# A Fast Sub-pel Motion Estimation Scheme using a Parabolic SAD Model

*Sangsoo Ahn*[1], *Bumshik Lee*[1] and *Munchurl Kim*[1]
*Changseob Park*[2], *Sangjin Hahm*[2] and *Injun Cho*[2]

School of Electrical Engineering[1]
Information and Communication University
119, Muji St. Yuseong-gu, Deajeon, KOREA
E-mail: {ssahn, bslee, mkim}@icu.ac.kr

Technical Research Institute, Korean Broadcasting System[2]
11 Yoidodong, Youngdeungpo-gu, Seoul, KOREA
E-mail: {changseob, Caushy, sylphide}@icu.ac.kr

## ABSTRACT

Sub-pel level motion estimation contributes to significant increase in R-D performance for H.264|MPEG 4 Part 10 AVC. However, several supplements, such as interpolation, block matching, and Hadamard transform which entails large computational complexity of encoding process, are essential to find best matching block in sub-pel level motion estimation and compensation. In this paper, a fast motion estimation scheme in sub-pel accuracy is proposed based on a parabolic model of SAD to avoid such computational complexity. In the proposed scheme, motion estimation (ME) is only performed in integer-pel levels and the following sub-pel level motion vectors are found from the parametric SAD model for which the model parameters are estimated from the SAD values obtained in the integer-pel levels. Fall-back check is performed to ensure the validity of the parabolic SAD model with the estimated parameters. The experiment result shows that the proposed scheme can reduce the motion estimation time up to about 30% of the total ME times in average with negligible amount of PSNR drops (0.14dB in maximum) and bit increments (2.54%in maximum).

**Keywords:** H.264|MPEG 4 Part 10 AVC, Parabola model, Motion estimation.

## 1. Introduction

The H.264|MPEG-4 part 10 AVC has shown momentous improvements in terms of R-D performance by incorporating the sub-pel level (1/2 pel and 1/4 pel) motion estimation [1] into the encoding processing. Since practical motion of a moving object can occur not only in integer-pel level but also in arbitrary sub-pel level, it is appropriate to depict practical motion of a moving object in sub-pel level and this may enhance coding efficiency [2]. This is clearly shown in Fig. 1 in which R-D performance is significantly improved with 1/4-pel level compared to integer-pel level. However, as sub-pel level gets deeper, the number of search points where the encoder should compute R-D costs is also increased. For example, if 1/4-pel level motion estimation is used, after integer-pel searching process eight half pixels around a best integer pixel should be examined to find a best half pixel and then eight quarter pixels around the best half pixel are also checked to find the final best matching pixel. This means that sub-pel level motion estimation requires increased searching process with 16 points. In addition, to construct the pixels at sub-pel levels, interpolation operation is required which causes to increase computational complexity. Table 1 shows a relative portion of computational complexity for integer-pel and sub-pel motion estimations in time.

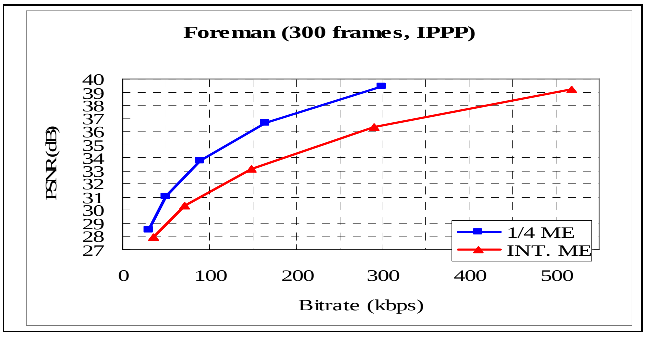

Fig. 1 R-D performance according to sub-pel level

Table 1 Proportion of sub pixel motion estimation time in total motion estimation time

| ME | Portions (%) |
|---|---|
| Sub pixel ME Time | 44 |
| Integer Pixel ME Time | 56 |

Due to proportion of sub-pel motion estimation time, much effort has been made to reduce the computational complexity on sub-pel motion estimation. Some previous works suggested cost function prediction methods using parabolic models. *Li* and *Gonzales* assumed that the cost function surface can be modeled by parabolic surfaces and, if using parabolic models, the minimum cost function for motion estimation can be predicted without full motion search for each macroblock [3].

However, the parabolic model may fail to predict exact cost functions because, according to characteristic of a moving object such as texture and motion speed, the cost function can form arbitrary surface. A fall back check method was proposed by *Hill et. al.* to avoid this failure [4]. After prediction of the cost function using a parabolic model, a predicted cost function is compared with a predefined threshold value whether it exceeds the threshold or not. If it exceeds the threshold, it will be then assumed that the predicted cost function is not correct so the motion estimation is performed without such a prediction. If the predicted cost is less than the threshold value, the predicted cost function is assumed to be correct so the encoders directly determine the final motion vectors from the model. In the *Hill*'s method, the SAD values are used to determine the threshold values. However, as the SAD values vary according to the characteristics of video contents, it is not easy to determine an optimal threshold value which is applicable for the characteristics of any video content.

In this paper, we propose a new method to overcome this weakness. The proposed method determines the threshold values in two ways: first, the threshold values are obtained by off-line training in different motion vector sizes and the finally selected block modes for test sequences; second, threshold values are obtained by on-line training. According to motion vector sizes and the finally selected block modes, the threshold values are updated during encoding process so that it can better adapt to the changing characteristic of video sequences than off-line training.

The remaining part of this paper is organized as follows: In Section 2, parabolic models are described with the comparison between the full motion search and the motion vector estimation. A fall back check method is also explained; then, our proposed method is described for off-line training and on-line training. For the fall back check, we propose a thresholding scheme; Section 3 shows experiments results with analysis; finally, Section 4 provides conclusion and future work.

## 2. Model Description and Training Methods

For real time applications of software video encoders, it is essential to accelerate the motion estimation process without considerable R-D performance degradation. The method used in this paper is to predict the cost function in sub-pel levels using a parabolic model. Fig. 2 shows a practical cost function surface that can be modeled as a parabolic form. The final motion vector can be found at the location where the cost function value is minimum.

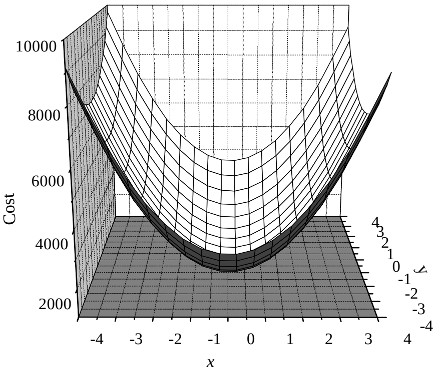

Fig. 2 A cost function surface by a parabolic model

### 2.1 A parabola Model for cost function in motion search

A mathematical model for the parabola surface is given by [4] such as

$$S(x,y) = Ax^2 + By^2 + Cxy + Dx + Ey + F \tag{1}$$

For the H.264|MPEG-4 Part 10 AVC encoders, $S$ is the cost function in SAD or SATD of residues for a macroblock, and $x$ and $y$ are the center position of the macroblock. The parabola model coefficients $A$, $B$, $C$, $D$, $E$, and $F$ can be computed from the SAD values of the eight neighbor integer-pel points where motion estimation is performed *a priori*. The nearest eight neighbors are shown in Fig 3 where the SAD values at the eight pixel locations are found. The center position of a macroblock is (0, 0).

| | | |
|---|---|---|
| 5(-1,1) | 6(0,1) | 7(1,1) |
| 4(-1,0) | 8(0,0) | 0(1,0) |
| 3(-1,-1) | 2(0,-1) | 1(1,-1) |

Fig. 3 Center and its eight nearest neighbor pixels position in a macroblock

In [4], the model parameter values are found by Eq. (2). Some parameters such as $A$ and $B$ can be determined in two ways in Eq. (2).

$$\begin{aligned}
C &= \frac{1}{4}\{S(d_3)+S(d_7)-S(d_1)-S(d_5)\} \\
D &= \frac{1}{4}\{S(d_1)+S(d_7)-S(d_3)-S(d_5)\} \quad \text{or} \\
E &= \frac{1}{4}\{S(d_5)+S(d_7)-S(d_1)-S(d_3)\} \\
A &= S(d_2)-D-F \\
A &= S(h_1)+D-F \\
B &= S(v_1)+E-F \quad \text{or} \\
B &= S(v_2)-E-F
\end{aligned} \tag{2}$$

The best candidates for the parameters $A$ and $B$ can be chosen using the predicted SAD and the original SAD in integer pixel unit at the pixel position where the minimum difference between the predicted SAD and the original SAD is obtained among the pixel position 1, 3, 5, and 7 in Fig. 8.

### 2.2 Validity check of parabolic models

Sometime, the parabolic model may fail to model the cost function surface, especially in a moving object area. Hill *et. al.* proposed a fall back check method to examine the validity of the parametric mode by comparing with a predefined threshold the difference in Eq. (3) between the predicted cost function values and the block matching based cost function value in integer-pel.

$$Diff = \sum_{i\in\{1,3,5,7\}} \left|S_i - S_{ki}\right| \tag{3}$$

$S_i$ and $S_{ki}$ are the predicted and original SAD values, respectively. If *Diff* exceeds a predefined threshold value, the estimated SAD by the parabolic model is not used. Instead, the full motion vector search is performed. If *Diff* is less than the threshold value, then it is assumed that the parabolic model can reasonably predict the SAD function which leads to direct determination of the motion vector in sub-pel levels.

### *2.2.1 Off-line learning of threshold values*

In this paper, for the fallback check on the estimated parabolic model, a threshold value, *Th*, is defined as the ratio of average minimum SAD value in integer-pel levels to the average minimum SAD value in 1/4-pel levels, both of which are obtained by the motion vectors based on full search (MVFS), and is given by

$$Th = \frac{Average\ SAD_{MVFS@1\text{-}pel}}{Average\ SAD_{MVFS@1/4\text{-}pel}} \tag{4}$$

With the SAD values in integer-pel levels for a MB after motion estimation, the parabolic model parameters for the MB are estimated. Then, the final motion vector is estimated based on the parabolic model with the estimated model parameter. We define a model validity index (*MVI*) as the ratio of two minimum SAD values by

$$MVI = \frac{SAD_{MVFS@1\text{-}pel}}{SAD_{MVPM@1/4\text{-}pel}} \tag{5}$$

where $SAD_{MVFS@1\text{-}pel}$ and $SAD_{MVMP@1/4\text{-}pel}$ are the minimum SAD values obtained by the motion vectors based on full search in integer-pel levels and based on the parabolic model in 1/4-pel levels, respectively. Fallback check is performed such that the estimated parabolic model is valid if $MVI > Th$, otherwise it's not valid.

Fig 4 shows the distributions of *Th* values versus different motion vector magnitudes for various kinds of different video sequences. Therefore, *Th* is obtained according to different motion vector magnitude groups.

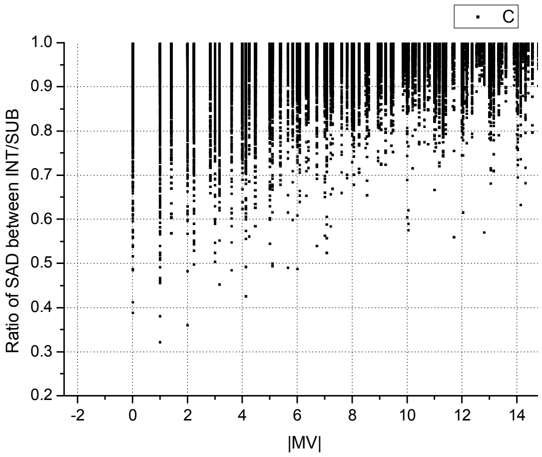

Fig. 4 *Th* distributions versus motion vector magnitudes ( |MV|)

Table 2 indicates the averages of *Th* values for different motion vector magnitude groups. The motion vector magnitude groups are MVG 1, MVG 2, MVG 3, MVG 4, MVG 5, MVG 6, MVG 7, MVG 8 and MVG 9 for the ranges of the motion vector magnitudes with 0, 0~2, 2~4, 4~6, 6~8, 8~10, 10~12, 12~14 and above 14, respectively.

Table 2 *Th* value for nine motion vector groups for four different QP values for a training set

| QP | 24 | 28 | 32 | 36 |
|---|---|---|---|---|
| MVG 1 | 0.585 | 0.589 | 0.601 | 0.623 |
| MVG 2 | 0.752 | 0.723 | 0.702 | 0.697 |
| MVG 3 | 0.742 | 0.715 | 0.702 | 0.703 |
| MVG 4 | 0.730 | 0.716 | 0.706 | 0.708 |
| MVG 5 | 0.750 | 0.732 | 0.715 | 0.713 |
| MVG 6 | 0.743 | 0.736 | 0.735 | 0.736 |
| MVG 7 | 0.727 | 0.745 | 0.749 | 0.764 |
| MVG 8 | 0.896 | 0.876 | 0.860 | 0.837 |
| MVG 9 | 0.725 | 0.740 | 0.741 | 0.777 |

Fig. 5 shows the scatters of the predicted SAD values by the parabolic model and the original SAD values by full search. The deviations away from the diagonal line indicate the mismatch between the parabolic SAD model and the original SAD maps. If the deviations are compensated, then more accurate fallback check can be performed.

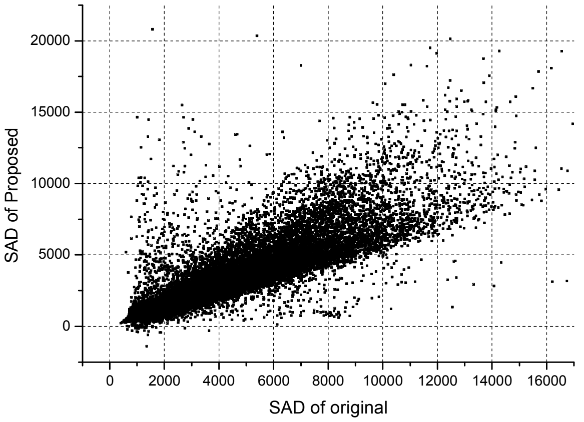

Fig. 5 A scatters plot of the predicted SAD values by the parabolic model and the original SAD values by full search

Table 3 tabulates the average MCF values for nine different motion vector groups. It can be noticed that the average MCF values vary according to different motion vector magnitude groups. Therefore, this is taken into account for fallback check on the parametric model.

Table 3 Average MCF values for nine different motion vector groups

| QP | 24 | 28 | 32 | 36 |
|---|---|---|---|---|
| MVG 1 | 1.784 | 1.741 | 1.679 | 1.605 |
| MVG 2 | 1.514 | 1.543 | 1.560 | 1.560 |
| MVG 3 | 1.487 | 1.513 | 1.520 | 1.516 |
| MVG 4 | 1.465 | 1.472 | 1.487 | 1.494 |
| MVG 5 | 1.440 | 1.449 | 1.469 | 1.476 |
| MVG 6 | 1.470 | 1.466 | 1.462 | 1.459 |
| MVG 7 | 1.460 | 1.440 | 1.424 | 1.410 |
| MVG 8 | 1.300 | 1.308 | 1.332 | 1.354 |
| MVG 9 | 1.457 | 1.416 | 1.435 | 1.398 |

For fallback check on the parabolic model, *MVI* in Eq. (5) is first compensated by a model compensation factor (MCF) prior to comparing it with *Th* defined in Eq. (4). The MCF is defined as

$$MCF = \frac{SAD_{MVFS@1\text{-}pel}}{SAD_{MVPM@1\text{-}pel}} \tag{6}$$

where $SAD_{MVPM@1\text{-}pel}$ is the minimum SAD values obtained by the motion vectors based on the parabolic model in integer-pel levels. Therefore, the validity check on the estimated parabolic model is modified as

$$MVI \cdot MCF \underset{\text{Not valid}}{\overset{\text{Valid}}{\substack{> \\ \leq}}} Th \tag{7}$$

### 2.2.2 On-line learning of threshold values

Since the characteristics of video contents vary frame by frame or content by content, the SAD values also vary. Therefore, usage of a global threshold value may not be appropriate for different kinds of video sequences. In this paper, we propose an on-line update of threshold values. To obtain a threshold value for each MVG, *Th* is trained on-line as a moving average for each MVG.

$$Th(n+1) = Th(n) + \frac{1}{N}\left\{ \frac{SAD(n)_{MVFS@1\text{-}pel}}{SAD(n)_{MVFS@1/4\text{-}pel}} + \frac{SAD(n-N)_{MVFS@1\text{-}pel}}{SAD(n-N)_{MVFS@1/4\text{-}pel}} \right\} \tag{6}$$

where $N$ is the window size and $Th(n)$ is a threshold value at time *n*. *Th* is updated on-line for each MVG for each MB mode for the MBs with the motion vector full search . Fig 6 summarizes our proposed scheme as a flowchart.

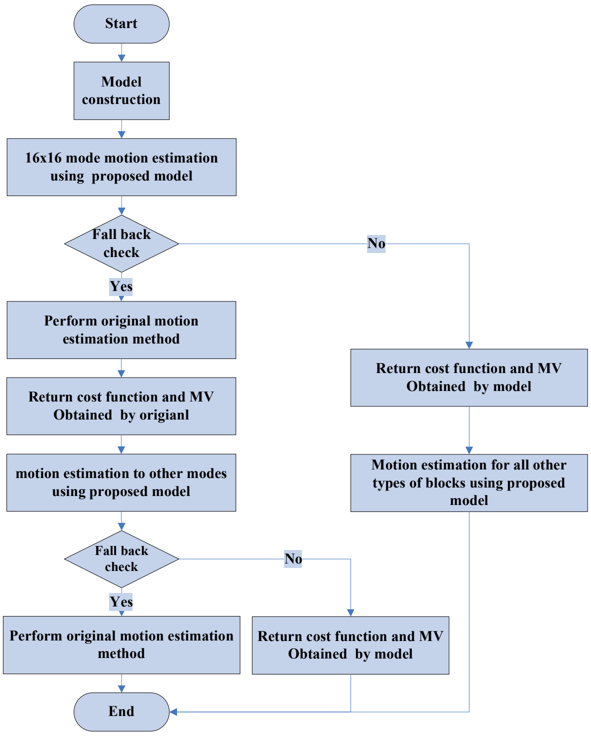

Fig. 6 A flow chart of proposed method

First, after integer-pel motion estimation, a parabolic model is constructed and SAD function is predicted using the parabolic model at 16x16 mode. Next, the ratio between the predicted minimum SAD cost in quarter-pel level and the minimum SAD value in integer-pel level is examined, and then the ratio is compared with threshold value. If the ratio is less than threshold value, the predicted cost function is kept without any further refinement. However, if the ratio exceeds the threshold value, the full search is performed for motion estimation. If the parabolic model turns out to predict the SAD cost function precisely, the model applies to all other block matching modes without any other further check. However, if the full search for motion estimation method is used at 16x16 mode instead of the parabolic model, the validity of parabolic model is evaluated at all other modes.

## 3. Experimental Results

H.264|MPEG-4 Part 10 AVC reference software, Joint Model (ver11.0) is used and the platform used for the experiment is a PC with Intel core™ 2 2.4GHz and 2.4GHz CPU and 2 GB RAM. All sequences are in CIF format and the frame rate of the sequences is 30frame/sec. The baseline profile of H.264|MPEG-4 Part 10 AVC is used which includes I- and P-frame coding only, variable block size matching and CAVLC. Single and multiple reference frame scheme are used.

The R-D performance is compared for the original JM, a modified JM with off-line training and another modified JM with on-line training for *Football*, *Foreman*, and *Mother&Daughter* sequences. Fig. 7 shows their R-D performances.

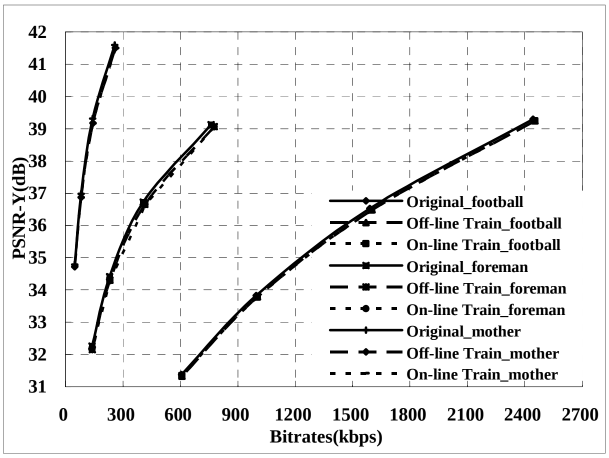

Fig. 7 R-D Performance of Sequences in Single Reference Frame Case

As shown in Fig. 7 there is no significant drop in R-D performance between the original JM and the modified JM's.

Table 4 Execution time savings taken for sub-pel motion estimation : 1 reference is used for Baseline profile.

| | QP | 24 | 28 | 32 | 36 |
|---|---|---|---|---|---|
| *Football* | Off-line | 23.42 | 21.04 | 21.04 | 27.06 |
| | On-line | 22.57 | 33.82 | 34.77 | 19.16 |
| *Foremen* | Off-line | 16.11 | 44.73 | 34.20 | 59.05 |
| | On-line | 24.83 | 43.05 | 45.34 | 45.84 |
| *Mother& Daughter* | Off-line | 39.35 | 60.41 | 54.99 | 59.42 |
| | On-line | 28.54 | 73.31 | 60.77 | 55.09 |

Table 4 shows the time savings on sub-pel motion estimation for the modified JM with off-line training and on-line training. Even though there is fluctuation according to sequences, time saving in sub-pel motion estimation is about 20~30% for *Football*, about 15~60% for *Forman*, and almost 40~60% for *Mother&Daughter* sequences for the modified with off-line training method. The modified JM with on-line training yields the time savings with about 19~35% % for *Football*, about 24~45% for *Forman*, and 28 ~73% for *Mother* & 4 sequences.

Table 5 Bit rate and PSNR Variation

| | Off-line Train | | On-line Train | |
|---|---|---|---|---|
| | Bitrate(%) | PSNR(dB) | Bitrate(%) | PSNR(dB) |
| *Football* | 0.70 | -0.04 | 0.66 | -0.05 |
| *Foreman* | 2.06 | -0.07 | 2.54 | -0.08 |
| *Mother* | 0.62 | -0.13 | 0.73 | -0.14 |
| *Coast* | 0.74 | -0.04 | 0.59 | -0.04 |
| *Hall* | -1.16 | -0.06 | -0.78 | -0.07 |
| *Crew* | 1.07 | -0.06 | 1.11 | -0.07 |

Table 5 demonstrates bit rate fluctuation and PSNR drops for the off-line training and the on-line training. The bit rate varies between -1.16~2.06% for the off-line training and -0.78~2.54% for the on-line training. The PSNR drops are in the range between -0.04 and -0.13dB for the off-line training and -0.04~0.14dB for the on-line training. The R-D performance drop is negligible.

Table 6 Sub-pel Motion Estimation Time Saving

| | Off-line Train | On-line Train |
|---|---|---|
| | Time Saving(%) | Time Saving(%) |
| Football | 21.04 | 33.82 |
| Foreman | 44.73 | 43.05 |
| Mother | 60.41 | 73.31 |
| Coast | 33.94 | 21.41 |
| Hall | 77.67 | 75.20 |
| Crew | 38.29 | 55.13 |

Table 6 reveals sub-pel motion estimation time reduction in two training cases. Sub-pel motion estimation time is reduced from 21.04% to 77.67% for the off-line trained threshold and 21.41% to 75.20% for the on-line trained threshold. The on-line training method is more advantageous than off-line training when the characteristics of video changes rapidly such as *Football* and *Crew* sequences.

Proportion of sub-pel motion estimation in total encoding time gets enlarged as the number of reference frames increases. So the proposed fast sub-pel motion estimation method more reduces the total encoding times as the number of reference frames increases. Fig. 8 exhibits the relative times taken for motion estimation versus the number of reference frames. With one single reference frame, motion estimation just occupies about 19% of the total encoding time but, with 3 reference frames, it occupies almost 50% of the total encoding time. The deeper the sub-pel levels get, the more the motion estimation time is taken.

Fig. 9 shows the time savings of the total encoding time. As the number of reference frame increases, the amount of time saving is increased.

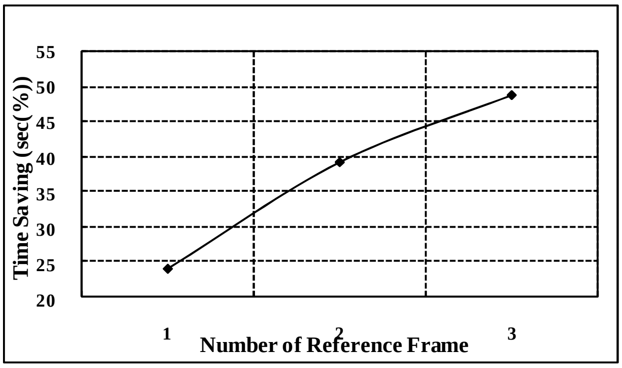

Fig. 8 Proportion of ME times vs. number of reference frames

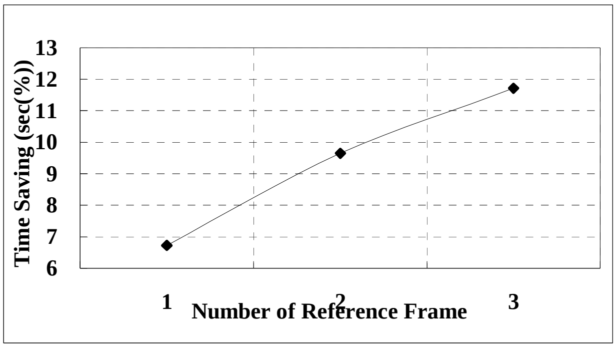

Fig. 9 Time savings vs. number of reference frames

## 4. Conclusion

Sub-pel motion estimation contributes to improve R-D performance in H.264|MPEG-4 Part 10 AVC encoders, but it causes large computational complexity. To accelerate the encoding process, parabolic model based motion estimation is proposed and a method to examine the validity of proposed model is also introduced. For the validity check, the on-line trained threshold is more advantages to fast moving areas, which can accelerate the encoding speed with negligible R-D performance drop.

## 5. References

[1] Thomas Wedi and Hans Georg Musmann, “Motion-and Aliasing-Compensated Prediction for Hybrid Video Coding” CSVT. Vol.13, No.7 July 2003.

[2] Zengyou Wang and Quan Xue et al, “Fast Algorithm for Sub-Pixel Motion Estimation Using Parabola Model for H.264/MPEG4 AVC” Wireless Communications, Networking and Mobile Computing, 2006. WiCOM 2006.International Conference on , vol., no., pp.1-4, 22-24 Sept. 2006.

[3] Xiaoming Li, Cesar Gonzales.: “A Locally Quadratic Model of the Motion Estimation Error Criterion Function and Its Application to Subpixel Interpolations”, IEEE Trans. Circuits Syst. Video Technol.,vol. 6, no. 1, pp. 118–122, FEB. 1. 1996.

[4] P.R.Hill, T.KChiew. et al: “Interpolation Free Subpixel Accuracy Motion Estimation”, IEEE Trans. Circuits Syst. Video Technol.,vol. 16, no. 12, pp. 1519–1525, DEC. 2006.