TEMPORAL SEARCH RANGE PREDICTION BASED ON A LINEAR MODEL OF MOTION-COMPENSATED RESIDUE

Liwei Guo †, Oscar C. Au †, Mengyao Ma ‡, Zhiqin Liang † and Peter H. Wong †

†Dept. of Electrical and Computer Engineering, ‡Dept. of Computer Science and Engineering
Hong Kong University of Science and Technology
email:{eeglw, eeau, myma, zhiqin, eepeter}@ust.hk

ABSTRACT

An efficient temporal search range prediction method is proposed to reduce the complexity of multiple reference frames motion estimation (MRFME) in video coding. Based on a linear model of motion-compensated residue, the behavior of residues under MRFME is investigated, and the gain of multiple reference frames is analyzed. The proposed method utilizes the current residue to estimate the gain of searching more reference frames, and predicts the temporal search range that maintains the coding performance with minimum complexity. Experimental results show that the proposed scheme can significantly reduce the complexity in motion estimation while the degradation of the coding performance is negligible.

Index Terms— Multiple reference frames motion estimation (MRFME), video modelling, video coding

1. INTRODUCTION

Motion estimation (ME) plays a key role in video coding. In ME, the best-match in the reference frame is found to predict the current video block, and only the prediction error (motion-compensated residue) needs to be encoded. Traditionally, only one reference frame is used in ME. The state-of-art video coding standard H.264 extends the temporal search range by utilizing multiple reference frames [1]. While significantly improve the coding efficiency, multiple reference frames motion estimation (MRFME) works at the expense of high complexity that linearly increases with the number of reference frames, which restricts its applicability. It is thus desirable to find the optimal temporal search range which can maintain the coding performance with minimum complexity.

Abundant efforts have been made on developing efficient temporal search range prediction methods [2, 3, 4]. By investigating the relation between the reference frame buffer utilization and the temporal search range, a content-adaptive scheme was proposed in [3] to dynamically control the search range. Huang $et\ al$ [4] applied statistical analysis to investigate the relation between the temporal search range and the available encoder information. Based on extensive experiments, a scheme [4] was proposed where the temporal search range is determined based on available encoder information such as intra prediction cost and motion vectors.

In this work, we present a linear model of motion-compensated residue, and propose a temporal search range prediction scheme based on this model. With the presented residue model, the relation between the behavior of residues under MRFME and the gain of multiple reference frames is investigated. The proposed scheme utilizes the current residue to estimate the gain of searching more reference frames, and determine the temporal search range. Experimental results show that the proposed scheme can save up to 75% of ME complexity with negligible degradation in the coding performance. Moreover, the proposed scheme can be combined with any fast block matching algorithms to accelerate the ME further.

The rest of this paper is organized as follows: Section 2 analyzes the temporal search range of MRFME based on a linear model of motion-compensated residue. The proposed algorithm is described in Section 3. Experimental results are shown in Section 4 and the conclusion is given in Section 5.

2. ANALYSIS OF TEMPORAL SEARCH RANGE OF MRFME BASED ON RESIDUE MODEL

2.1. Multiple Reference Frames Gain (MRFGain)

For different sequences, the performance improvements of MRFME are different. For quantitative evaluation, we define multiple reference frames gain (MRFGain) to be the average PSNR improvement of MRFME relative to single reference frame ME over a number of sampling QPs, where the average PSNR improvement is calculated according to [5]. Table 1 shows MRFGain of some test sequences.

MRFME is a very computationally intensive module in video encoder. For the sequences with small MRFGain, searching all the reference frames may only provide very limited

This work has been supported in part by the Innovation and Technology Commission (projects no. GHP/033/05) of the Hong Kong Special Administrative Region, China.

Table 1. MRFGain (dB) (TT=Table Tennis, MC=Mobile & Calender, FG=Flower Garden)

Akiyo	FG	Foreman	MC	Tempete	TT
0.2	0.36	0.44	1.03	0.71	0.15

performance improvement, while the complexity is much higher than that of single reference frame ME. So in this case, a short temporal search range is expected to avoid wasting computational power. On the contrary, for those sequences with large MRFGain, to search each more reference frame may increase the coding performance rather much, so a large temporal search range is more desirable than a short one.

In the next section, a linear model of motion-compensated residue is proposed. Later we will use this model to estimate MRFGain and determine the temporal search range.

2.2. A Linear Model of Motion-Compensated Residue

Suppose F is the current frame and its reference frames are the previous ones: $\{Ref(1), Ref(2), \ldots Ref(k), \ldots\}$, where k is the temporal distance between F and reference frame $R(k)$. Let s be a pixel in F and $p(k)$ be its prediction from $Ref(k)$. The motion-compensated residue is denoted as $r(k)$, $r(k) = s - p(k)$. Residue $r(k)$ is assumed to be a random variable with zero-mean and variance $\sigma_r^2(k)$.

We assume that $r(k)$ can be decomposed as

$$r(k) = r_t(k) + r_s(k), \tag{1}$$

where $r_t(k)$ is the temporal innovation between F and $Ref(k)$, and $r_s(k)$ is the sub-integer pixel interpolation error in reference frame $Ref(k)$.

Let $\sigma_{r_t}^2(k)$ and $\sigma_{r_s}^2(k)$ be the variances of $r_t(k)$ and $r_s(k)$ respectively. Assuming that $r_t(k)$ and $r_s(k)$ are independent,

$$\sigma_r^2(k) = \sigma_{r_t}^2(k) + \sigma_{r_s}^2(k). \tag{2}$$

When the temporal distance k increases, the temporal innovation between the current frame and the reference frame tends to be larger, causing $\sigma_{r_t}^2(k)$ to increases. We assume that $\sigma_{r_t}^2(k)$ linearly increases with k,

$$\sigma_{r_t}^2(k) = C_t \cdot k, \tag{3}$$

where C_t is the increasing rate of $\sigma_{r_t}^2(k)$ with respect to k.

When an object moves with a non-integer pixel displacement, *i.e.*, non-integer pixel motion, between reference frame $Ref(k)$ and current frame F, the sampling positions of the object in F and $Ref(k)$ may be different. In this case, the prediction pixels from $Ref(k)$ are at sub-integer locations and have to be interpolated using those at integer positions, incurring sub-integer pixel interpolation error $r_s(k)$. Obviously, $r_s(k)$ should not be related to the temporal distance k, so we model $\sigma_{r_s}^2(k)$ using a k-invariant parameter C_s, $\sigma_{r_s}^2(k) = C_s$.

Therefore, a linear residue model is proposed

$$\sigma_r^2(k) = C_s + C_t * k. \tag{4}$$

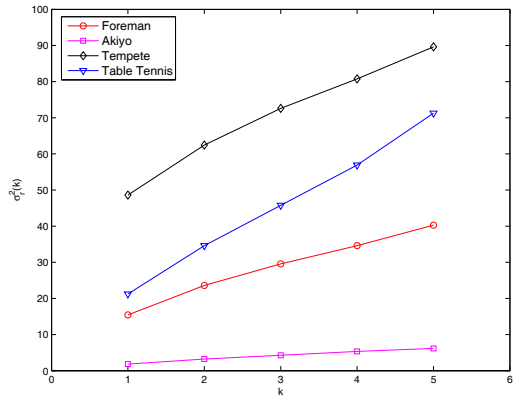

Fig. 1. Experimental relationship of $\sigma_r^2(k)$ and k

Table 2. Model Parameters for some test sequences

	Akiyo	FG	Foreman	MC	Tempete	TT
C_t	1.07	29.57	6.07	20.07	9.44	23.23
C_s	0.94	31.6	10.50	56.81	41.89	9.24

A number of test sequences are encoded by H.264 encoder with MRFME, and the residue variances corresponding to different reference frames are calculated and compared with the proposed model. As $r(k)$ is assumed to be zero-mean, we approximate $\sigma_r^2(k)$ using $r^2(k)$ averaged over the whole sequence. Due to limited space, only part of experimental results are shown in Fig.1. It can be seen that although quite different in video contents, for all the sequences, the relation between $\sigma_r^2(k)$ and k appears to be quite linear, which is consistent with the proposed model. The best-fit C_s and C_t for some test sequences are shown in Table 2.

2.3. Analysis of MRFGain using the Proposed Model

In video coding, block-level motion estimation is performed. We define the block residue energy as $\overline{r^2(k)}$, which is $r^2(k)$ averaged over the block. Normally, smaller $\overline{r^2(k)}$ means better prediction and leads to higher coding performance. In MRFME, if $\overline{r^2(k+1)}$ is smaller than $\overline{r^2(k)}$, searching more reference frames can improve performance.

We define $\overline{r_t^2(k)}$ and $\overline{r_s^2(k)}$, which are $r_t^2(k)$ and $r_s^2(k)$ averaged over the block respectively. As $r_s(k)$ and $r_t(k)$ are independent, we have $\overline{r^2(k)} \approx \overline{r_s^2(k)} + \overline{r_t^2(k)}$. To analyze MRFGain, the behaviors of $\overline{r_t^2(k)}$ and $\overline{r_s^2(k)}$ with increasing k are investigated as follows.

When the temporal distance increases, on one hand, the temporal innovation between frames tends to increase, and thus $r_t(k+1)$ tends to have larger amplitude than $r_t(k)$, giving rise to $\overline{r_t^2(k+1)} > \overline{r_t^2(k)}$. On the other hand, it is possible that the object in the current frame F have non-integer pixel motion with respect to $Ref(k)$, but integer pixel motion with respect to $Ref(k+1)$. In this case, while there is sub-integer

pixel interpolation error in $r(k)$, $i.e.$, $\overline{r_s^2(k)} > 0$, the interpolation error in $r(k+1)$ is zero, $i.e.$, $\overline{r_s^2(k+1)} = 0$.

Let $\Delta_t = \overline{r_t^2(k+1)} - \overline{r_t^2(k)}$ and $\Delta_s = \overline{r_s^2(k)}$. Suppose the object in F has integer pixel motion with respect to $Ref(k+1)$, which means $\overline{r_s^2(k+1)} = 0$. When extending temporal search range from from $Ref(k)$ to $Ref(k+1)$, the increase of residue energy $\Delta(k)$ would be

$$\begin{aligned}\Delta(k) &= \overline{r^2(k+1)} - \overline{r^2(k)} \\ &= \left(\overline{r_t^2(k+1)} - \overline{r_t^2(k)}\right) + \left(\overline{r_s^2(k+1)} - \overline{r_s^2(k)}\right) \\ &= \left(\overline{r_t^2(k+1)} - \overline{r_t^2(k)}\right) + \left(0 - \overline{r_s^2(k)}\right) \\ &= \Delta_t(k) - \Delta_s(k). \end{aligned} \quad (5)$$

Obviously, if $\Delta_t(k) < \Delta_s(k)$, $\Delta(k)$ would be negative, meaning searching one more reference frame $Ref(k+1)$ would result in smaller residue energy and improve coding performance. Furthermore, for large $\Delta_s(k)$ and small $\Delta_t(k)$, large residue energy reduction (MRFGain equivalently) tends to be achieved.

The values of $\Delta_s(k)$ and $\Delta_t(k)$ are related to the parameters of the proposed model, $i.e.$, C_s and C_t. Parameter C_s is the interpolation error variance $\sigma_{r_s}^2(k)$. Therefore, for video signal with large C_s, $r_s(k)$ tends to have large amplitude, and thus $\Delta_s(k) = \overline{r_s^2(k)}$ tends to be large. Parameter C_t is the increasing rate of $\sigma_{r_t}^2(k)$. Hence, for video signal with small C_t, $\sigma_{r_t}^2(k)$ and $\sigma_{r_t}^2(k+1)$ tends to be similar, so $\Delta_t(k) = \overline{r_t^2(k+1)} - \overline{r_t^2(k)}$ tends to be small. Based on the above analysis, it seems that for video signal with large C_s and small C_t, the corresponding MRFGain tends to be large. On the contrary, in the case of small C_s and large C_t, MRFGain tends to be small.

To validate the above analysis, we compare C_s and C_t shown in Table 2 with MRFGain in Table 1. As predicted by our analysis, large MRFGain is observed for video sequences with relatively large C_s and relatively small C_t, such as Mobile & Calendar, while for video sequence with relatively small C_s and relatively large C_t, such as Table Tennis, the MRFGain is small. Inspired by this, we define $G = C_s/C_t$ as an estimation of MRFGain.

3. THE PROPOSED SCHEME

With the analysis in Section 2, we propose a simple yet efficient block-level temporal search range prediction method based on the estimation of G for every block.

We suppose MRFME is performed in a time-reverse manner, with $Ref(1)$ being the first to be searched. For different $Ref(k)$ ($k > 1$ vs $k = 1$), the estimation methods of G are different, which are described as follows.

Suppose the current reference frame is $Ref(k)$ ($k > 1$) and the search on this frame has been finished. To determine if the next reference frame $Ref(k+1)$ should be searched, we will estimate C_s and C_t from the available information $\overline{r^2(k-1)}$ and $\overline{r^2(k)}$. Statistically $\overline{r^2(k)}$ converges to $\sigma_r^2(k)$. Therefore, we use $\overline{r^2(k)}$ as the estimation of $\sigma_r^2(k)$. Substituting $\overline{r^2(k-1)} = \sigma_r^2(k-1)$ and $\overline{r^2(k)} = \sigma_r^2(k)$ into (4), parameters C_s and C_t can be easily obtained, and the corresponding $G = C_s/C_t$ is

$$G = \frac{k \cdot \overline{r^2(k-1)} - (k-1) \cdot \overline{r^2(k)}}{\overline{r^2(k)} - \overline{r^2(k-1)}}. \quad (6)$$

If the current reference frame is $Ref(1)$ ($k = 1$), $\overline{r^2(k-1)}$ is not available, so we cannot calculate C_s and C_t using (6). In this case, we will evaluate $\overline{r^2(1)}$ and the mean of residues in the block $\overline{r(1)}$ to estimate G. As sub-integer pixel interpolation filter a is a low-pass filter (LF), it cannot recover the high frequency (HF) component in the reference frame so that the HF of the current block cannot be compensated. As a result, the interpolation error tends to have small LF component and large HF component. Therefore, if $\overline{r(1)}$ is small and $\overline{r^2(1)}$ is large, $i.e.$, the residue has small LF component and large HF component, the dominant component in the residue should be $r_s(k)$, meaning large C_s and small C_t, $i.e.$, large G. Hence, in this case G is estimated using

$$G = \gamma \cdot \frac{\overline{r^2(1)}}{(\overline{r(1)})^2}, \quad (7)$$

where factor γ is tuned from training data. For different sequence, a fixed value of $\gamma = 6$ is used.

The value of G is compared with a predefined threshold T_G. If G is larger than T_G ($G > T_G$), probably searching more reference frame will improve the performance, so ME continues with $Ref(k+1)$; otherwise ($G \leq T_G$), MRFME of the current block terminates, and the rest reference frames will not be searched. Obviously, the higher the T_G is, the more computation is saved; the lower the T_G is, the less performance drop is achieved.

Apart from G, based on our analysis in Section 2, motion vector (MV) is also used to determine the temporal search range. For $Ref(k)$, if the found best motion vector $MV(k)$ is an integer pixel MV, probably the object has integer motion between $Ref(k)$ and F. According to our analysis, there is no sub-pixel interpolation error in $\overline{r^2(k)}$, and thus it would be difficult to find a better prediction in the rest reference frames. So MRFME of the current block terminates.

The proposed algorithm can be summarized as follows:

1. Set $k = 1$ (first reference frame $Ref(1)$). Perform ME with respect to $Ref(k)$, $MV(k)$, $\overline{r^2(1)}$ and $\overline{r(1)}$ can be obtained. Estimate G using (7). If $G \leq T_G$ or $MV(k)$ is an integer pixel MV, MRFME of the current block terminates; otherwise, go step 2;

2. Set $k = k + 1$ (move to the next reference frame). Perform ME with respect to $Ref(k)$, $MV(k)$ and $\overline{r^2(k)}$ can be obtained. Estimate G using (6). If $G \leq T_G$ or $MV(k)$ is an integer pixel MV, MRFME of the current block terminates;

Table 3. Akiyo

QP	JM+Ref5		FMRFME[4]			Proposed		
	BR	PSNR	BR	PSNR	$\overline{Ref}$	BR	PSNR	$\overline{Ref}$
26	118	40.61	120	40.53	1.20	120	40.54	1.17
28	85	39.34	87	39.24	1.11	87	39.28	1.22
30	62	37.99	63	37.88	1.06	62	37.96	1.27
32	45	36.63	46	36.50	1.03	45	36.61	1.31

Table 4. Foreman

QP	JM+Ref5		FMRFME[4]			Proposed		
	BR	PSNR	BR	PSNR	$\overline{Ref}$	BR	PSNR	$\overline{Ref}$
26	671	37.28	685	37.19	2.36	686	37.20	2.08
28	488	35.99	499	35.88	2.00	500	35.90	1.75
30	351	34.64	359	34.51	1.72	358	34.55	1.63
32	256	33.34	261	33.19	1.49	261	33.25	1.58

otherwise, go step 2.

For more efficient implementation, $\overline{r^2(k)}$ in the proposed scheme can be replaced by video encoder ME output $Mcost$, which is a function of residue to measure the prediction accuracy and has high correlation with $\overline{r^2(k)}$. Our experiments show that this will not cause performance loss while the calculation of $\overline{r^2(k)}$ can be saved.

4. EXPERIMENTAL RESULTS

Four CIF sequences, including Akiyo, Foreman, Flower Garden and Mobile & Calendar, are used in the experiment. The proposed scheme was integrated into H.264 encoder JM 10.2 to encode these sequences. Two other schemes are also used for comparison. One is the original JM 10.2 with 5 reference frames (JM+Ref5). The other one is the temporal search range prediction method proposed in [4], which we denote as FMRFME.

The coding performance is measured by bit rate (BR) in terms of kbps, and PSNR in terms of dB. The complexity of ME is measured by the average number of searched reference frames, which we denote as $\overline{Ref}$. Table 3-5 summarizes the experimental results of the test sequences. It can be seen that compared to JM+Ref5, the proposed scheme can significantly reduce the number of searched reference frames, while the coding performance decreasing very mildly.

For better comparison between FMRFME [4] and the proposed algorithm, Table 7 shows their average PSNR loss relative to JM+Ref5, which we denote as Δ PSNR, and the average complexity reduction $\Delta\overline{Ref}$. The Δ PSNR is calculated according to [5], and the calculation of $\Delta\overline{Ref}$ is given in Table 7. Compared to FMRFME [4], the proposed scheme tends to achieve better coding performance while searching fewer reference frames, such as the case of Flower Garden.

5. CONCLUSION

In this paper, an efficient temporal search range prediction method for MRFME is presented. First, a linear model of motion-compensated residue is presented. Then, the behavior of residues under MRFME is investigated. The proposed method analyzes the current prediction residue to determine if it is necessary to search more reference frames. Experimental results show that up to 75% of the complexity of ME can be saved by the proposed method, while the degradation of performance is negligible.

Table 5. Mobile & Calender

QP	JM+Ref5		FMRFME[4]			Proposed		
	BR	PSNR	BR	PSNR	$\overline{Ref}$	BR	PSNR	$\overline{Ref}$
26	2202	35.26	2236	35.24	3.78	2259	35.23	3.09
28	1650	33.61	1684	33.57	3.59	1718	33.56	2.76
30	1182	31.81	1210	31.72	3.34	1229	31.74	2.71
32	824	30.13	851	30.06	2.98	854	30.03	2.68

Table 6. Flower Garden

QP	JM+Ref5		FMRFME[4]			Proposed		
	BR	PSNR	BR	PSNR	$\overline{Ref}$	BR	PSNR	$\overline{Ref}$
26	2601	35.30	2613	35.28	3.40	2616	35.28	2.74
28	1982	33.56	1992	33.54	3.30	1994	33.54	2.37
30	1437	31.65	1445	31.61	3.18	1445	31.63	2.26
32	1013	29.87	1018	29.82	3.05	1017	29.84	2.21

6. REFERENCES

[1] T. Wiegand, G. J. Sullivan, G. Bjøtegaard, and A. Luthra, "Overview of the h.264/avc video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, July 2003.

[2] A. Chang, O. C. Au, and Y.M. Yeung, "A novel approach to fast multi-frame selection for h.264 video coding," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2003, pp. 413–416.

[3] Z. Liang, J. Zhou, O. C. Au, and L. Guo, "Content-adaptive temporal search range control based on frame buffer utilization," in *Proc. International Workshop on Multimedia Signal Proccessing*, Oct. 2006, pp. 399–402.

[4] Y. Huang, B. Hsieh, S. Chien, S. Ma, and L. Chen, "Analysis and complexity reduction of multiple reference frames motion estimation in h.264/avc," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, Apr. 2006.

[5] G. Bjøtegaard, "Calculation of average psnr differences between rdcurves," Joint Video Team (JVT),VCEG-M33.

Table 7. Average PSNR loss and complexity reduction relative to JM+Ref5 ($\Delta\overline{Ref} = (\overline{Ref} - 5) \cdot 100\%$).

Sequence	FMRFME[4]		Proposed	
	Δ PSNR	$\Delta\overline{Ref}$	Δ PSNR	$\Delta\overline{Ref}$
Akiyo	-0.19	-78.00%	-0.10	-75.15%
Foreman	-0.06	-62.15%	-0.06	-64.80%
Mobile & Calender	-0.19	-31.60%	-0.25	-45.60%
Flower Garden	-0.06	-35.40%	-0.05	-52.10%