1 & I \le \text{a} \\ \beta_2 I + \gamma_2 & a < I \le \text{b} \\ \alpha I & b < I \le c \\ \beta_3 I + \gamma_3 & I > c \end{cases} \quad (9)$$

And the experimental results are similar to (3). Here I is a variable number between 0 and 255, a is 31, b is 50 and c is 200. α is 1/5, β_1 is 0.5, γ_1 is 2.5; β_2 is -0.3, γ_2 is 26; β_3 is 4.25, γ_3 is -815.

1) We use a 3x3 matrix $G_{surround}$, as shown in Fig. 4(a), to extract the brightness of the surrounding pixels.

$$G_{surround} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (10)$$

The surrounding brightness of pixel (x, y) can be computed as

$$I(x,y) = G_{surround} * A / 8 \quad (11)$$

Here A represent a 3x3 original pixel matrix shown in Fig. 4(b), whose center pixel brightness is f(x, y).

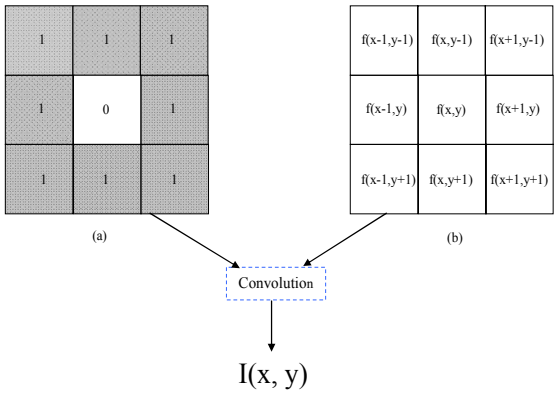

Figure 4. Extract the brightness of the surrounding pixels by convolution; (a) A 3x3 matrix Gsurround; (b) A 3x3 original pixel matrix

The operator * denotes convolution. According to (11), by computing the convolution of matrices $G_{surround}$ and A, then we can yield (12).

$$I(x,y) = \frac{\sum_{i=0}^{2}\sum_{j=0}^{2} f(x-1+i, y-1+j) \times G(i,j)}{8} \quad (12)$$

2) Substituting (12) into (9) can yield the vision threshold ΔI.

B. Sobel Operator and the Gradient Vector

In [10], the author uses the Sobel operator to detect the edge directional information of the images. Technically, it is a discrete difference operator for computing the image brightness function gradient approximation. Any point in the image using this operator will produce a gradient vector, and it is used to extract the edge information. Sobel operator can provide accurate edge direction information and good edge detection results, and it calculates the partial derivatives of x and y directions [11][12]. Its horizontal and vertical convolution operators are

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \text{ and } \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}.$$

$$G_{x\_grad} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * A \quad (13)$$

$$G_{y\_grad} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * A \quad (14)$$

Here $G_{\text{x\_grad}}$ is the horizontal gradient of pixel (x, y) and G_{y_grad} is the vertical gradient of pixel (x, y).

Then, normalize $G_{\text{x\_grad}}$ and G_{y_grad} by dividing 4 and we get:

$$G_{\text{x}} = | G_{x\_grad} | / 4 \quad (15)$$

$$G_{y} = | G_{y\_grad} | / 4 \quad (16)$$

C. Our Proposed Mode Decision Method

1) Calculate the number of edge points

Based on the description of A and B, we first calculate the number of the edge points. Here we define three types of edge points. If $G_{\text{x}} > \Delta I$ or $G_{\text{y}} > \Delta I$, then the pixel located at (x, y) is a MB level edge point (MLEP); If $G_{\text{x}} \le \Delta I$ and $G_{\text{y}} > \Delta I$, then the point located at (x, y) is a vertical edge point (VEP); If $G_{\text{x}} > \Delta I$ and $G_{\text{y}} \le \Delta I$, then the point located at (x, y) is a horizontal edge point (HEP). Then we choose the BIM from

16x16, 16x8, 8x16 and 8x8 based on the values of MLEP, VEP and HEP [15].

2) Mode Decision Based on the Number of Edge Points

The whole algorithm flow is shown in the following.

> Step 1) calculate the number of MLEP, the number of VEP and the number of HEP in one MB;
>
> Step 2) if the number of MLEP is no greater than 10, choose 16x16 mode as a BIM, else proceed to step 3;
>
> Step 3) if the number of MLEP is no greater than 128 but greater than 10, proceed to step 4, else proceed to step 6;
>
> Step 4) if the number of HEP is greater than VEP, choose 16x8 mode as a BIM, else proceed to step 5;
>
> Step 5) if the number of VEP is greater than HEP, choose 8x16 mode as a BIM;
>
> Step 6) choose 8x8 mode as a BIM.

In AVS, besides the four inter modes talked above, we also need to process direct mode and intra mode. After choosing the BIM from 16x16, 16x8, 8x16 and 8x8, we then get the three candidate modes (direct mode, intra mode, BIM) and calculate RD costs to select the best MB mode from the three candidate modes. Finally we can show the whole algorithm flowchart as Fig. 5.

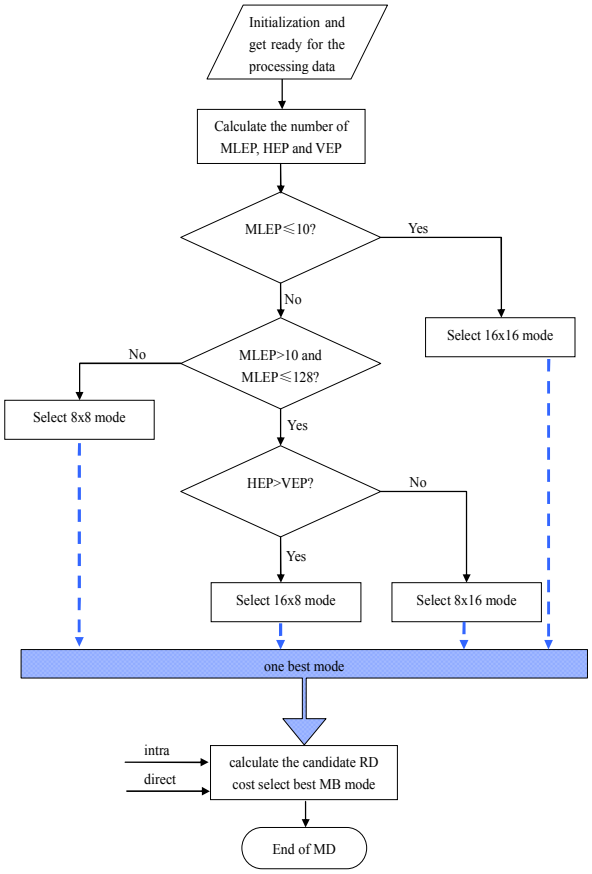

Figure 5. Overall algorithm flowchart

III. Our Proposed Mode Decision Architecture

A. Analyze and Redesign the Hardware Pipeline Structure

Generally, In MB level pipeline, MD is in the third-stage [16] as shown in Fig. 6, and according to the predicted pixels and original pixels from fractional motion estimation (FME) to make a inter mode decision. But FME requires a long processing time due to the various modes of interpolation, and the computational complexity of mode decision based on RDO is very high.

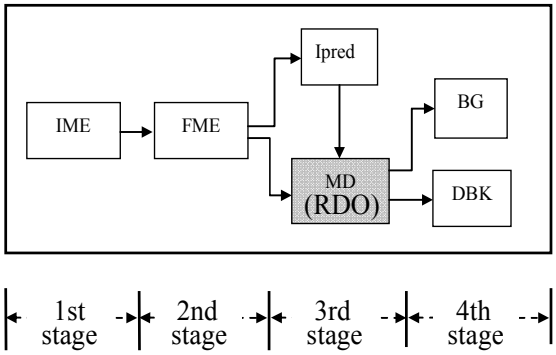

Figure 6. Top level pipeline of video encoder

In the new algorithm proposed, we could optimize the structure to save more hardware resources and processing time. By using the proposed method, we can select the BIM at the first MB stage of the encoder pipeline. By doing this FME module just need to interpolate for the BIM among the four inter modes. Meanwhile, the traditional RDO based MD method could reduce the candidate modes to improve encoding capacity even more. We divide the MD module into two part: MD1 and MD2, in MD1, we could use the proposed method to select the BIM from 16x16, 16x8, 8x16 and 8x8 by using the original pixels before the IME, thus FME module just need to process one BIM and one direct mode. In MD2, we just need to compare the RD costs for the three candidate modes: BIM, intra mode and direct mode, the new pipeline is shown in Fig. 7. Finally we optimize the original 4-stage pipeline structure to 5-stage.

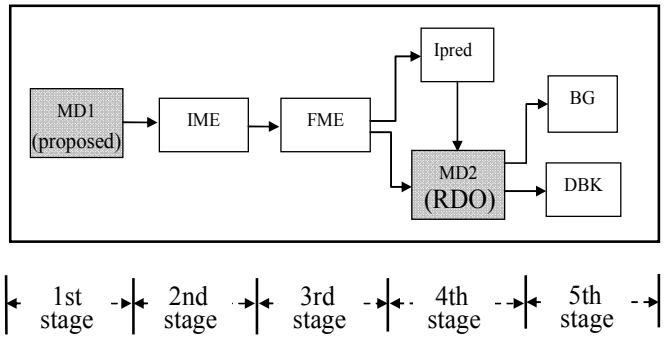

Figure 7. Proposed top level pipeline of video encoder

The original MB-level pipeline architecture of the encoder is shown in [6], and the optimized MB-level pipeline structure is shown in Fig. 8.

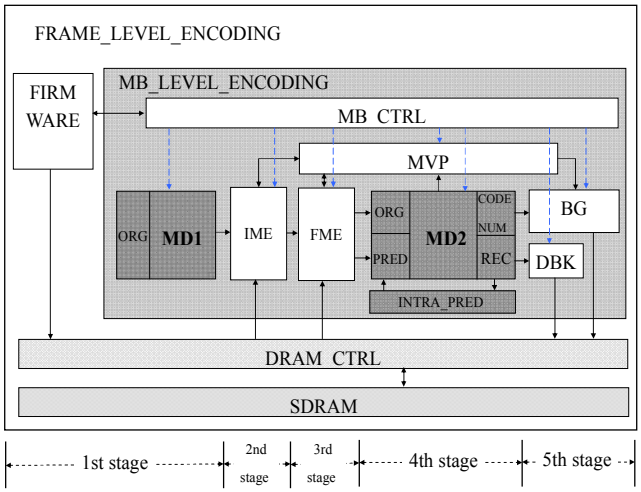

Figure 8. Optimized MB-level pipeline architecture

In Fig.8, MD module is integrated to the encoder system. As the figure shows, firmware takes on the central control of the Frame-level pipelining. MB_CTRL is for the control of the hardware part, which configures all the 5-stage MB-level pipelining structure, and they are IME, FME, MD, bit stream generating unit (BG) and de-blocking effect unit (DBK). We use Motion Vector Prediction (MVP) Schedule Strategy for AVS HD Encoder to solve the data dependency problem.

The proposed architecture is implemented on a Xilinx Vertex-XC6VLX760 FPGA prototyping system. It costs 6.5%(the count is 61969) slice LUTs and 9%(the count is 92953) slice registers. Having been embedded in an AVS HD encoder system, the mode decision module with synthesis frequency 200 MHz can support real time 1080p@30fps.

B. FME Performance Analysis

In AVS, traditional FME algorithm has four reference modes: the forward mode, backward mode, bi-directional mode and direct mode. Especially in three reference modes (forward mode, backward mode and bi-directional mode), we have to process current MB with all the partitions: 16x16, 16x8, 8x16 and 8x8. In order to meet the requirement of real-time encoding, the four partition modes need four interpolation circuits for parallel processing. In this structure, the parallel circuit consumption is very high. By using the algorithm proposed, the inter mode partition has been determined before FME, thus we just need to process one partition in forward, backward or bi-directional reference mode and the calculation is greatly reduced. If we treat each partition as a single MB unit, traditional FME system need forward, backward and bi-directional three reference modes for a total of 12 (3x4) MB units and one direct mode (direct mode has only 8x8 mode). In the proposed algorithm, FME totally only needs to deal with four MB units, and the calculation has been reduced by 70%; the circuit could be reused, meanwhile reducing the circuit area by 60%. The FME hardware architecture is shown in Fig. 9.

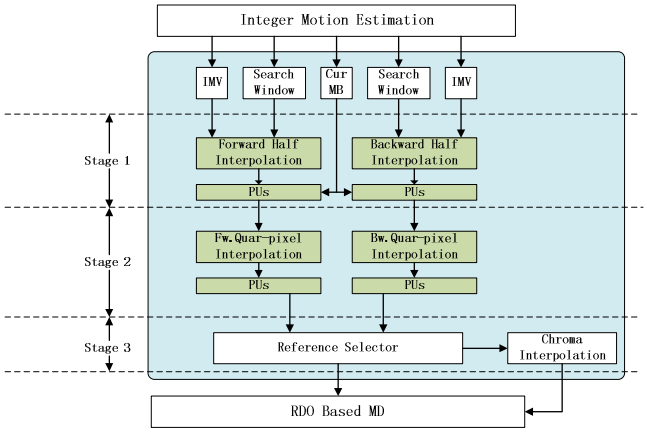

Figure 9. Hardware architecture of FME

There are three stages in FME pipeline structure: half interpolation, quarter interpolation, the direct mode interpolation. We use two-way parallel structure: one way for forward interpolation and one way for backward interpolation as shown in Fig. 10.

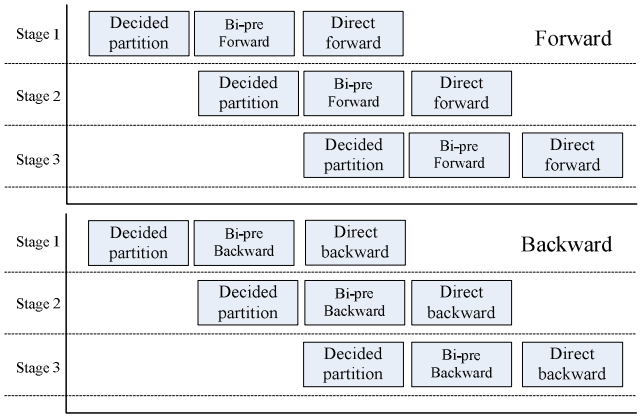

Figure 10. FME two-way parallel structure

The forward mode is performed in the forward circuit and the backward mode in the backward circuit. And in the bi-directional mode or direct mode, the forward or backward mode are performed in the both circuits. And the new structure saves the processing time by 200 cycles.

IV. EXPERIMENTAL RESULTS

TABLE I. TEST SEQUENCES

Sequence Name	Type	Resolution	Frame Rate	GOP
akiyo	cif	352×288	30fps	IPPP
foreman				
city	720p	1280×720	30fps	IPPP
night				
blue_sky	1080p	1920×1080	30fps	IPPP
sunflower				

In the experiment, we tested six sequences which are listed in Table 1, including three different resolutions: cif, 720p and 1080p.

RDO-based method is targeted for comparing with the proposed scheme. Table 2 illustrates the comparison results for encoding of six sequences in different quantization parameters (QP) among two algorithms: the original RDO-based reference software and the proposed method.

TABLE II. COMPARISON BETWEEN THE RDO-BASED METHOD AND PROPOSED ALGORITHM

Sequence	akiyo	city	blue_sky
ΔPSNR(dB)	-0.0519	-0.0921	-0.2194

Sequence	Foreman	Night	sunflower
ΔPSNR(dB)	-0.1839	-0.2025	-0.1293

The RD curves of the 6 sequences are shown in Fig. 11(a)-(f). As shown in Table 2, compared with RDO-based method, our proposed scheme have degression from -0.2194dB to -0.0519 dB for the 6 sequences, and the average degression is about 0.1098dB.

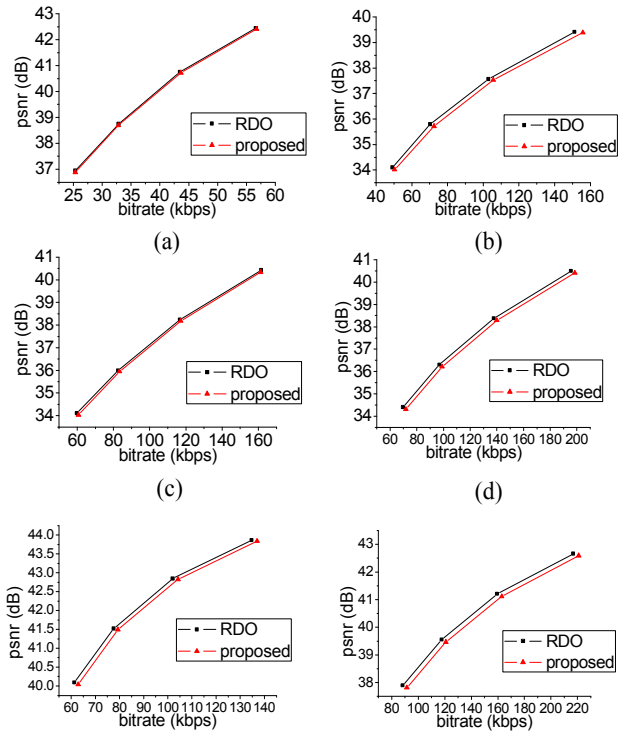

Figure 11. (a) RD curves of cif sequences "akiyo" in QP range from 28 to 40; (b) RD curves of cif sequences "foreman" in QP range from 28 to 40; (c) RD curves of 720p sequences "city" in QP range from 28 to 40; (d) RD curves of 720p sequences "night" in QP range from 28 to 40; (e) RD curves of 1080p sequences "sunflower" in QP range from 28 to 40; (f) RD curves of 1080p sequences "blue_sky" in QP range from 28 to 40

V. CONCLUSION

In video encoder, RDO plays an important role in mode decision, how to reduce its complexity is a top priority. In this paper, we have proposed a fast inter mode selection algorithm based on the visual perception determining model and Sobel operator edge detection model. The performance of the proposed algorithm is close to original RDO method. At the same time, we proposed an efficient MB-level pipeline structure. The proposed hardware structure saves FME 60% of its areas and reduces the processing time by 200 cycles. The proposed mode decision architecture can support real time processing of 1080p@30fps. In the future, we will further our study in mode decision to improve our chip design performance.

ACKNOWLEDGMENT

This work is partially supported by grants from the Chinese National Natural Science Foundation under contract No.61171139 and No. 61035001, and National Basic Research Program of China under contract No.2009CB320906.

REFERENCES

[1] China AVS Group, "Information Technology Advanced Audio Video Coding Standard Part 2: Video", 2003,12, AVS-N1063

[2] D. Wu, F. Pan, K. P. Lim, S. Wu, Z. G. Li, X. Lin, S. Rahardja, and C. C. Ko, "Fast Intermode Decision in H.264/AVC Video Coding," IEEE Transactions on Circuits and Systems for Video Technology, Vol.15, No.6, July 2005

[3] Yun Cheng, Silian Xie, Jianjun Guo, Zhiying Wang, Minlian Xiao, "A Fast Inter Mode Selection Algorithm for H.264," 1st International Symposium on Pervasive Computing and Applications, 2006

[4] Huaibin HUANG, Dong HU, "An Efficient Fast Inter Mode Decision Algorithm for H.264/AVC," International Conference on Communications and Mobile Computing, 2009

[5] Jie Yang, Yin Chen, "A Novel Fast Inter-mode Decision Algorithm for H.264/AVC Based on Motion Estimation Residual," International Conference on Information Engineering (ICIE), 2009

[6] Chuang Zhu, et al, "A Highly Efficient Pipeline Architecture of RDO-based Mode Decision Design For AVS HD Video Encoder," Multimedia and Expo (ICME), IEEE International Conference, 2011

[7] James A. Ferwerda, "Elements of Early Vision for Computer Graphics," IEEE Computer Graphics and Applications, 2001, 21(5):22—33

[8] J. Noll, A. S. Pandya, and W. Glenn, "A Visual Perception Threshold Matching Algorithm for Real-Time Video Compression," Manuscript received July IS. 1993

[9] Jiaji Cheng, et al, "A Predicted Compensation Model of Human Vision of Human Vision System for Low-light Image," 3rd International Congress on Image and Signal Processing (CISP), 2010

[10] F. Pan, X. Lin, S. Rahardja, E. P. Ong, W. S. Lin, "Using edge direction information for measuring blocking artifacts of images," Multidim Syst Sign Process 18:297–308, 2007

[11] Caixia Deng, et al, "An Edge Detection Approach of Image Fusion Based on Improved Sobel Operator," 4th International Congress on Image and Signal Processing (CISP), 2011

[12] Liquan Shen, et al, "Fast Inter Mode Decision Using Spatial Property of Motion Fiel," IEEE Transactions on Multimedia, 2008

[13] Salah Ameer, Otman Basir, "Modifying Weber Fraction Law to Postprocessing and Edge Detection Applications," 3rd International Symposium on Communications, Control and Signal Prosessing (ISCCSP), Malta, 12-14 March 2008

[14] W. Pratt, "Digital Image Processing," John Wiley & Sons Inc, 3rd edition, 2001, pp 30–32.

[15] Shao Juan, Zhang Weining, Chen Dong, et al, "Fast Macroblock Mode Decision Algorithm for B Frame in AVS," Computer Engineering and Applications, 2011, 47(5): 179-181

[16] Tung-Chien Chen, et al, "Hardware Architecture Design of an H.264/AVC Video Codec," Asia and South Pacific Conference on Design Automation (ASPCDAC), 2006