

NEW SCALING TECHNIQUE FOR DIRECT MODE CODING IN B PICTURES

Xiangyang Ji¹, Debin Zhao¹, Wen Gao¹, Yan Lu², Siwei Ma¹

Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100080, China¹
 Microsoft Research Asia, Beijing, 150001, China²
 {xyji, dbzhao, wgao, swma}@jdl.ac.cn; t-yanlu@microsoft.com

ABSTRACT

To leave the maximum flexibility in encoder to optimize the trade-off between coding performance and complexity, in video coding standards such as H.264/AVC [1], H.263 [2] and MPEG-4 [3] etc, any number of B pictures and any arrangement of P pictures within a GOP of arbitrary length are permitted. In addition, the multiple reference picture prediction is also permitted in some video coding systems such as H.264/AVC to achieve the efficient coding by allowing the encoder to select reference pictures among a large number of coded pictures. Both of the above cases without fixing the temporal distance between forward and backward reference pictures will require the division operation for deriving the motion vectors of direct mode, which can efficiently exploit the temporal correlation among pictures and does not require any bits for coding the motion vectors. However, the division is an expensive and undesired operation in video decoder hardware design. Although H.264/AVC video standard has provided a scaling technique to tackle this problem, unfortunately, its performance is also deteriorated. This paper presents a new scaling technique to both remove the division operation for deriving direct mode motion vectors and efficiently improve the accuracy of derived direct mode motion vectors compared with the scaling technique in H.264/AVC.

1. INTRODUCTION

In the hybrid video coding, B pictures can achieve higher compression ratio by more effectively exploiting the temporal correlation between the reference pictures and the current B picture. B pictures also can be coarsely quantized in some application because the B picture is not used as a reference picture and does not propagate errors. In other words, B pictures play an important role in video coding in term of both coding performance and video transmissions system, especially real-time video transmissions service.

The direct mode takes advantage of bi-directional prediction and does not need to transmit its forward and backward motion vectors. In the decoder, its forward and backward motion vectors are derived from that used in its co-located block in the backward reference picture (P picture). As illustrated in Fig.1, forward and backward direct mode motion vectors MV_F and MV_B are derived as follows:

$$MV_F = \frac{TRb}{TRp} \times MV_D, \quad (1)$$

$$MV_B = \frac{TRb - TRp}{TRp} \times MV_D, \quad (2)$$

where TRb denotes the temporal distance between the current B picture and the forward reference picture, TRp denotes the temporal distance between the forward and backward reference pictures, and MV_D denotes the motion vector of the co-located block in the backward reference picture.

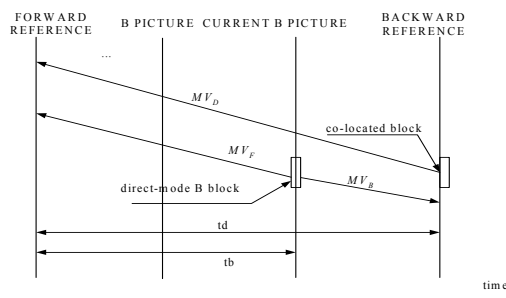


Fig.1. the direct mode block with forward and backward motion vectors MV_F and MV_B derived from motion vector MV_D of the co-located block

In general, when using multiple reference picture prediction, the backward reference picture for the current direct mode block is the most recent temporally subsequent picture (P picture) and the forward reference picture for the current direct mode block is the same as the reference picture of the co-located block in the backward reference picture [4]. Therefore, the temporal distance between the forward reference picture and the backward reference picture varies with the different reference pictures used for the co-located block in the backward reference picture. In addition, even in the case of using only one forward and one backward reference pictures, increasing the number of B pictures will make more pictures employ bi-prediction so as to efficiently exploit the temporal correlation between the reference pictures and the current B picture for the video sequence with simply translational motion. Whereas, the less B pictures is used when the motion in video sequence is high. Therefore, adaptive number of B pictures will also lead to different temporal distance between reference pictures (I/P pictures). Considering both the above cases may require the division operation for deriving the direct mode motion vectors, H.264/AVC proposed a scaling technique without division operation for deriving direct

mode motion vectors as follows:

$$tx = (16384 + \text{ABS}(TRp) / 2) / TRp$$

$$\text{DistScaleFactor} = \text{Clip}(-1024, 1023, (TRb * tx + 32) \gg 6)$$

$$MV_F(i) = (\text{DistScaleFactor} * MV_D(i) + 128) \gg 8$$

$$MV_B(i) = MV_F(i) - MV_D(i)$$

where function ABS is used to get the absolute value of TRp , function Clip is used to restrict DistScaleFactor within range between -1024 and 1023, $MV_F(i)$, $MV_B(i)$, $MV_D(i)$ separately represent the x-coordination or the y-coordination component of forward motion vector MV_F , backward motion vector MV_B of the direct mode block and the motion vector MV_D of the co-located block in the backward reference picture.

This paper presents a new scaling technique to remove the division operation for deriving the motion vectors of direct mode. The purpose is to efficiently improve the accuracy of the derived direct mode motion vectors compared with the scaling technique in H.264/AVC. The rest of this paper firstly presents the proposed method in detail, and then presents the simulated results to evaluate the performance of the proposed algorithm.

2. SCALING TECHNIQUE

Considering direct mode is mainly used to efficiently solves the simply translational motion with constant velocity, scaling technique for deriving direct mode motion vectors should keep the trajectory of object motion from one picture to another picture as a straight line as possible. This method can mainly be classified as three cases in implementations.

Case 1: When the motion vector of the co-located block in the backward reference picture points to temporally previous reference picture of the current B picture, the current B picture will be temporally between the forward reference picture and the backward reference picture, as shown in Fig.1.

The direct mode forward and backward motion vectors MV_F , MV_B can be derived from the following formulas:

$$\text{if}(MV_D(i) < 0)$$

$$MV_F(i) = -(((\text{SCALE_FACTOR} / TRp) \times (1 - MV_D(i) \times TRb) - 1) \gg \text{SHIFT_LEN})$$

$$\text{else}$$

$$MV_F(i) = ((\text{SCALE_FACTOR} / TRp) \times (1 + MV_D(i) \times TRb) - 1) \gg \text{SHIFT_LEN}$$

$$\text{if}(MV_D(i) < 0)$$

$$MV_B(i) = -(((\text{SCALE_FACTOR} / TRp) \times (1 - MV_D(i) \times TRd) - 1) \gg \text{SHIFT_LEN})$$

$$\text{else}$$

$$MV_B(i) = ((\text{SCALE_FACTOR} / TRp) \times (1 + MV_D(i) \times TRd) - 1) \gg \text{SHIFT_LEN}$$

where

$$\text{SCALE_FACTOR} = 2^{\text{SHIFT_LEN}}$$

$$TRd = TRp - TRb,$$

\gg is an arithmetic right shift operator.

$\text{SCALE_FACTOR}/TRp$ is a set of pre-calculated values based on the temporal distance TRp (always a positive value in our method) between forward and backward reference pictures. We

can select one of them according to parameters TRp in implementations instead of employing division operation.

Case 2: When multiple backward reference prediction is allowed and the backward reference picture with co-located block points to the other temporally subsequent reference picture of the current B pictures, as shown in Fig. 2, a pair of forward and backward motion vectors for the current direct mode block is derived from the following formulas:

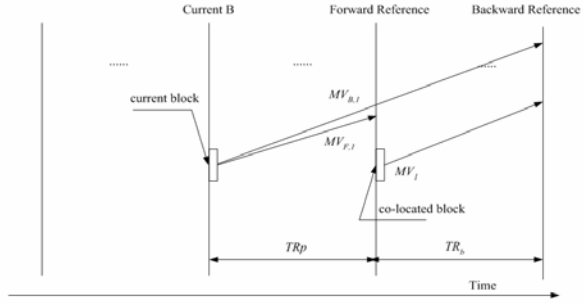


Fig.2 the motion vector of the co-located block points to temporally subsequent reference picture

$$\text{if}(MV_D(i) < 0)$$

$$MV_F(i) = -(((\text{SCALE_FACTOR} / TRp) \times (1 - MV_D(i) \times TRb) - 1) \gg \text{SHIFT_LEN})$$

$$\text{else}$$

$$MV_F(i) = ((\text{SCALE_FACTOR} / TRp) \times (1 + MV_D(i) \times TRb) - 1) \gg \text{SHIFT_LEN}$$

$$\text{if}(MV_D(i) < 0)$$

$$MV_B(i) = -(((\text{SCALE_FACTOR} / TRp) \times (1 - MV_D(i) \times TRd) - 1) \gg \text{SHIFT_LEN})$$

$$\text{else}$$

$$MV_B(i) = ((\text{SCALE_FACTOR} / TRp) \times (1 + MV_D(i) \times TRd) - 1) \gg \text{SHIFT_LEN}$$

where $TRd = TRp + TRb$.

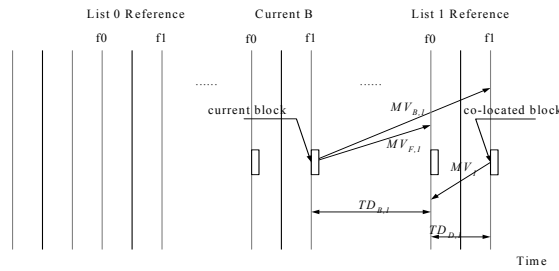


Fig.3 the motion vector of the co-located block in field 1 of the backward reference frame points to field 0 of the same frame.

Case 3: In field mode coding, the backward reference picture (field picture) of the current direct mode block is temporally subsequent reference picture (field reference picture) with the same parity. When the backward reference picture of the current direct mode block in field 1 is the temporally subsequent reference picture (field 1), as shown in Fig. 3, a pair of forward

and backward motion vectors for current direct mode block in field 1 are derived from the following formulas:

if($MV_D(i) < 0$)

$$MV_F(i) = ((SCALE_FACTOR / TRp) \times (1 - MV_D(i) \times TRb) - 1) \gg SHIFT_LEN$$

else

$$MV_F(i) = -(((SCALE_FACTOR / TRp) \times (1 + MV_D(i) \times TRb) - 1) \gg SHIFT_LEN)$$

if($MV_D(i) < 0$)

$$MV_B(i) = ((SCALE_FACTOR / TRp) \times (1 - MV_D(i) \times TRd) - 1) \gg SHIFT_LEN$$

else

$$MV_B(i) = -(((SCALE_FACTOR / TRp) \times (1 + MV_D(i) \times TRd) - 1) \gg SHIFT_LEN)$$

where $TRd = TRp + TRb$.

For case 1, Table 1 shows the comparisons of scaling technique deriving direct mode motion vector between H.264/AVC and proposed method with different parameters TRp and TRb , when the motion vector of the co-located block in backward reference picture is equal to (11,-17). It verifies the forward and backward direct mode motion vector MV_F , MV_B derived by our scaling technique can be more effectively keep at a straight line so as to represent the true motion trajectory.

Table.1 deriving direct mode motion vector comparisons between H.264/AVC and proposed

	motion vector	TRb=1, TRP=2 (one B picture)	TRb=1, TRP=3 (two B pictures)	TRb=1, TRP=4 (three B pictures)	TRb=2, TRP=3 (two B picturese)	TRb=3, TRP=4 (three B pictures)
H.264 /AVC	MV_F	(6,-8)	(4, -6)	(3, -4)	(7,-11)	(8,-13)
	MV_B	(-5,9)	(-7,11)	(-8,13)	(-4, 6)	(-3, 4)
new	MV_F	(5,-8)	(3, -5)	(2, -4)	(7,-11)	(8,-12)
	MV_B	(-5,8)	(-7,11)	(-8,12)	(-3, 5)	(-2, 4)

3. SIMULATED RESULTS

To evaluate the general performance of the proposed scaling technique for deriving the direct mode motion vectors compared with the existing method in H.264/AVC. We have integrated the proposed method into the H.264/AVC reference software JM750b [5]. SCALE_FACTOR with 1024 and both IBPBP... and IBBPBBP... GOP structures with two reference pictures, three reference pictures, and five reference pictures are tested, respectively. The test sequences include *foreman*, *coastguard*, *paris*, *tempete*, *mobile*, *flower*, *stefan* and *akiyo* with CIF format at 30fps. The test conditions are showed in Table 2. In addition, to evaluate the average PSNR vs. bit-rate, we employ the method described in [6], which is widely used during H.264/AVC development.

Since the direct mode derives its forward and backward motion vectors by scaling motion vector used in co-located block, improving the accuracy of derived motion vector by our proposed scaling technique can efficiently save the bits coding motion vectors in B pictures, as shown in Fig. 5. In addition, Fig. 6 illustrates that the number of blocks (in 8×8 units) coded with direct mode in each B picture in sequences of mobile and flower is almost always larger than that of the direct mode in

H.264/AVC when IBBPBBP... GOP structure is used. In other words, it verifies that the improving of the accuracy of the derived direct mode motion vector can efficiently increase the numbers of the direct mode in each B picture, and consequently, obtain the better prediction values.

Table 3 shows that the performance of our method is generally better than the existing scaling technique in H.264/AVC when IBPBP... GOP structure with the different number of reference pictures is used. The maximum PSNR gain is up to 0.535dB in terms of B pictures and up to 0.191dB for total pictures. Meanwhile, it also shows that the performance of our method is also generally better than the scaling technique in H.264/AVC when IBBPBBP... GOP structure with the different number of reference pictures is used. The maximum PSNR gain is up to 0.358dB in terms of B pictures and up to 0.178dB for total pictures. All the detailed results further verify our method can generally achieve better coding performance compared with the scaling technique in H.264/AVC.

Table 2. Test conditions

MV resolution	1/4 pel
Hadamard	ON
RD optimization	ON
Search Range	± 16
Restrict Search Range	2 (no restrictions)
Symbol Mode	CAVLC
QP	20,24,28,32
Loop Filter	ON

4. CONCLUSIONS

The proposed scaling technique for deriving direct mode motion vectors not only removes the division operation for deriving direct mode motion vector, but also efficiently improves the accuracy of derived direct mode motion vectors. As a set of pre-calculated values, which only require a few memory, based on temporal distance between the forward reference picture and backward reference picture, SCALE_FACTOR/TRp always accounts for minor cost of video decoder hardware design compared with division operation, and therefore it can simplify the derivation of motion vectors.

5. ACKNOWLEDGMENTS

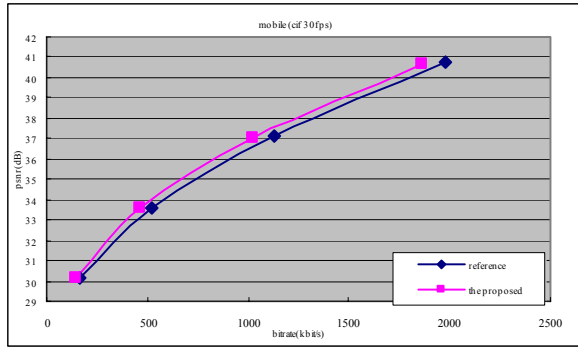
This work has been supported by National Science Foundation of China (60333020), and National Hi-Tech Development Program of China (2002AA119010).

6. REFERENCES

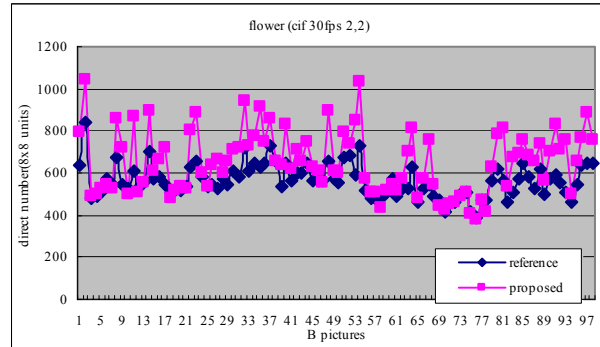
- [1] ITU-T Video Coding Experts Group and ISO/IEC Moving Picture Experts Group, Study of Final Committee Draft of Joint Video Specification (ITU-T Rec. H.264, ISO/IEC 14496-10 AVC), Mar. 2003, JVT-G050r1, [ftp://ftp.intmfiles.org/jvt-experts/](http://ftp.intmfiles.org/jvt-experts/)
- [2] ITU-T Recommendation H.263, "Video coding for low bit-rate communication," Nov. 1995

[3] ISO/IEC JTC1/SC29/WG11 N2502, FDIS of ISO/IEC14496-2, "Generic coding of audio-visual objects: Part 2- Visual," (MPEG-4), Nov. 1998.
 [4] M.Flierl and B.Girod. "Generalized B Pictures and the Draft H.264/AVC Video Compression Standard", IEEE

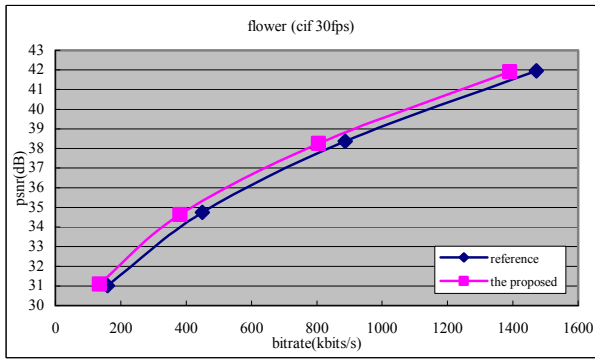
Transactions on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 587-597, July 2003.
 [5] Joint Video Team (JVT) Reference Software, Version 750b, <http://bs.hhi.de/~suehring/tml/download/>
 [6] G. Bjontegaard, "Calculation of average PSNR differences between RD-Curves", Doc. VCEG-M33, Mar. 2001



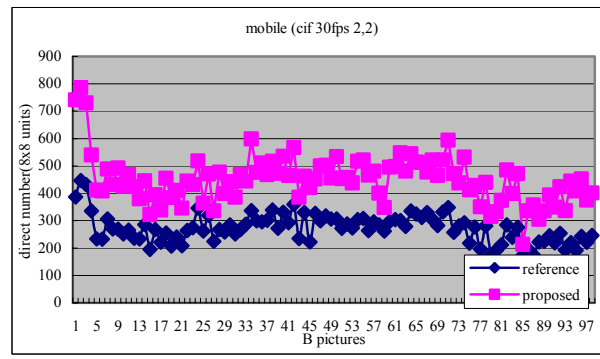
(a)



(a)



(b)



(b)

Fig.5. Rate-distortion curves for B pictures in sequences (a) mobile and (b) flower when two forward reference pictures and one backward reference picture is used with IBBPBBP... GOP structure is used

Fig.6. Number of blocks coded with direct mode in each B picture for sequences (a) flower and (b) mobile when two forward reference pictures and one backward reference picture is used with IBBPBBP... GOP structure is used

Table 3. Performance comparisons for scaling technique between H.264/AVC and the proposed algorithm

Video sequence			IBBPBBP...								IBBPBBPBBP...							
			akiyo	coastguard	flower	foreman	mobile	paris	stefan	tempe	akiyo	coastguard	flower	foreman	mobile	paris	stefan	tempe
One forward reference	B pictures	Ave PSNR gain	0.805	0.349	0.615	0.049	0.535	0.355	0.320	0.064	0.254	0.032	0.345	0.130	0.019	0.178	0.153	0.394
		Ave BR Saving	19.66%	7.68%	11.02%	1.34%	10.17%	7.78%	8.71%	1.46%	4.61%	0.87%	6.82%	2.91%	0.42%	3.78%	3.29%	7.99%
	All pictures	Ave PSNR gain	0.150	0.084	0.176	0.014	0.191	0.100	0.115	0.031	0.112	0.014	0.174	0.060	0.010	0.094	0.046	0.125
		Ave BR Saving	4.33%	1.96%	3.41%	0.38%	3.90%	2.28%	3.21%	0.68%	2.23%	0.38%	3.64%	1.31%	0.22%	2.10%	1.08%	2.80%
two forward reference	B pictures	Ave PSNR gain	0.902	0.346	0.601	0.046	0.487	0.393	0.356	0.069	0.244	0.024	0.358	0.133	0.018	0.196	0.138	0.380
		Ave BR Saving	22.03%	7.73%	10.60%	0.59%	9.42%	8.40%	9.68%	1.55%	4.51%	0.66%	7.08%	3.05%	0.38%	4.31%	2.99%	7.85%
	All pictures	Ave PSNR gain	0.161	0.082	0.167	0.001	0.158	0.107	0.122	0.029	0.102	0.015	0.178	0.061	0.009	0.094	0.042	0.123
		Ave BR Saving	4.67%	1.91%	3.25%	0.08%	3.25%	2.42%	3.53%	0.64%	2.04%	0.40%	3.62%	1.33%	0.21%	2.11%	0.98%	2.66%
four forward reference	B pictures	Ave PSNR gain	0.983	0.337	0.575	0.037	0.489	0.369	0.331	0.067	0.230	0.019	0.320	0.123	0.012	0.201	0.117	0.420
		Ave BR Saving	23.75%	7.48%	10.35%	1.05%	9.39%	7.98%	9.59%	1.51%	4.31%	0.49%	6.45%	2.83%	0.28%	4.38%	2.63%	8.64%
	All pictures	Ave PSNR gain	0.170	0.082	0.154	0.009	0.151	0.100	0.111	0.027	0.101	0.010	0.155	0.057	0.008	0.102	0.038	0.121
		Ave BR Saving	4.78%	1.91%	3.02%	0.25%	3.13%	2.30%	3.28%	0.61%	2.01%	0.29%	3.18%	1.24%	0.19%	2.20%	0.87%	2.63%