Block-matching translational and rotational motion compensated prediction using interpolated reference frame

Ng, Ka-Ho; Po, Lai-Man; Cheung, Kwok-Wai; Wong, Ka-Man

Research Article

Block-Matching Translational and Rotational Motion Compensated Prediction Using Interpolated Reference Frame

Ka-Ho Ng,[1] Lai-Man Po,[1] Kwok-Wai Cheung,[2] and Ka-Man Wong[1]

[1] *Department of Electronic Engineering, City University of Hong Kong, Hong Kong*
[2] *Department of Computer Science, Chu Hai College of Higher Education, Hong Kong*

Correspondence should be addressed to Ka-Ho Ng, kahomike@gmail.com

Motion compensated prediction (MCP) implemented in most video coding schemes is based on translational motion model. However, nontranslational motions, for example, rotational motions, are common in videos. Higher-order motion model researches try to enhance the prediction accuracy of MCP by modeling those nontranslational motions. However, they require affine parameter estimation, and most of them have very high computational complexity. In this paper, a translational and rotational MCP method using special subsampling in the interpolated frame is proposed. This method is simple to implement and has low computational complexity. Experimental results show that many blocks can be better predicted by the proposed method, and therefore a higher prediction quality can be achieved with acceptable overheads. We believe this approach opens a new direction in MCP research.

1. Introduction

Modern video coding schemes achieve high compression efficiency by exploring the temporary redundancy between frames via motion compensated prediction (MCP). In MCP, a block of pixels in reference frames is chosen as prediction candidate for the block in the current frame. Conventional MCP assumes objects moving along the imaging plane with translational motion, and most video coding standards implement MCP based on this classical translational motion model. Lots of research works are done to increase the efficiency of translational MCP, for example, in H.264/AVC [1], advanced block matching motion estimation algorithm is adopted [2]. Multiple reference frames (MRF) [3] is also adopted to provide additional candidates for prediction over a longer period of time. Another MCP technique is variable block size (VBS) [4] motion compensation. These techniques almost push the performance of translational motion-based MCP to the limit.

However, projection of real-world moving objects onto a 2D imaging plane will not always result in pure translational objects motion. Rotation, zoom [5], and other nonrigid motions are also pervasive in videos. Researches

on higher-order motion models such as affine [6, 7], bilinear [8], quadratic [9], perspective [10], and elastic [11] ones are conducted. These higher-order motion models aim to include nontranslational motions so that MCP prediction accuracy can be increased at the expense of additional motion vectors or parameters. However, these methods require motion parameter estimations. A commonly used method for motion parameter estimation is Gauss-Newton minimization algorithm in which motion parameters are iteratively updated until a minimum is found for the cost function [12]. Motion parameter estimation is in general of very high computational complexity. Moreover, subpixel reconstruction is required for these higher-order motion models because the transformed positions may not be a sampling point of the image. Interpolation is required to obtain the intensity values of these positions. This further increases the computational complexity. As a result, higher-order motion models are seldom used in practical coding applications.

In this paper a new translational and rotational MCP method is proposed. In this method, special subsampling in the interpolated reference frame effectively predicts the rotational block motions. In the next section, this method will be

TABLE 1: Rotation angle distribution in sequence *Foreman*.

Foreman ($\Delta\theta = 0.1°$)					
Angle range (in degree)	No. of blocks selected RMCP	% in total no. of Blocks	Angle range (in degree)	No. of blocks selected RMCP	% in total no. of blocks
$0.1 \sim 1$	3450	8.80%	$-0.1 \sim -1$	3357	8.56%
$1.1 \sim 2$	872	2.22%	$-1.1 \sim -2$	889	2.27%
$2.1 \sim 3$	468	1.19%	$-2.1 \sim -3$	462	1.18%
$3.1 \sim 4$	334	0.85%	$-3.1 \sim -4$	349	0.89%
$4.1 \sim 5$	170	0.43%	$-4.1 \sim -5$	148	0.38%
$5.1 \sim 6$	85	0.22%	$-5.1 \sim -6$	111	0.28%
$6.1 \sim 7$	93	0.24%	$-6.1 \sim -7$	126	0.32%
$7.1 \sim 8$	87	0.22%	$-7.1 \sim -8$	93	0.24%
$8.1 \sim 9$	29	0.07%	$-8.1 \sim -9$	32	0.08%
$9.1 \sim 10$	25	0.06%	$-9.1 \sim -10$	27	0.07%
$10.1 \sim 11$	21	0.05%	$-10.1 \sim -11$	21	0.05%
$11.1 \sim 12$	16	0.04%	$-11.1 \sim -12$	16	0.04%
$12.1 \sim 13$	10	0.03%	$-12.1 \sim -13$	11	0.03%
$13.1 \sim 14$	11	0.03%	$-13.1 \sim -14$	8	0.02%
$14.1 \sim 15$	6	0.02%	$-14.1 \sim -15$	11	0.03%
$15.1 \sim 16$	7	0.02%	$-15.1 \sim -16$	7	0.02%
$16.1 \sim 17$	5	0.01%	$-16.1 \sim -17$	3	0.01%
$17.1 \sim 18$	5	0.01%	$-17.1 \sim -18$	1	0.00%
$18.1 \sim 19$	2	0.01%	$-18.1 \sim -19$	3	0.01%
$19.1 \sim 20$	5	0.01%	$-19.1 \sim -20$	2	0.01%
$20.1 \sim 21$	4	0.01%	$-20.1 \sim -21$	1	0.00%
$21.1 \sim 22$	2	0.01%	$-21.1 \sim -22$	6	0.02%
$22.1 \sim 23$	1	0.00%	$-22.1 \sim -23$	6	0.02%
$23.1 \sim 24$	1	0.00%	$-23.1 \sim -24$	1	0.00%
$24.1 \sim 25$	0	0.00%	$-24.1 \sim -25$	2	0.01%
$25.1 \sim 26$	1	0.00%	$-25.1 \sim -26$	0	0.00%
$26.1 \sim 27$	4	0.01%	$-26.1 \sim -27$	2	0.01%
$27.1 \sim 28$	0	0.00%	$27.1 \sim -28$	0	0.00%
$28.1 \sim 29$	1	0.00%	$-28.1 \sim -29$	2	0.01%
$29.1 \sim 30$	7	0.02%	$-29.1 \sim -30$	6	0.02%
$30.1 \sim 30.6$	9	0.02%	$-30.1 \sim -30.6$	8	0.02%
	Sum:			11442	29.19%

discussed in detail. In Section 3, experimental results will be provided and discussed. In the last section, a conclusion will be given.

2. Translational and Rotational Motion Compensated Prediction

In theory, rotational motions are common in videos, for example, the rotation of car wheel and waving of hand, and so forth. Moreover, many complex motions can be modeled by translational and rotational motion combined. Figure 1 gives an example of block matching using translational and rotational motion model. In this example, a man is waving his arm and the best matched block is a rotated block in the reference frame. Figure 2 shows the rotation representation

in our motion model. As our motion model combines both translational and rotational motion, the motion vector (MV) representation is MV (x, y, θ), where x and y represent the translational displacement as in traditional MCP, θ is the rotation angle (in degree) of the block matching.

2.1. Rotated Subsampling in Interpolated Reference Frame. Fractional-pixel accuracy MCP [13], which is adopted in the latest video coding standards, performs block matching on interpolated reference frames. For example, 1/4-pixel and 1/8-pixel MCP require 4 and 8 times interpolated reference frames respectively. In [5, 14], zoom MCP is implemented by subsampling in the interpolated reference frame. Similarly, to implement the translational and rotational MCP, pixels in the interpolated reference frames are subsampled in ways such

TABLE 2: Rotation angle distribution in sequence *Stefan*.

Stefan ($\Delta\theta = 0.1°$)					
Angle range (in degree)	No. of blocks selected RMCP	% in total no. of Blocks	Angle range (in degree)	No. of blocks selected RMCP	% in total no. of blocks
$0.1 \sim 1$	5785	14.76%	$-0.1 \sim -1$	5648	14.41%
$1.1 \sim 2$	605	1.54%	$-1.1 \sim -2$	625	1.59%
$2.1 \sim 3$	257	0.66%	$-2.1 \sim -3$	282	0.72%
$3.1 \sim 4$	239	0.61%	$-3.1 \sim -4$	233	0.59%
$4.1 \sim 5$	81	0.21%	$-4.1 \sim -5$	82	0.21%
$5.1 \sim 6$	84	0.21%	$-5.1 \sim -6$	71	0.18%
$6.1 \sim 7$	69	0.18%	$-6.1 \sim -7$	63	0.16%
$7.1 \sim 8$	52	0.13%	$-7.1 \sim -8$	58	0.15%
$8.1 \sim 9$	25	0.06%	$-8.1 \sim -9$	28	0.07%
$9.1 \sim 10$	22	0.06%	$-9.1 \sim -10$	29	0.07%
$10.1 \sim 11$	13	0.03%	$-10.1 \sim -11$	25	0.06%
$11.1 \sim 12$	6	0.02%	$-11.1 \sim -12$	18	0.05%
$12.1 \sim 13$	4	0.01%	$-12.1 \sim -13$	6	0.02%
$13.1 \sim 14$	2	0.01%	$-13.1 \sim -14$	13	0.03%
$14.1 \sim 15$	4	0.01%	$-14.1 \sim -15$	5	0.01%
$15.1 \sim 16$	10	0.03%	$-15.1 \sim -16$	4	0.01%
$16.1 \sim 17$	0	0.00%	$-16.1 \sim -17$	1	0.00%
$17.1 \sim 18$	2	0.01%	$-17.1 \sim -18$	4	0.01%
$18.1 \sim 19$	2	0.01%	$-18.1 \sim -19$	2	0.01%
$19.1 \sim 20$	1	0.00%	$-19.1 \sim -20$	3	0.01%
$20.1 \sim 21$	0	0.00%	$-20.1 \sim -21$	3	0.01%
$21.1 \sim 22$	1	0.00%	$-21.1 \sim -22$	2	0.01%
$22.1 \sim 23$	0	0.00%	$-22.1 \sim -23$	1	0.00%
$23.1 \sim 24$	3	0.01%	$-23.1 \sim -24$	4	0.01%
$24.1 \sim 25$	0	0.00%	$-24.1 \sim -25$	1	0.00%
$25.1 \sim 26$	1	0.00%	$-25.1 \sim -26$	1	0.00%
$26.1 \sim 27$	1	0.00%	$-26.1 \sim -27$	1	0.00%
$27.1 \sim 28$	0	0.00%	$27.1 \sim -28$	5	0.01%
$28.1 \sim 29$	0	0.00%	$-28.1 \sim -29$	0	0.00%
$29.1 \sim 30$	0	0.00%	$-29.1 \sim -30$	3	0.01%
$30.1 \sim 30.6$	6	0.02%	$-30.1 \sim -30.6$	6	0.02%
	Sum:			14502	36.99%

that the composite reference blocks are the rotated versions of the reference block. Block matching is then performed between these rotated reference blocks and the current picture block. The one with the lowest distortion will be selected. Figure 3 shows an example of the proposed MCP. The best match is the one with $(+1, -2)$ displacement and then with a $-20°$ rotation. The coordinates of the subsampled pixels can be obtained by the rotation transformation equations

$$x_2 = \cos\theta(x_1 - x_0) - \sin\theta(y_1 - y_0) + x_0,$$
$$y_2 = \sin\theta(x_1 - x_0) + \cos\theta(y_1 - y_0) + y_0, \quad (1)$$

where θ is the rotation angle, and x_0 and y_0 are the coordinates of the center of rotation, x_1 and y_1 are the original coordinates, x_2 and y_2 are the rotation coordinates. The rotated coordinates will be rounded into integers.

2.2. Computational Complexity Determined by Search Angle Range and Interval. Interpolated reference frames exist in all codecs which implement fractional-pixel accuracy MCP. The rotation coordinates can be precomputed. Therefore the only computation complexity increase in our proposed method is the additional number of block matching between the rotated blocks and the current block. Another overhead will be the number of bits required to code the best matched rotation angle. However, if we control the number of rotation angles to be used, we can use a few bits to represent the angles and at the same time reduce the computation complexity required. Experiments show that up to 37% of the blocks can be better predicted with rotational MCP. The angle of rotation is usually small, especially for static video sequences like *Akiyo*. This is reasonable as objects between frames

TABLE 3: Rotation angle distribution in sequence *Akiyo*.

| Akiyo ($\Delta\theta = 0.1°$)					
Angle range (in degree)	No. of blocks selected RMCP	% in total no. of Blocks	Angle range (in degree)	No. of blocks selected RMCP	% in total no. of blocks
$0.1 \sim 1$	849	2.17%	$-0.1 \sim -1$	918	2.34%
$1.1 \sim 2$	97	0.25%	$-1.1 \sim -2$	114	0.29%
$2.1 \sim 3$	26	0.07%	$-2.1 \sim -3$	26	0.07%
$3.1 \sim 4$	15	0.04%	$-3.1 \sim -4$	16	0.04%
$4.1 \sim 5$	4	0.01%	$-4.1 \sim -5$	4	0.01%
$5.1 \sim 6$	2	0.01%	$-5.1 \sim -6$	1	0.00%
$6.1 \sim 7$	1	0.00%	$-6.1 \sim -7$	0	0.00%
$7.1 \sim 8$	2	0.01%	$-7.1 \sim -8$	3	0.01%
$8.1 \sim 9$	2	0.01%	$-8.1 \sim -9$	0	0.00%
$9.1 \sim 10$	2	0.01%	$-9.1 \sim -10$	2	0.01%
$10.1 \sim 11$	0	0.00%	$-10.1 \sim -11$	0	0.00%
$11.1 \sim 12$	1	0.00%	$-11.1 \sim -12$	0	0.00%
$12.1 \sim 13$	1	0.00%	$-12.1 \sim -13$	0	0.00%
$13.1 \sim 30.6$	0	0.00%	$-13.1 \sim -30.6$	0	0.00%
	Sum:			2086	5.32%
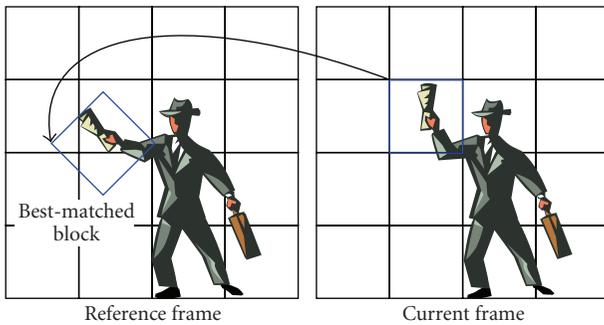
FIGURE 1: An example of block matching using translational and rotational motion model.
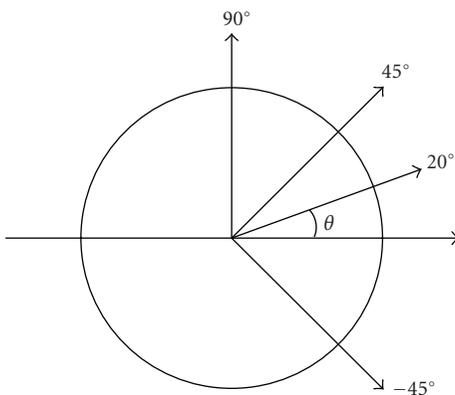

FIGURE 2: Rotation representation.

usually will not rotate a lot. We define search angle interval and search angle range so that the number of block matching between the rotated blocks and the current block is fixed. For a search angle interval $\Delta\theta$, block matching will be performed for each block-rotation of $n\Delta\theta$, $n \in Z$, starting from 0° within a search angle range. For example, if the search angle interval is 0.1° and the search range is ±5°, a total number of $10°/0.1° = 100$ rotated block matching will be performed. For practical implementation, we choose a lower number of rotational searches. For example the performance of using 16 searches in both clockwise and anticlockwise directions, that is, 32 in total, is investigated in detail. Only 5 bits are required to represent the 32 rotational angles. For applications requiring lower computational complexity, we can use even fewer rotational searches. With fixed number of searches, the search angle interval and range can be varied. For example, search angle interval 0.1° with search angle range ±1.6° and search angle interval 0.2° with search angle range ±3.2° both have 32 rotational searches. By comparing the prediction accuracy of different search angle intervals with the same total number of searches, we found that using a larger angle interval performs better for complex motion sequences, for example, *Foreman*. For small motion sequences, for example, *Akiyo*, using a smaller interval angle has better prediction quality. This is logical because complex motion sequences contain blocks with rotation of larger extent. With the same number of searches, using larger angle interval covers larger rotation. On the other hand, static sequences contain blocks with very small rotation. For these sequences a smaller angle of interval is more suitable. In this stage we have not yet found a search angle interval and range which is extremely robust. In the next section we will show the experimental results of some typical values we tested.

The proposed MCP method can be summarized below:

Step 1. Find the best translational motion vector (MV) in integer-pixel accuracy using traditional integer motion estimation
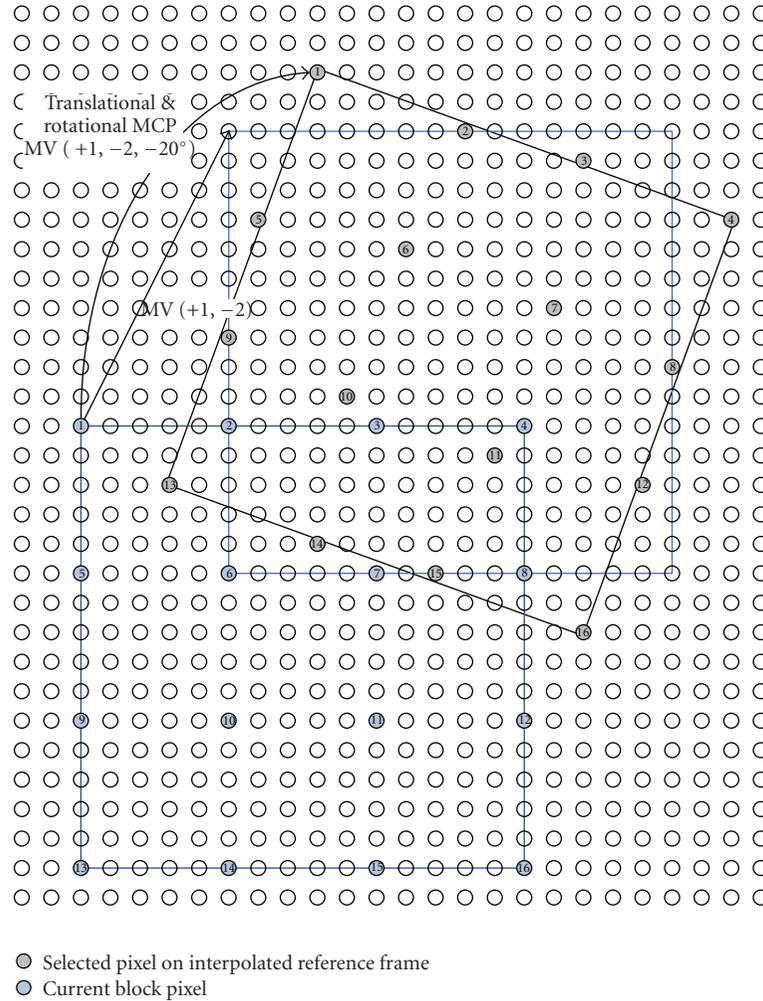
FIGURE 3: An example of translational and rotational MCP implemented by special subsampling on the interpolated reference frame.

Step 2. At the position pointed by the best translational MV and each of its surrounding fractional-pixel accuracy positions, performs original (nonrotated) block matching and rotated block matchings using the special subsampling method.

Step 3. The position and the rotation angle which has the lowest distortion is returned. This is the translational and rotational MV (x, y, θ) which will be encoded and transmitted to the decoder side.

3. Experimental Results

Experiments using CIF sequences *Soccer*, *Stefan*, *Crew*, *Foreman*, *Mobile*, and *Akiyo* are performed to analyze the performance of the proposed translational and rotational MCP. The block size is 16×16 pixels. The search window size is ± 16 pixels. Integer-pixel motion estimation is performed using exhaustive search (full search) algorithm, which searches each integer position in the search window.

Because the coordinates of the subsampled pixels will eventually be rounded into integers, some of the $\Delta\theta$ will be too small such that the rounded coordinates will be the same as that of the previous angle. To avoid repeated calculations, we will skip those angles in our experiments.

3.1. Prevalence of Rotational Motion. To analyze the rotation angle distribution, 600 differently rotated blocks are searched, with half of them rotated clockwise and the other half rotated anticlockwise. The angle of interval is $0.1°$ and the rotation angles without effects are skipped. Table 1 shows the percentages of rotated blocks selected (in total number of blocks) in sequence *Foreman*. To reduce table size, the rotation angles are grouped in ranges of one degree. That is, the percentages of blocks rotated $0.1°, 0.2°, \ldots$, to $1°$ are summed up and tabulated in the 1st data row, the percentages of blocks rotated $1.1°, 1.2°, \ldots$, to $2°$ are summed up and tabulated in the 2nd data row, and so on. From Table 1, it can be observed 29.19% of the blocks can be better predicted using rotational MCP. Clockwise and anticlockwise
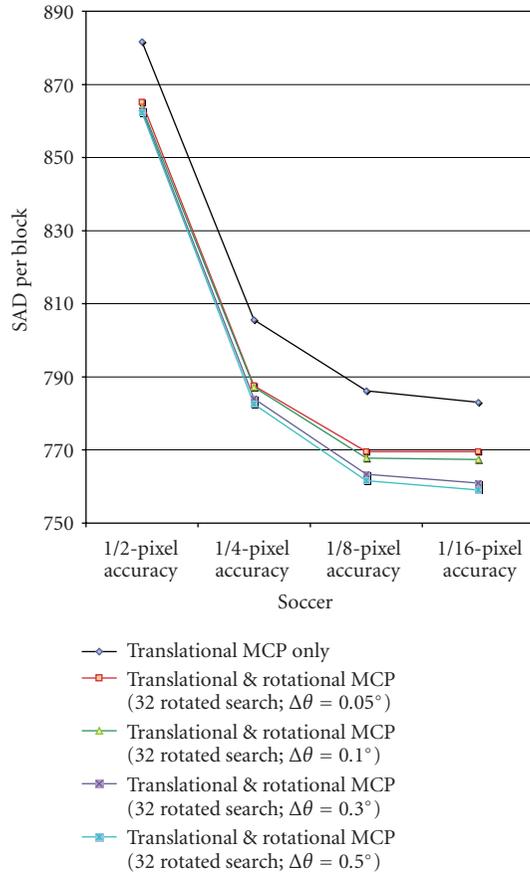
Figure 4: Average SAD per block in sequence *Soccer*.
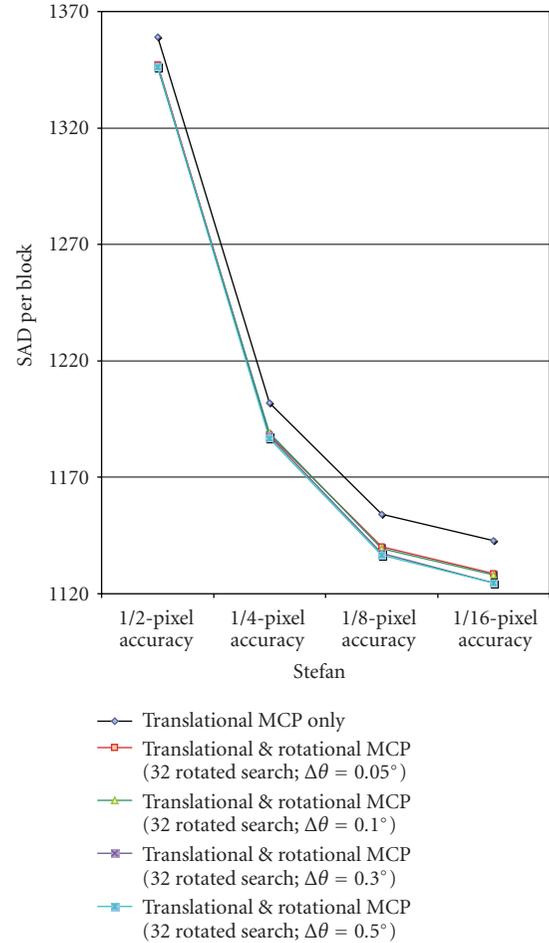
Figure 5: Average SAD per block in sequence *Stefan*.

rotations have similar distributions. That means the blocks are not biased to a particular rotation direction. In addition, most of the rotated blocks have small angles of rotation. 24% of the blocks have rotation angles within $\pm 3°$. Table 2 shows the results in sequence *Stefan*. 36.99% of the blocks are better predicted using rotational MCP and 33.68% of the blocks have rotation angles within $\pm 3°$. Table 3 shows the results in sequence *Akiyo*. There are fewer rotation blocks selected. 5.32% of the blocks are better predicted by rotational MCP. There is no rotation block with clockwise rotation greater than 13° selected. In anticlockwise rotation, there is also no block selected with rotation angle greater than $-10°$. From these results we can see that rotation properties vary among sequences with different motion contents.

3.2. Optimum Search Angle Range and Interval. Figures 4, 5, 6, 7, 8, and 9 compare the average SAD per block achieved using translational-only MCP and the proposed MCP with 32 rotated searches and angles of interval 0.05°, 0.1°, 0.3°, and 0.5°, under the conditions of 1/2, 1/4, 1/8, and 1/16-pixel accuracy MCP. The lower the average SAD per block, the better the prediction quality. We can see that in 1/2-pixel accuracy, the improvement by rotational MCP is lower. It is because in MCP of 1/2-pixel accuracy, the reference frame is interpolated only two times. The rounding problem

of the coordinates of the rotated pixel is severe such that a small change in rotation angle does not always have effect. With higher fractional-pixel accuracy, reference frames with higher resolutions will be available. The improvement in prediction quality by rotational MCP can be fully reflected. The increase of rotation angle interval from 0.05° to 0.5° can improve the prediction quality for sequences *Soccer*, *Foreman*, *Crew* and *Stefan*. It is because if the rotation angle interval is increased, with the same 32 rotated searches a larger rotation search range can be covered. The prediction quality can be improved because these sequences contain large and complex motions. These motions can be better represented by MVs with larger rotational angles. Further increase of the rotation angle interval, for example, 0.7° or 0.9°, will decrease the prediction quality. It is because from the statistical results (Tables 1 to 3) we can see that high percentage of the rotated blocks have relatively small rotation angles. With fixed number of searches, larger angle interval will miss these blocks and thus have lower prediction quality.

For static motion sequence *Akiyo*, the prediction quality cannot be improved with larger rotation angle interval. On the contrary, the quality is slightly dropped because in static sequences blocks actually rotate very slightly. In

TABLE 4: Comparison of computational complexity and prediction accuracy.

|	Complexity		*Soccer*		*Stefan*		*Crew*		*Foreman*		*Mobile*		*Akiyo*	
	No. of 16×16 SAD calculations after Interger ME		Average PSNR per frame (dB)		Average PSNR per frame (dB)		Average PSNR per frame (dB)		Average PSNR per frame (dB)		Average PSNR per frame (dB)		Average PSNR per frame (dB)	
	Transla-tional-only MCP	TRM CP (4 rotated search-es, $\Delta\theta = 2.0°$)	Transla-tional-only MCP	TRM CP (4 rotated search-es, $\Delta\theta = 2.0°$)	Transla-tional-only MCP	TRM CP (4 rotated search-es, $\Delta\theta = 2.0°$)	Transla-tional-only MCP	TRM CP (4 rotated search-es, $\Delta\theta = 2.0°$)	Transla-tional-only MCP	TRM CP (4 rotated search-es, $\Delta\theta = 2.0°$)	Transla-tional-only MCP	TRM CP (4 rotated search-es, $\Delta\theta = 2.0°$)	Transla-tional-only MCP	TRM CP (4 rotated search-es, $\Delta\theta = 2.0°$)
1/2-pixel accuracy	8	44	30.16	30.38	26.9	26.99	32.93	33.06	34.25	34.36	26.16	26.25	43.92	43.98
1/4-pixel accuracy	48	*244*	30.51	*30.78*	27.46	27.58	33.49	*33.66*	34.75	*34.91*	27.32	27.49	44.91	44.98
1/8-pixel accuracy	*224*	**1124**	*30.58*	**30.87**	27.6	**27.72**	*33.61*	**33.79**	*34.85*	**35.03**	27.58	**27.78**	45.35	45.43
1/16-pixel accuracy	**960**	4804	**30.59**	30.88	**27.63**	27.75	**33.65**	33.82	**34.87**	35.05	**27.63**	27.84	45.47	45.56
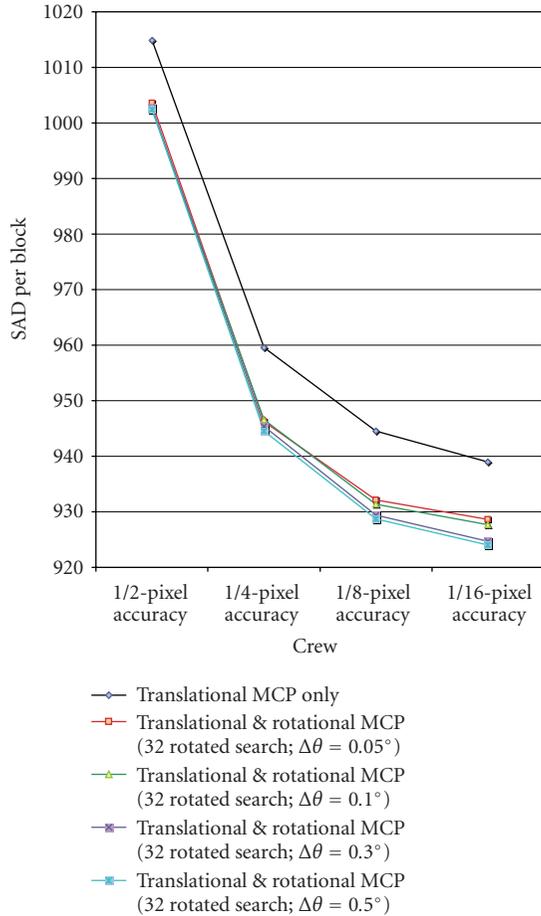
FIGURE 6: Average SAD per block in sequence *Crew*.

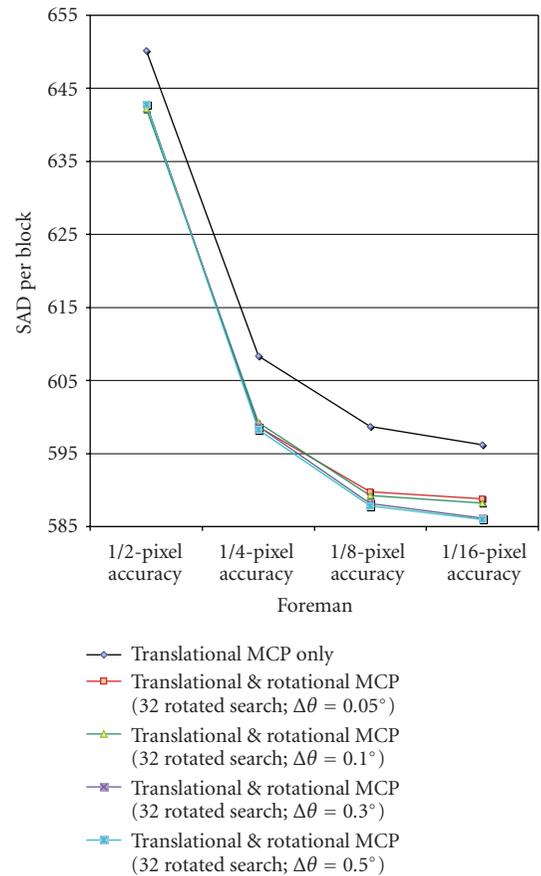
FIGURE 7: Average SAD per block in sequence *Foreman*.
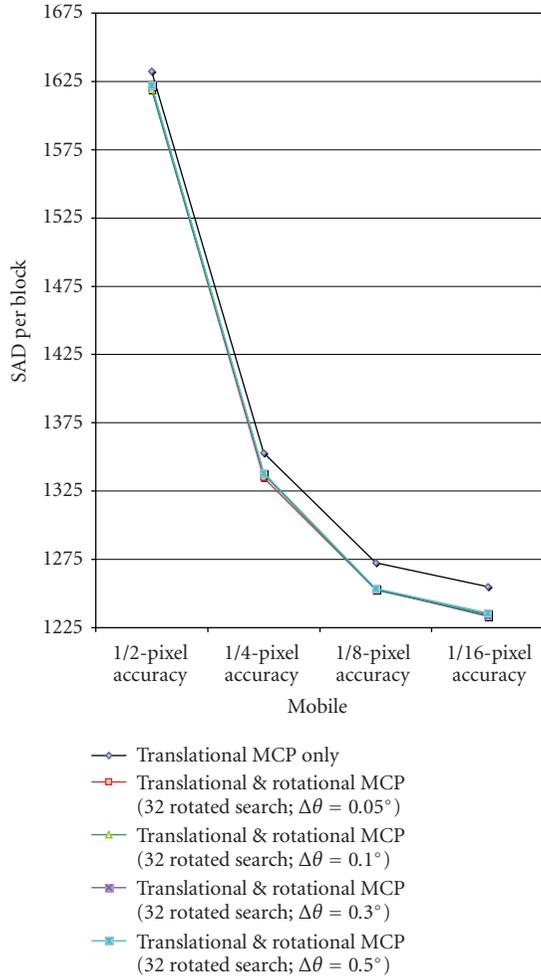
Figure 8: Average SAD per block in sequence *Mobile*.

Figure 9: Average SAD per block in sequence *Akiyo*.

sequence *Mobile*, using a smaller rotation angle interval also has slightly better performance.

3.3. Computational Complexity of Proposed MCP Method. To estimate the computational complexity of the proposed MCP method in a practical system, we measured the peak signal-to-noise ration (PSNR) achieved with four rotated searches and angle of interval $2.0°$. The computational complexity of four rotated searches with certain fractional-pixel accuracy is similar to that of translational-only MCP with the next higher fractional-pixel accuracy. For example the number of SAD calculation of the proposed MCP method with four rotated searches at 1/4-pixel accuracy is 244 because at each 1/4-pixel position four rotated and one original matching are performed. The number of SAD calculation of translational-only MCP with 1/8-pixel accuracy is 224 because there are 224 search candidate positions. These two numbers are comparable which means their computational complexities are similar. If the proposed method with certain fractional-pixel accuracy can achieve better prediction accuracy than the traditional MCP with higher fractional-pixel accuracy, it
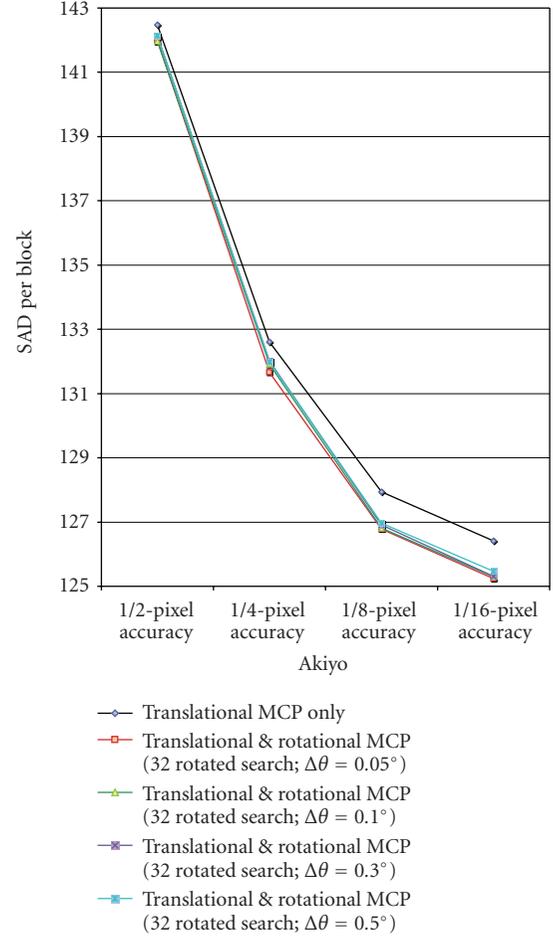
has great potential to replace traditional MCP with higher fractional-pixel accuracy.

Table 4 shows the results. TRMCP is the proposed translational and rotational MCP. In all test sequences except *Akiyo*, TRMCP with 1/8-pixel accuracy has better prediction accuracy in terms of PSNR than translational MCP with 1/16-pixel accuracy. In *Soccer*, *Crew*, and *Foreman*, TRMCP with 1/4-pixel accuracy has better prediction accuracy than translational MCP with 1/8-pixel. In 1/2-pixel accuracy, the proposed method cannot perform better than translational MCP with 1/4-pixel. It is because in 1/2-pixel accuracy the rounding problem of the coordinates of the rotated pixel is severe, as mentioned in the previous subsection. The results also show that the proposed method works better in sequences with complex motions, for example, *Soccer* and *Foreman*. This matches with our assumption that complex motions can be better modeled by the proposed translational and rotational motion model.

4. Conclusion

In this paper, translational and rotational MCP implemented by special subsampling in the interpolated frame is proposed.

It is found that up to 37% of the blocks can be better predicted with rotational MCP. The proposed method has the merits of easy implementation and low overhead. The interpolated frame used by rotational MCP is the same as that used by fractional-pixel accuracy MCP, which exists in most recent video coding standards. Experimental results show that higher fractional-pixel accuracies, for example, 1/16-pixel, cannot much further improve the prediction accuracy in translational MCP. Moreover, they require the additional computation overhead of extra interpolation calculation. With regard to the side information overhead, MCP with higher fractional-pixel accuracy needs more bits to transmit the higher fractional-pixel accuracy MV. For example the number of candidate search positions of 1/16-pixel accuracy MCP is around four times that of 1/8-pixel accuracy MCP. Our proposed method only needs to transmit one rotational angle parameter. For example four rotational angles can be represented by 2 bits, and so on. The increase in side information overhead is negligible.

In view of the decreasing effectiveness of MCP with higher fractional-pixel accuracies, the proposed method shows a new research direction to further improve the performance of MCP. Further works in this direction include the determination of an optimized search angle interval and range which are robust for video sequences of different motion contents. Furthermore, the correlation between the translational MV and the rotation angle is also under investigation.

Acknowledgment

References

[1] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.

[2] A. M. Tourapis, "Enhanced predictive zonal search for single and multiple frame motion estimation," in *Viual Communications and Image Processing*, vol. 4671 of *Proceedings of SPIE*, pp. 1069–1079, San Jose, Calif, USA, 2002.

[3] T. Wiegand, X. Zhang, and B. Girod, "Long-term memory motion-compensated prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 1, pp. 70–84, 1999.

[4] G. J. Sullivan and R. L. Baker, "Rate-distortion optimized motion compensation for video compression using fixed or variable size blocks," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '91)*, pp. 85–90, Phoenix, Ariz, USA, December 1991.

[5] K.-M. Wong, L.-M. Po, K.-W. Cheung, and K.-H. Ng, "Block-matching translation and zoom motion-compensated prediction by sub-sampling," in *Proceedings of IEEE International Conference on Image Processing (ICIP '09)*, pp. 1597–1600, Cairo, Egypt, November 2009.

[6] T. Wiegand, E. Steinbach, and B. Girod, "Affine multipicture motion-compensated prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 2, pp. 197–209, 2005.

[7] R. C. Kordasiewicz, M. D. Gallant, and S. Shirani, "Affine motion prediction based on translational motion vectors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 10, pp. 1388–1394, 2007.

[8] G. J. Sullivan and R. L. Baker, "Motion compensation for video compression using control grid interpolation," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '91)*, vol. 4, pp. 2713–2716, Toronto, Canada, April 1991.

[9] M. Karczewicz, J. Nieweglowski, and P. Haavisto, "Video coding using motion compensation with polynomial motion vector fields," *Signal Processing: Image Communication*, vol. 10, no. 1-3, pp. 63–91, 1997.

[10] Y. Nakaya and H. Harashima, "Motion compensation based on spatial transformations," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 4, no. 3, pp. 339–356, 1994.

[11] M. R. Pickering, M. R. Frater, and J. F. Arnold, "Enhanced motion compensation using elastic image registration," in *Proceedings of IEEE International Conference on Image Processing*, pp. 1061–1064, October 2006.

[12] B. Zitová and J. Flusser, "Image registration methods: a survey," *Image and Vision Computing*, vol. 21, no. 11, pp. 977–1000, 2003.

[13] B. Girod, "Motion-compensating prediction with fractional-pel accuracy," *IEEE Transactions on Communications*, vol. 41, no. 4, pp. 604–612, 1993.

[14] L.-M. Po, K.-M. Wong, K.-H. Ng et al., "Motion compensated prediction by subsampled block matching for zoom motion contents," in *ISO/IEC JTC1/SC29/WG11 MPEG2010/ M17163, (91th MPEG Meeting)*, Kyoto, Japan, 2010.