Variable Sub-pixel Motion Vector Resolution Based on Block Mode Motion Estimation for H.264/AVC

Trung-Kien Tran, *Dae-Yeon Kim, Yung-Lyul Lee
DMS Lab., School of Computer Engineering, Sejong University
e-mail : kien@dms.sejong.ac.kr, dykim@dms.sejong.ac.kr, yllee@sejong.ac.kr

Abstract

In H.264, sub-pel motion estimation (ME) has strong effect when coding video sequences. 1/4-pel performs better at low bitrate while 1/8-pel gives better results at high bitrate. In this paper, a variable sub-pixel motion vector resolution based on block mode motion estimation method is proposed. Experiment results show that the proposed method can take the advantage of 1/4-pel at low bitrate and 1/8-pel at high bitrate. In addition to that, time is reduced from 14% to 53% compared to KTA1.3 with 1/8-pel motion vector (MV) resolution.

I. INTRODUCTION

Jointly developed by the ITU-T Video Coding Experts Group and the ISO Moving Picture Experts Group, the H.264 AVC standard has achieved much higher coding efficiency compared to previous standards. However, its computational complexity is also significant higher. ME is the most important part in exploiting temporal redundancies, but it is most time consuming part in the coding framework. ME is conducted in two parts: the integer-pixel ME and the sub-pixel ME. Both parts are performed with tree structured variable block size and multi-reference frames. Depending on the configuration used, ME process can occupy about 60% ~ 90% of total computational time of the whole encoder.

In this paper, a variable sub-pixel MV resolution based on block mode is proposed for reducing the ME time without any performance loss.

II. OBSERVATION AND ANALYSIS

In order to see how 1/4-pel and 1/8-pel work when coding video sequences, we conducted experiments with JM11.0 (H.264 standard reference software with 1/4-pel MV resolution) and KTA1.3 (reference software with 1/8-pel MV resolution[1]). PSNR results and mode usage statistical information were collected to make comparison between the two encoders. From the experiment results, it's clear that at low bitrate JM_14 outperforms KTA_18, but at high bitrate KTA_18 outperforms JM_14.

In H.264, ME is performed in tree structured with 7 modes for inter-prediction. We can categorize the modes into 2 groups based on their block sizes: large block size modes (16x16, 16x8, and 8x16) and small block size modes (8x8, 8x4, 4x8, and 4x4). To see what made the difference in the performance of JM_14 and KTA_18, we examine the mode usage data collected from the experiments. And we can conclude:

-At low bitrate, JM_14 has better performance than KTA_18 and it uses more modes with large block size than KTA_18.

-At high bitrate, KTA_18 outperforms JM_14 and it uses more modes with large block size and fewer modes with small block size than JM_14.

So we can cut off the 1/8-pel refinement of modes with small block size without any compression loss and also reduce the encoding time.

III. PROPOSED ALGORITHM

From the observation, a simplified MV search procedure was proposed as described bellow:

-With 16x16, 16x8, 8x16 modes: search the best MV and refine up to 1/8-pel resolution.
-With 8x8, 8x4, 4x8 modes: search the best MV and refine up to 1/4-pel resolution.
-With 4x4 mode: search the best MV and refine to 1/2-pel resolution only.

By applying this simplified search procedure, we will make the encoder redistributes the modes usage similar to JM_14 at low bit rate and similar to KTA_18 at high bit rate.

In the proposed search method, each block may be coded with a different MV resolution. The predicted MV will be predicted from neighbouring blocks. It's clear that those neighbours can be previously encoded in different MV resolutions, so before predicting the MV, we must find and scale the MVs of the neighbours to the same resolution of current block.

Another change is the sub-pixel interpolation filters, three filters from KTA are used instead. Pixel values are interpolated in horizontal and vertical directions using filters:

H1 = { -3, 12, -37, 229, 71, -21, 6, -1 }
H2 = { -3, 12, -39, 158, 158, -39, 12, -3 }
H3 = { -1, 6, -21, 71, 229, -37, 12, -3 }

1/8 pixel values are generated using bilinear filter.

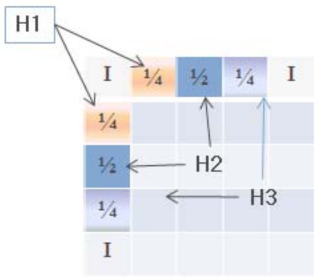

Fig. 1. Sub-pixel interpolation filters

Ⅳ. EXPERIMENT RESULTS

The proposed method was implemented based on JM11.0. Fig. 2 shows the RD curves of Foreman sequence. From the figure, it's clear that the PSNR of the proposed method is very close to that of JM_14 at low bit rate and the same as KTA_18 at high bit rate. And the proposed method can reduce the ME time compared to KTA with 1/8-pel MV resolution as shown in Table. 1.

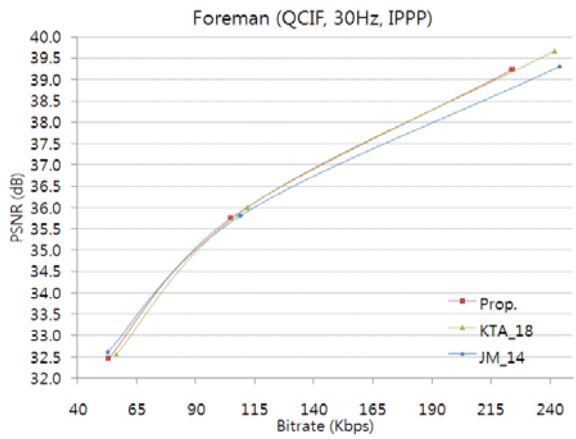

Fig. 2. RD curves of Foreman sequence

Table 1. ME time saving of the proposed method

Prop. vs	JM_14	KTA_18
Foreman(QCIF)	-2.8%	+32.3%
Carphone(QCIF)	-2.7%	+14.0%
Foreman(QCIF)	-3.4%	+26.7%
Mobile(QCIF)	-1.9%	+53.1%

Ⅴ. CONCLUSIONS

In this paper, a variable sub-pixel MV resolution based on block mode is proposed for taking the advantage of the 1/4-pel at low bit rate and 1/8-pel at high bit rate. As a result, the performance of the encoder is improved. Furthermore, it can significantly reduce the ME time of the encoder.

Acknowledgments

"This work was supported by the Korea Research Foundation Grant Funded by the Korea Government(MOEHRD)" (KRF-2006-211-D00264)

References

[1] Jörn Ostermann, Matthias Narroschke, "Motion compensated prediction with 1/8-pel displacement vector resolution", VCEG-AD09, 30th Meeting: Hangzhou, China, 23-27 Oct, 2006.