A Fast Motion Estimation Algorithm Based on Motion Vector Distribution Prediction

Yuan Gao
Beijing University of Technology, Beijing, China
Email: yuangaoyg001@sina.com

Peng-yu Liu and Ke-bin Jia
Beijing University of Technology, Beijing, China
Email: liupengyu@bjut.edu.cn and jiakebin@bjut.edu.cn

***Abstract*—the main factor that impacts the implementation and application of the latest video coding standard H.264/AVC is the rapid decline encoding speed which is caused by the cost of increased computational complexity. Reducing time-consuming of motion estimation is the key to improve encoding efficiency. In this paper, according to UMHexagonS algorithm, a method of motion vector distribution prediction is proposed and combines with designed patterns to achieve adaptive sub-regional searching. Simulation results show that the proposed motion estimation scheme achieves reducing 21.40% motion estimation encoding time with a good rate-distortion performance compared with UMHexagonS algorithm in JM18.4. The proposed algorithm improves the performance of real-time encoding.**

***Index Terms*—H.264/AVC, Motion Estimation, UMHexagonS algorithm, Motion Vector Distribution Prediction**

I. Introduction

A series of advanced technologies, such as variable-value block motion compensation, fractional pixel accuracy motion estimation, multi-reference frames motion compensation, multi-mode inter predictions and in-loop de-blocking filter, are adopted by H.264/AVC. H.264/AVC is proposed formally by the International Telecommunication Union (ITU-T) Video Coding Experts Group and the International Organization for Standardization (ISO/IEC) Moving Picture Experts Group [1]. The above features make H.264/AVC achieve high video image quality and low bit rate encoding advantages. Compared with other video coding standards, H.264/AVC achieves higher compression performance and more reliable transmission performance. At the same time, the computational complexity leads to high encoding time consuming, which seriously impacts on the real-time performance of H.264/AVC. With the increasing numbers of reference frames, the proportion of computational complexity is growing [2]. As one of the significant technologies, motion estimation (ME) consumes as much as 60%-80% computation of whole encoding process. Therefore, H.264/AVC has been dedicated to the study of efficient motion estimation algorithm in order to reduce the encoding time and improve the encoding efficiency.

Motion estimation is based on block-matching technique. Full search (FS) algorithm calculates all the search points for 7 type macro-blocks, so it costs a huge amount of time. Early fast ME algorithms, such as three-step (TSS) algorithm, four-step (FSS) algorithm and hexagonal (HEXBS) algorithm have too many search points and are easy to trap in local minima when their search patterns do not match with motion accurately.

In recent years, a lot of advanced motion estimation search algorithms have been proposed. Su [3] raised an improved cross diamond motion search algorithm which adopted many technologies including prediction of initial search point, suitable search pattern choosing and auxiliary search points to reduce search points. Chen [4] promoted the performance of motion estimation with CUDA. He applied global motion search algorithm to avoid loss of image quality and parallel computing capacity of graphics processors to accelerate the encoding process. Nguyen [5] optimized motion estimation with hardware accelerator. He proposed a coarse-grained dynamically reconfigurable computing system to develop the real-time speed. With the further study of temporal and spatial correlation and human visual characteristics, some new algorithms have made a progress. The typical algorithm is Unsymmetrical-Cross Multi-Hexagon Search (UMHexagonS) algorithm [6]. UMHexagonS algorithm is regarded as a representative for the hybrid search algorithms and adopted formally in H.264/AVC reference software JM7.6 [7]. Compared with full search algorithm, UMHexagonS algorithm claims that it can reduce 90% of motion estimation time. In order to make the initial search

Corresponding author: GAO Yuan.
E-mail:yuangaoyg001@sina.com

doi:10.4304/jsw.8.11.2863-2870

point close to the best prediction point, UMHexagonS algorithm provides 4 type predictions. Although UMHexagonS algorithm overcomes the above shortcomings, it still has calculation redundancy. In recent year, some proposed algorithms [8-10] based on UMHexagonS have been improved. Besides, a named New-UMHexagonS (NUMHexagonS) algorithm involves with the preliminary discussions on macro-block correlation and achieves a good effect on optimizing UMHexagonS algorithm. In spite of the above algorithms can reduce the consuming of motion estimation in varying degrees, the mining and use of motion characteristics in macro-block still need further research.

This paper proposes a novel motion estimation search algorithm. It makes use of motion vector distribution characteristics to decrease the search range, and designs new search patterns based on motion features to achieve selecting search regions adaptively. The proposed algorithm enhances the performance of UMHexagonS algorithm remarkably on the condition that the proposed algorithm maintains a low bit rate and high video image quality.

The remainder of this paper is organized as follows. In section 2, some related works about UMHexagonS algorithm and NUMHexagonS algorithm are introduced. Section 3 analyses the motion vector distribution characteristics. Section 4 describes the method of predicting motion vector distribution and the search strategy with new search patterns. Simulation results are given to verify the effectiveness of the proposed algorithm in section 5. Section 6 gives the conclusion.

II. Overview Of Related Algorithms

UMHexagonS algorithm provides many predictions, such as the Median Prediction, the Up-Layer Prediction, the Corresponding-block Prediction and the Neighboring Reference frame Prediction. UMHexagonS searching strategy begins with cursory search pattern, then turns to elaborate search patterns. With multi-patterns, UMHexagonS algorithm gets rid of the disadvantage that the traditional fast algorithms are easy to trap in local minima. In addition, adaptive early termination threshold makes UMHexagonS algorithm more efficient. To sum up, UMHexagonS algorithm improves the effectiveness and the robustness of prediction greatly.

Although there are different kinds of predictions, UMHexagonS algorithm does not combine patterns search with motion vector characteristics, which causes a lot of unnecessary search points during search process. Because of non-uniform distributed motion vectors at each search step, there is no need to traverse all search points to determine the best matched point. The large search pattern does not take the motion characteristics of the current macro-block into consideration. Point-by-point blind searching contributes little to improve the accuracy of motion estimation, while consuming lots of encoding time. On the basis of the above features, Liu et al. did a large number of researches and proposed NUMHexagonS algorithm [11]. NUMHexagonS algorithm gets the following three ways to optimize:

(1) Based on the feature that layers of search points are progressively decreasing by search radius decreasing designs a new uneven multi-hexagon-grid search pattern;

(2) Based on macro-block motion intensity adaptively selects the number of layers in uneven multi-hexagon-grid search pattern;

(3) Based on macro-block motion intensity adaptively selects whether perform 5 × 5 full search.

NUMHexagonS algorithm search process is shown in Fig. 1. Compared with UMHexagonS algorithm, it can reduce a lot of motion estimation search points. It is very rough that NUMHexagonS algorithm combines motion vector characteristics with search strategy. NUMHexagonS algorithm only carries out the macro-block motion intensity. It is not precise to predict motion vector. This paper will make better use of motion vector characteristics based on NUMHexagonS algorithm. The statistic of motion vectors distribution will be figured out, and depend on statistical results to develop a more precise search strategy. Next part will mainly analyze motion vector distribution characteristics.

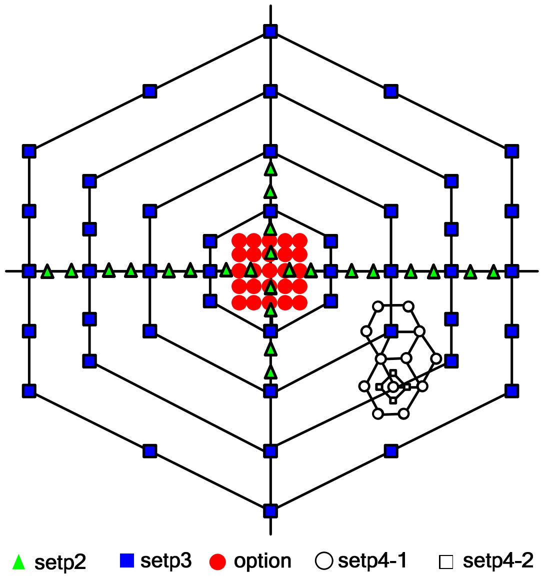


Figure 1. Search process and search pattern of NUMHexagonS

III. Motion Vector Distribution Analyze

In UMHexagonS algorithm, uneven multi-hexagon-grid search has a big search range and lots of search points which costs too much search time. Therefore, it is necessary to research the feature of motion vector distribution.

In order to acquire the statistic of motion vector distribution, search window is divided by the boundary of octagon as shown in Fig. 2. The central region indicates no motion, and it means motion vector is 0. Other regions indicate the positions where the best matched points appear. The regions are defined as the following formulas (1):

$$\begin{cases} central\ region & Origin \\ (-22.5°, 22.5°] \cup (157.5°, 202.5°] & Range1 \\ (22.5°, 67.5°] \cup (-157.5°, -112.5°] & Range2 \\ (67.5°, 112.5°] \cup (-112.5°, -67.5°] & Range3 \\ (112.5°, 157.5°] \cup (-67.5°, -22.5°] & Range4 \end{cases} \quad (1)$$

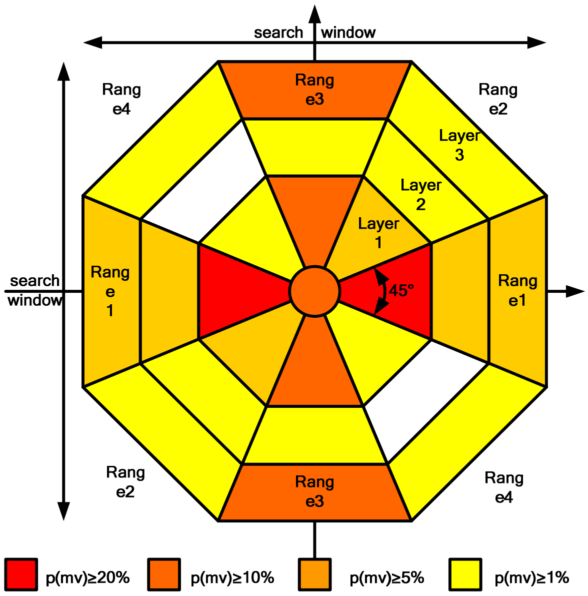


Figure 2. MV distribution probability

In formula (1) each Range is divided into Layer 1, Layer 2 and Layer 3 from inside to outside thereby constitute 19 regions. Take JM18.4 of H.264/AVC software as the experimental platform and select QCIF format (176 × 144) standard test sequences to figure out the best matched point probability that appears in each region of the uneven multi-hexagon-grid search. The statistical results are shown in TABLE I.

According to TABLE I, video sequence harbour with low motion, the best matched points mainly appear in Origin, Layer 1-Range 1, and Layer 1-Range 3 which indicates that the majority of macro-block motion vectors concentrate in the origin. Video sequence football with high motion, the best matched points mainly appear in Layer 3-Range 1 and Layer 3-Range 3 which indicates that the majority of macro-block motion vectors concentrate in the boundary. Video sequence bus whose foreground motion is low and background motion is high, the best matched points appear intensively in Layer 1-Range 1, Layer 1-Range 3 and Layer 3-Range 1, which indicates that motion vector distribution is dispersed. The above analyses show that there is an intrinsic link between the motion vector distribution and the motion estimation search position.

In order to further research the characteristics of the motion vector distribution, figure out the averages of the numbers which are every sequence in the same region in TABLE I and then use different colors for plotting in Fig. 2. The colors varied from dark to light represent the motion vector distribution probabilities varied from higher to lower. The statistical results show that the motion vector distribution meets the following characteristics:

(1) Motion vector distribution in horizontal and vertical are more than that in other directions, and the distribution probability in horizontal is higher than that in vertical;

(2) A part of video sequences exist a large number of motion vectors whose value are 0;

(3) Near the origin or the boundary of search window appears higher probability of motion vectors.

TABLE I
MV DISTRIBUTION PROBABILITIES OF DIFFERENT VIDEO SEQUENCES

Region of MV distribution / Video sequence name	football	bus	city	crew	harbour	ice	mobile
Origin (%)	16.84	6.11	6.01	13.67	42.90	20.58	1.97
Layer1-Range1 (%)	5.09	36.18	29.34	8.81	16.51	13.64	39.38
Layer1-Range2 (%)	3.82	7.17	13.74	5.18	2.27	5.21	10.04
Layer1-Range3 (%)	9.36	13.74	26.86	23.47	21.78	10.56	18.02
Layer1-Range4 (%)	1.09	3.49	6.84	2.89	0.66	2.38	5.23
Layer2-Range1 (%)	2.08	9.49	3.43	2.56	3.99	4.42	9.68
Layer2-Range2 (%)	1.24	1.90	1.72	1.39	0.56	1.76	2.58
Layer2-Range3 (%)	2.78	3.14	3.6	5.76	5.71	3.92	4.22
Layer2-Range4 (%)	0.45	0.93	0.85	0.77	0.18	0.86	1.24
Layer3-Range1 (%)	18.44	10.31	2.88	9.27	2.18	14.97	4.40
Layer3-Range2 (%)	8.61	1.91	1.07	3.94	0.23	4.95	1.05
Layer3-Range3 (%)	25.96	4.72	3.16	20.13	2.95	14.45	1.78
Layer3-Range4 (%)	4.23	0.91	0.52	2.15	0.08	2.30	0.42

Characteristic (1) complies with the phenomena that most of video sequence motions exist in horizontal more than in vertical. Characteristic (2) and characteristic (3) indicate that the video sequence with low motion macro-block appears a high probability of motion vector distribution in the origin or near the origin, and the video sequence with high motion macro-block appears a high probability of motion vector distribution at the boundary of search window. According to the above characteristics, different search strategies cannot achieve the purpose of accurate search which only distinguish the video sequences simply by low motion or high motion. Therefore it is necessary to refine search patterns based on motion vector distribution, so that it can make motion estimation process more accurately and reduce the numbers of search points further to raise search efficiency.

IV. The Proposed Algorithm

A. Principle

Based on motion vector distribution characteristics, the proposed algorithm makes NUMHexagonS algorithm further optimized. The patterns of NUMHexagonS algorithm are divided into different regions. In addiction the proposed algorithm can adaptively select the search regions by motion vector distribution prediction, thereby reduce the numbers of search points.

B. Design Patterns

The unsymmetrical cross search and the uneven multi-hexagon-grid search belong to cursory search processes. The motion vector distribution prediction can pinpoint the search position without wide search range. So it is necessary to divide original patterns into different regions and draw up new search strategies.

The unsymmetrical cross search pattern of NUMHexagonS algorithm is divided into four Groups as shown in Fig. 3. There are 8 search points in horizontal and 4 search points in vertical. The modified unsymmetrical cross search pattern complies with the requirement of the characteristic (1) in section III. The matched point is searched in one of the four Groups determining by motion vector distribution prediction. The search points of modified unsymmetrical cross search pattern are compressed to 1/3 of the original pattern approximately.

The uneven multi-hexagon-grid search pattern of NUMHexagonS algorithm is divided into 32 regions as shown in Fig. 3. To comply with the requirement of the characteristic (3) in Section III, the search points are distributed unevenly. More search points are distributed closer to the origin and the search boundary. In order to comply with the requirement of the characteristic (1) in Section III, there are 62 total search points in horizontal, ± 45° direction and 22 total search points in vertical, ±45° direction. The matched point is searched in some of the 32 regions determining by motion vector distribution prediction. The search points of modified uneven multi-hexagon-grid search pattern are compressed to 1/10 of the original pattern approximately.

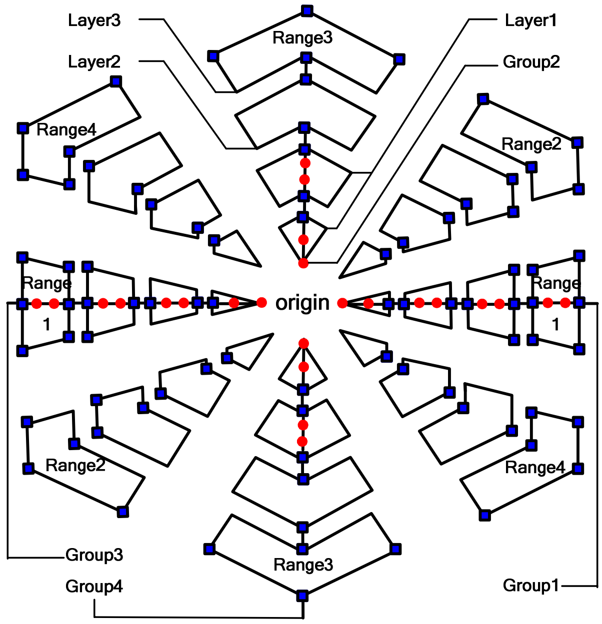


Figure 3. Modified search pattern

During the search process, the search patterns will be skipped as soon as the prediction value is 0. This search strategy complies with the requirement of the characteristic (2) in Section III. A large number of experiments prove that the modified patterns not only maintain motion estimation accuracy, but avoid unnecessary search points, and then decrease the motion estimation encoding time.

C. Predict Motion Vector Distribution

To predict motion vector distribution, it is essential to calculate the value and the direction of motion vector.

Predicting the value of motion vector refers to method of Ref. [11]. This paper compares current macro-block motion vector to predicted motion vector threshold. Some parameters which are related to the predicted motion vector threshold are defined as follows: $(1+\gamma)\,pred_{mincost}$ represents the upper limit threshold of motion vector prediction; $(1+\delta)\,pred_{mincost}$ represents the lower limit threshold of motion vector prediction, in addition $pred_{mincost}$ represents the minimum $RD_{mincost}$ of predicted motion vector. According to Fig. 3, the value of motion vector will determine the motion vector distributing in Layer 1, Layer 2 or Layer 3. Define the following formula (2):

$$\begin{cases} RD_{mincost} \le (1+\gamma)\,pred_{mincost} \\ (1+\gamma)\,pred_{mincost} < RD_{mincost} < (1+\delta)\,pred_{mincost} \\ RD_{mincost} \ge (1+\delta)\,pred_{mincost} \end{cases} \quad (2)$$

In addiction $RD_{mincost}$ is the current macro-block motion vector minimum rate distortion cost; γ and δ are defined as follows:

$$\gamma = \frac{Bsize[blocktype]}{pred_{mincost}^{2}} - \alpha_1[blocktype] \quad (3)$$

$$\delta = \frac{Bsize[blocktype]}{pred_{mincost}^{2}} - \alpha_2[blocktype] \quad (4)$$

$$\alpha_1[blocktype] = [-0.23, -0.23, -0.23, -0.25, -0.27, -0.27, -0.28] \quad (5)$$

$$\alpha_2[blocktype] = [-2.39, -2.40, -2.40, -2.41, -2.45, -2.45, -2.48] \quad (6)$$

According to formula (2), the rules are made as follows: the value of current motion vector is equal to or less than the lower limit threshold, motion vector is distributed in Origin or Layer 1; the value of current motion vector is between the upper limit threshold and the lower limit threshold, motion vector is distributed in Layer 1 and Layer 2; the value of the current motion vector is equal to or greater than the upper limit threshold, motion vector is distributed in Layer 3. The method of predicting motion vector value is also used in other search patterns.

Predicting the direction of motion vector refers to method of Ref. [12]. Assuming the coordinate of motion vector is (MV_x, MV_y), and then the direction of motion vector can be described by Direction Vector, namely $\overline{MV}=(MV_x, MV_y)$, According to Fig. 3, the direction of motion vector will determine motion vector to distribute in Origin, Range 1, Range 2, Range 3, Range 4, Group 1, Group 2, Group 3 and Group 4. Define the parameter k which is related to the direction of motion vector as following formula (7). The corresponding search positions of $\overrightarrow{MV}$ is as shown in Fig. 4.

$$k = \left| \frac{MV_y}{MV_x} \right| \quad (7)$$

Predicting the value and the direction of motion vector can obtain motion vector distribution accurately. The motion vector distribution prediction will be the condition that the modified patterns select the search regions adaptively.

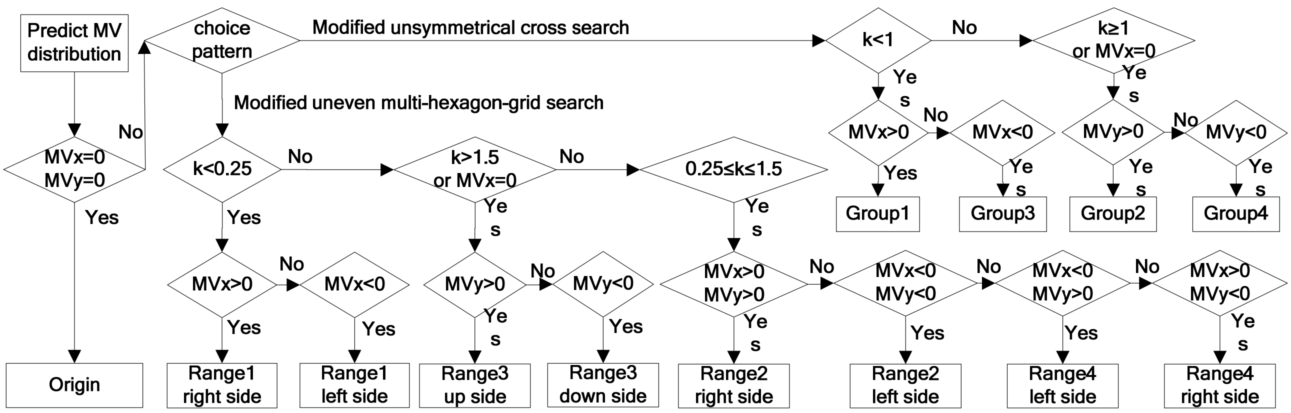


Figure 4. Flow chart of predicting the direction of motion vector

V. Performance Analysis

To further test the effectiveness of the proposed algorithm, simulation has been performed over different intensity of QCIF format (176 x 144) standard video sequences. Five sequences are crew, harbour, ice, mobile and city. Sequence city and harbour include slow-speed motion, sequence mobile and crew include middle-speed motion, and sequence ice includes high-speed motion.

Simulation presents the results of comparing the proposed algorithm with UMHexagonS algorithm and NUMHexagonS algorithm by ME-time Gain, Y-PSNR Gain, Bit-rate Gain and Average ME search points. The reference software version is JM 18.4. Each sequence contains 100 frames to be encoded. The frame rate is 30fps. Content Adaptive Variable Length Coding (CAVLC) entropy is used for simulation. The Group of Picture (GOP) is IPPP. The numbers of reference frames is configured for 5. Search range is 32 pixels. In order to evaluate the rate-distortion performance, the quantization parameter (QP) is 28, 32, 36 and 40 respectively. Simulation conditions are the same for all sequences.

Simulation results are shown in TABLE II and TABLE III. Compared with UMHexagonS algorithm, the proposed algorithm maintains the quality of reconstruction video, PSNR varies from +0.05% to -0.14%, Bit-rate varies from +1.54% to -0.1%. The proposed algorithm keeps H.264/AVC high compression ratio performance. It can reduce 15.97% to 28.52% of the motion estimation time compared with UMHexagonS algorithm and reduce 8.91% to 17.50% of the motion estimation time compared with NUMHexagonS algorithm as well. The simulation verifies the rationality of the modified patterns and the search point distribution. The search strategy based on motion vector distribution prediction is effective.

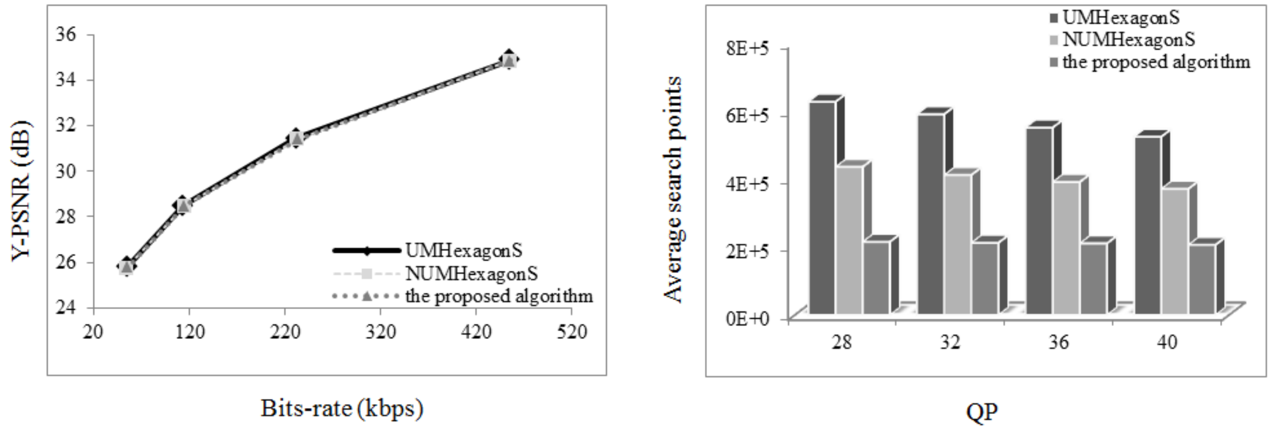
TABLE II
PERFORMANCE OF THE PROPOSED ALGORITHM COMPARED
(ME-TIME, Y-PSNR, BIT-RATE)

video sequence name	QP	UMHexagonS algorithm			NUMHexagonS algorithm			Proposed algorithm		
		ME-time (sec)	Y-PSNR (dB)	Bits-rate (kbps)	ME-time (sec)	Y-PSNR (dB)	Bits-rate (kbps)	ME-time (sec)	Y-PSNR (dB)	Bits-rate (kbps)
crew	28	126.25	37.16	362.97	109.68	37.16	364.09	93.61	37.14	368.01
	32	121.53	34.20	205.01	105.27	34.20	205.51	90.14	34.19	208.14
	36	115.78	31.60	114.68	99.81	31.55	115.05	84.80	31.58	115.49
	40	107.52	29.02	61.93	92.40	29.02	62.29	76.86	29.00	62.24
harbour	28	93.08	34.88	455.56	83.22	34.87	455.35	68.65	34.88	455.10
	32	90.35	31.44	232.16	81.62	31.42	232.74	68.12	31.43	233.67
	36	86.63	28.51	114.23	79.39	28.50	114.90	67.35	28.50	114.81
	40	83.10	25.79	55.31	76.44	25.76	55.12	65.48	25.78	55.47
ice	28	80.34	39.63	226.60	73.90	39.60	227.87	67.16	39.59	230.10
	32	80.48	36.41	147.98	73.39	36.45	148.51	66.38	36.42	150.21
	36	79.19	33.52	98.25	72.08	33.45	97.97	64.82	33.48	98.83
	40	75.76	30.62	63.65	68.29	30.69	63.81	62.21	30.60	64.03
mobile	28	83.57	34.60	641.78	74.63	34.59	643.92	65.83	34.59	645.70
	32	83.80	30.94	360.86	74.91	30.94	361.10	66.20	30.95	363.90
	36	84.40	27.80	199.74	75.41	27.80	200.72	66.42	27.78	199.97
	40	84.78	24.80	112.51	75.70	24.79	112.36	66.43	24.79	112.84
city	28	103.66	35.75	161.19	96.26	35.75	162.10	87.10	35.73	161.47
	32	105.80	32.67	86.49	96.98	32.66	86.63	87.18	32.68	86.95
	36	105.03	29.94	49.17	95.73	29.95	48.66	85.46	29.95	49.25
	40	98.54	27.35	29.37	90.10	27.36	29.22	79.54	27.34	29.39
Performance comparison with UMHexagonS & NUMHexagonS		ME-time Gain (%)	Y-PSNR Gain (%)	Bits-rate Gain (%)	ME-time Gain (%)	Y-PSNR Gain (%)	Bits-rate Gain (%)	**Annotation:** "-" means reduce "+"means increase		
		-21.40	-0.04	+0.63	-12.59	-0.02	+0.49			

Fig. 5 and Fig. 6 take harbour and ice as the typical video sequences to represent low motion and high motion respectively, and show the different results on the rate-distortion performance and the reduction of motion estimation search points under different QPs between the proposed algorithm, UMHexagonS algorithm and NUMHexagonS algorithm. According to rate-distortion functions, the proposed algorithm has an advantage to maintain low compression bit-rate and high reconstruction quality of H.264/AVC. The histograms of search point numbers show that the proposed algorithm reduces 55.75% and 35.60% search points respectively on average compared with UMHexagonS algorithm and NUMHexagonS algorithm, which improves the motion estimation real-time performance. The proposed algorithm has an obvious and stabile optimization, and reduces the complexity of motion estimation algorithm architecture effectively.

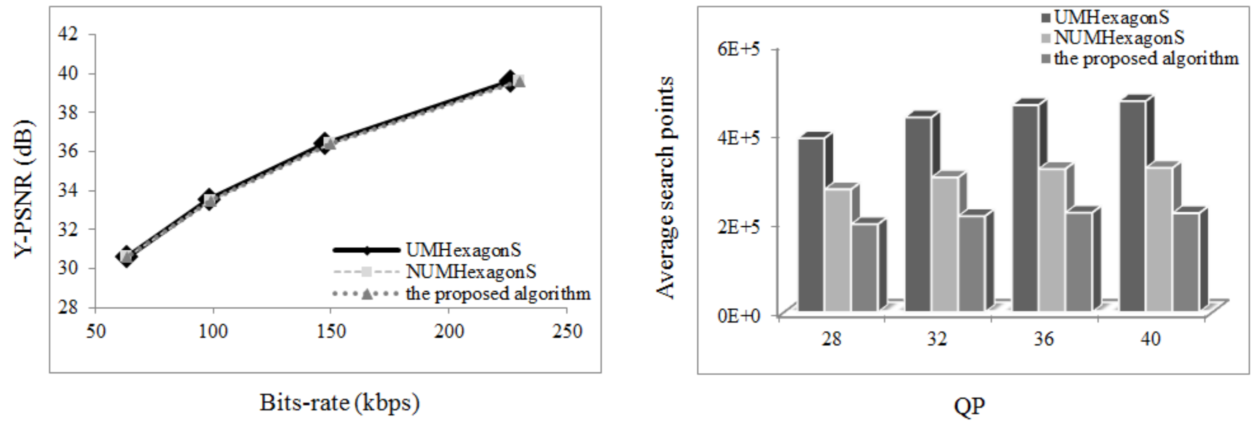
TABLE III
PERFORMANCE OF THE PROPOSED ALGORITHM COMPARED
(AVERAGE ME SEARCH POINTS)

video sequence name	QP	Average ME search points		
		UMHexagonS	NUMHexagonS	Proposed
crew	28	742687	498005	324906
	32	743806	489118	314239
	36	756629	484278	297774
	40	770473	475842	268478
harbour	28	627314	435678	215098
	32	590381	411549	212138
	36	551159	391170	210484
	40	524087	370948	206020
ice	28	388317	274319	196247
	32	435106	300862	214373
	36	462676	319701	222056
	40	471335	322059	221014
mobile	28	519607	351288	237723
	32	531464	361649	245956
	36	555376	379517	258505
	40	582498	399633	270534
city	28	546294	399629	284397
	32	601080	428045	302508
	36	639429	448423	311293
	40	627516	435181	290499
Performance comparison with UMHexagonS & NUMHexagonS		Average ME search points Gain (%)		Annotation: "-" means reduce "+"means increase
		-55.75%	-35.60%	



(a) Rate-distortion performance (b) ME search points

Figure 5. Performance between UMHexagonS, NUMHexagonS and the proposed algorithm (harbour)



(a) Rate-distortion performance (b) ME search points

Figure 6. Performance between UMHexagonS, NUMHexagonS and the proposed algorithm (ice)

VI. Conclusion

In this paper, on the basis of UMHexagonS algorithm and NUMHexagonS algorithm, the statistic of motion vector distribution was analyzed. We designed new search patterns and put forward the method of predicting motion vector distribution. The proposed algorithm made full use of motion vector characteristics, and combined motion vector distribution prediction with the new search patterns to make search region more accurate. The simulation results show that the proposed algorithm decreased by 15.97% to 28.52% of motion estimation time and 47.94% to 65.71% of motion estimation search points respectively compared with UMHexagonS algorithm (JM18.4), while maintaining the quality of the original structure and reconstruction bit streams. The proposed algorithm improves the performance of H.264/AVC real-time encoding effectively.

References

[1] Wiegand, Thomas, et al. "Overview of the H. 264/AVC video coding standard". *Circuits and Systems for Video Technology*, *IEEE Transactions on 13.7* (2003): 560-576.

[2] Tang, Qiang, and Panos Nasiopoulos. "Efficient motion re-estimation with rate-distortion optimization for MPEG-2 to H. 264/AVC transcoding". *Circuits and Systems for Video Technology*, *IEEE Transactions on 20.2* (2010): 262-274.

[3] Su, Xi, et al. "An Improved Cross Diamond Motion Search Algorithm Based on H. 264." *Journal of Computers* 6.12 (2011): 2603-2606.

[4] Chen, Zuo, Jialiang Ji, and Renfa Li. "Asynchronous parallel computing model of global motion estimation with CUDA." *Journal of Computers* 7.2 (2012): 341-348.

[5] Nguyen, Kiem-Hung, Peng Cao, and Xue-Xiang Wang. "An Efficient Implementation of H. 264/AVC Integer Motion Estimation Algorithm on Coarse-grained Reconfigurable Computing System." *Journal of Computers* 8.3 (2013): 594-604.

[6] Chen, Zhibo, et al. "Fast integer-pel and fractional-pel motion estimation for H. 264/AVC". *Journal of Visual Communication and Image Representation 17.2* (2006): 264-290.

[7] Xiao-Bin, Qiu, and Huang Chun-Qing. "An improved algorithm of fast motion estimation based on H. 264". *Computer Science and Education (ICCSE), 2010 5th International Conference on*. *IEEE*, 2010.

[8] Li, Ziyin, and Qi Yang. "A Fast Adaptive Motion Estimation Algorithm". *Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on*. Vol. 3. IEEE, 2012.

[9] Jambek, Asral Bahari, Yoong Yee Lai, and Arief Affendi Juri. "Design and analysis of fast search motion estimation architecture for video compression". *Intelligent and Advanced Systems (ICIAS), 2012 4th International Conference on*. *Vol. 1*. *IEEE*, 2012.

[10] YANG Hu, et al. Improvements on fast motion estimation strategy based on UMHexagonS for H. 264. *COMPUTER ENGINEERING AND DESIGN* 34.2 (2013): 550-555.

[11] Liu, Pengyu, and Kebin Jia. Low-Complexity Encoding Method for H. 264/AVC Based on Visual Perception. *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2011 Seventh International Conference on*. *IEEE*, 2011.

[12] Wei, Zhang, et al. "Directionality based fast fractional pel motion estimation for H. 264". *Systems Engineering and Electronics*, *Journal of 20.3* (2009): 457-462.

GAO Yuan was born in China, Beijing in 1990, graduated from Beijing University of Technology in 2013 and received a B.E. degree in Electronic Information and Engineering. He is currently pursuing the M.S. degree in circuit and system engineering. The major research field is video coding.

LIU Peng-yu was born in China, Huludao city of Liaoning Province in 1979, graduated from Beijing University of Technology in 2011 and received a doctor's degree in circuit and system engineering. The major research field is multimedia information processing.

Jia Ke-bin was born in China, Anyang city of Henan Province in 1962, graduated from the University of Science and Technology of China in 1990 and received a doctor's degree in signal and information processing. He is currently a Professor with the Beijing University of Technology. He is the dean of Electronic Information and Control Engineering, Beijing University of Technology. The major research field is signal processing.