International Journal of Multimedia and Ubiquitous Engineering
Vol.10, No.11 (2015), pp.343-358
http://dx.doi.org/10.14257/ijmue.2015.10.11.33

Improvement and Research on Efficient Motion Estimation Algorithm in Video Compression

Hong Chen[1] and Jia Ren[2]

[1]School of Electronic Information Engineering, Xi'an Technological University, Xi'an, 710021, China
[2]College of Information Sciences & Technology, Hainan University, Hainan, 570228, China
chenhong8008@126.com, wwwrenjia@gmail.com

Abstract

To improve and enhance the estimation speed and accuracy of motion estimation, this paper proposes a fast diamond search algorithm with improved search strategy; researches motion estimation algorithm by using pattern block matching; analyzes and set up fast searching algorithm model based on the fast convergence characteristics of three step search, the central distribution characteristics of new three step search and the approximate global optimal characteristics of diamond search; use the coarse positioning and fortified halfway stop technologies to establish improved diamond search algorithm; gives the method of reducing the number of search bits. Through experiment and analysis, the results show that the improved diamond search algorithm can effectively reduce the matching points and improve the search speed, and achieve a higher average peak signal to noise ratio.

Keywords: *Motion estimation, search speed, coarse positioning, fortified halfway stop operation*

1. Introduction

According to the statistics, more than 75% of information enters into the human brain through the eyes (in the form of images or videos). The vision is the most important way to obtain external information. The digitized video signals provide an important method to express the video signal which is as a visual information carrier effectively [1]. Undoubtedly, the digitized images or video information brings many advantages. As preserving and transmitting video must occupy lots of data channels and space, we can possibly transmit and preserve the video information effectively with the modern image and video compression technology.

Currently, motion estimation algorithm mainly refers to the block matching algorithm based on search pattern [2-3]. However, the computational complexity of full search to find the global optimal motion vector in this algorithm is enormous, especially when the search window is large, and this magnitude of calculation amount is unacceptable in real-time applications [4]. Therefore, fast algorithm aimed at such algorithms is proposed constantly. The purpose is to ensure the coding quality and to reduce the computational complexity as much as possible [5]. It is achieved by a faster way to find the global optimal motion vectors, or by a non-global optimal motion vector taking the place of global optimal motion vector. Approximated global optimal solution of motion vector is achieved by fast non-full search.

Fast block matching motion estimation algorithm based on search pattern is an important technology in video coding, which has been adopted as the main technique of inter-frame encoding by the various international video coding standards [6]. It also has been one of the research hot spots in the related fields, and many outstanding international

ISSN: 1975-0080 IJMUE

algorithms appear constantly in recent years [7]. Through the unremitting efforts of the scholars in different countries during the past years, these kinds of algorithms have reached a very high level. But the reality that the motion estimation module is the part with most wasteful resource and longest time in the video coding has not been changed. Furthermore, the motion estimation module is still the bottleneck of the high quality and complex real-time encoding. Aimed at the problems of improving the performance of motion estimation in speed and accuracy, you cannot eat the cake and have it too. These two indicators always cannot be achieved optimally simultaneously in the actual calculating process. Therefore, whether can we find the new breakthrough point to improve the search speed on the basis of ensuring the search quality has become the key of achieving a higher development in these kinds of algorithms. At the same time, how to resolve this tradeoff problem has a high theoretical and practical significance. It is a significant research topic in the field of image processing and image communication. The research and its application will promote the development of computer communication, image communication and multimedia technology.

2. Motion Estimation Algorithm Based on Pattern Block Matching

Block Matching Algorithm (BMA) is a very intuitive motion estimation algorithm, which is based on the mechanism of translational motion. Each point on the object has the same magnitude and direction of velocity during the translational motion process, so in the motion trajectory, the current position is obtained from offsetting of the previous position. In like manner, the content of the N-th frame is formed due to the translation in different directions of the corresponding parts in the N-1-th frame or N+1-th frame. Each frame is divided into two 16×16 pixel blocks. We assume that the pixels within each block move in the same translational direction. Then the best matched encoded block is looked for through certain matching criterion in a certain range around the adjacent frames[8]. Generally speaking, the best matching block and the relative displacement (dx, dy) of the current block are the Motion Vector (MV). While encoding the prediction image, we only need to transmit the Motion Vector (dx,dy), as well as the coding differences between the best matching block and the current block. At the receiving end, we find the best matching block of the current block in the adjacent frames which have been recovered through the motion vector, adding to the different blocks so as to recover the current block. The motion compensation process is like this. From the principle of BMA, we can see how to determine the two most important issues in the block matching method:

（1）The matching judgment standard of two sub-blocks.

（2）The search method with the minimum calculation value.

Different solutions to these two questions constitute the different search algorithms.

2.1. Block Matching Criterion

The matching criteria should not only reflect the degree of agreement between the estimated block of the current frame and the search block of the reference frame, but also require a small calculation value [9]. At present, some matching criteria functions are as follows:

$$MSD(dx, dy) = \frac{1}{mn} \sum_{i=-n/2}^{n/2} \sum_{j=-m/2}^{m/2} \left[F(i, j) - G(i+dx, j+dy)\right]^2 \tag{1}$$

$$MAD(dx, dy) = \frac{1}{mn} \sum_{i=-n/2}^{n/2} \sum_{j=-m/2}^{m/2} \left|F(i, j) - G(i+dx, j+dy)\right| \tag{2}$$

$$SAD(dx, dy) = \sum_{i=-n/2}^{n/2} \sum_{j=-m/2}^{m/2} \left| F(i, j) - G(i + dx, j + dy) \right| \quad (3)$$

$$CCF(dx, dy) = \frac{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} F(i, j) G(i + dx, j + dy)}{\sqrt{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} F^2(i, j)} \sqrt{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} G^2(i + dx, j + dy)}} \quad (4)$$

$$PDC(dx, dy) = \sum_i \sum_j T(dx, dy, i, j) \quad (5)$$

$$T(dx, dy, i, j) = \begin{cases} 1, \left| F(i, j) - G(i + dx, j + dy) \right| \le t \\ 0, otherwise \end{cases}$$ t is the predefined threshold

(6)

$$MME(dx, dy) = \min(\left| F(i, j) - G(i + dx, j + dy) \right|_{\max}, i = 1 \cdots n, j = 1 \cdots m) \quad (7)$$

Where, $F(i, j)$ is the current block, $G(i, j)$ is the reference block, (dx, dy) is the location of the motion vector.

2.2. Searching Mode Based on Pattern Block Matching

The problems of searching mode are how big the searching window is and what the searching path is. One way is a matching operation for the current blocks in all possible offsetting positions of Searching Window (SW). Then compare all the results to select the best matching block and its offsetting. This method is called full searching mode with high matching precision and higher time complexity of the algorithm. In order to reduce the algorithm complexity and improve the search speed, varieties of fast search algorithms appeared[10]. Classic three step search (TSS), new three step search (NTSS), four step search (FSS), two-dimensional logarithm (TDL), crossover method (CS) and diamond search (DS) and other methods are based on the block matching motion estimation fast search algorithm. A lager number of experiments show that these algorithms should be improved in the aspects of search performance or search speed while being applied in the practical environment.

3. Study of the Traditional Fast Search Algorithm

Researches show that the size and shape of search template affect not only the speed of motion estimation, but also the algorithm performance directly. This section will prove the above conclusion through the analysis of TSS, NTSS and DS, and provide theoretical support for the improved diamond search.

3.1. Three Step Search (TSS)

TSS algorithm has been paid much attention because of its characteristics of simple, robust and good performance. The basic idea of TSS algorithm uses a coarse-to-fine search mode, which starts from the original point, and tests the SAD values of eight adjacent points in the center and surrounding by the step of half of the maximum search length to find the SAD point with minimum value [11]. The second step is to place the minimum point in the center and halve the step size, looking for SAD minimum point in the center and around the square which has been narrowed, the rest can be concluded in this way, until step is 1. If the search length is 7 and the search precision is 1 pixel, and then the initial step is 4, and later reduces to 2 or 1. From above we can see that only three steps are needed.

A possible search process is as shown in Figure1. The points of [+4,+4], [+6,+4] is are the minimum block errors on the first and second step. The third step gets the final motion vector which is equal to [+7,+5]. The figures of each point show the candidate block positions of calculation at each search stage.

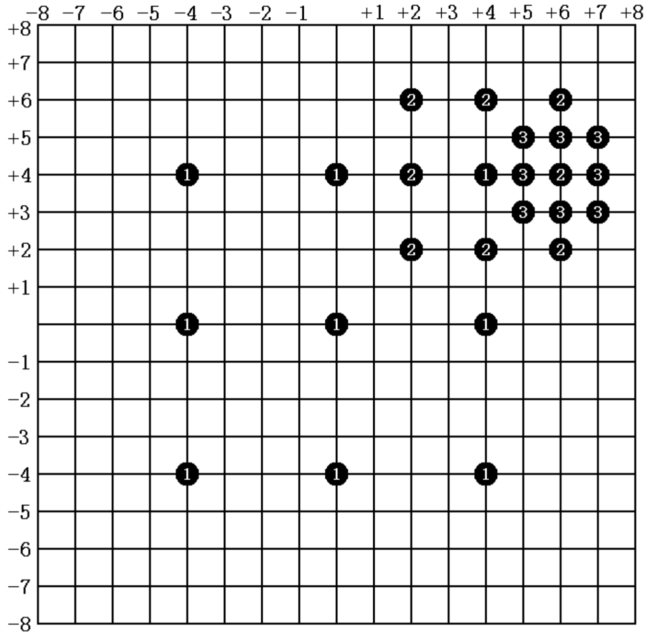

Figure 1. The TSS Search Process

In fact, when the search range is greater than 7, three steps is not enough. The general expression of search step is $\log_2(d_{max}+1)$, and the maximum search point is $1+8\log_2 W$, d_{max} is the maximum number of the displacement pixels of the current block relative to the reference. Generally speaking, the three-step is a typical fast search algorithm, so it is studied more, and then there have been many improved three-step to improve the performance on small motion.

3.2. New Three-Step Search (NTSS)

NTSS is another very classic fast algorithm. A large number of experiments show that with the distance between motion vector and center position increasing, the motion vector distribution in the probability will decrease rapidly. Therefore, the motion vector distribution in the center has the maximum probability for most video sequences. This is the characteristics of famous "center-bias" [12]. This idea makes people begin to take account of the statistical characteristics of the video sequence when they design fast algorithm [13]. At the same time the halfway stop technology used in the algorithm has been widely adopted in later algorithm.

As shown in Figure2, the center-biased characteristics are added on the NTSS based on TSS. The search procedure is as follow:

(1) We regard the initial position as the search center. [R/2] is the search radius to conduct a square. We look for the best matching (the round as shown in the figure) by searching the 8 points on the square and the central point, as well as the 8 points around the central point. If the best matching appears in search center, turn to (4). If the best matching appears in the 8 points around the central point, the point becomes a new search center and the search radius sets to 1. Otherwise, the same as TSS, update the search center and search radius.

(2) Construct a square with the search center and search radius. After matching the 8 points on the square, we find the best matching point(the square as shown in the figure);

(3) Set the best matching point to search center, and search radius is halved until the value of search radius is less than 1 (the triangle as shown in the figure);

(4)The current search center would correspond to the motion vector.

The search points of NTSS are dynamic. 17 is the fewest points (the best matching point is located at the initial position). On the worst-case scenario, the points is 8 more than TSS, namely, [9+8log2(R+1)]. Because of the center biased characteristics of video sequence, the probability of the worst-case scenario is very low. So the speed of NTSS is faster than TSS, To the search accuracy, NTSS is better than TSS because NTSS uses a smaller search mode near the center point.

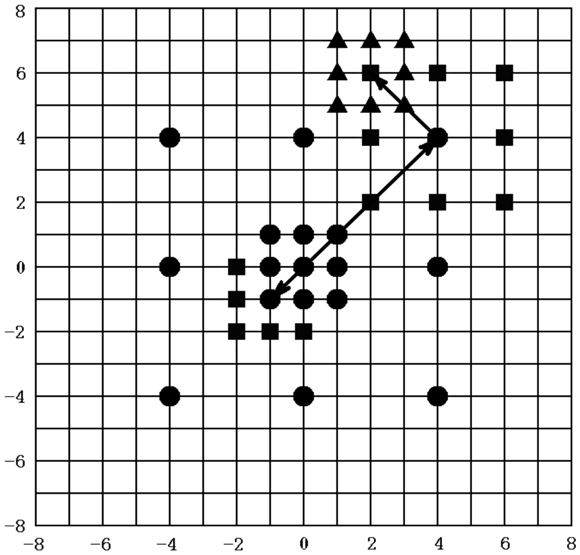

Figure 2. The NTSS Search Process

3.3. The Diamond Search Algorithm (DS)

DS algorithm has the characteristics of simple, robust, efficient. It is one of the fast search algorithms which have optimal performance, and is recommended by MPEG-4, as shown in Figure 4 and Figure5, DS algorithm uses two different search templates [14], one is a Large Diamond Search Pattern (LDSP) with 9 detection point, and the other is Small Diamond Search Pattern (SDSP) with 5 detection point. We use LDSP to calculate firstly. When Minimum Block Distortion (MBD) points appear in the center point, we transform LDSP into SDSP and then conduct the matching calculation [15], while MBD among the 5 points is the best matching point.

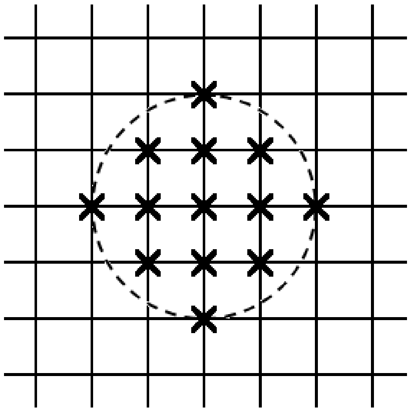

Figure 3. The Distribution of the Optimal Point

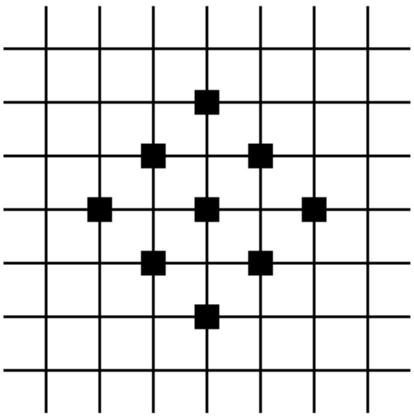

Figure 4. LDSP

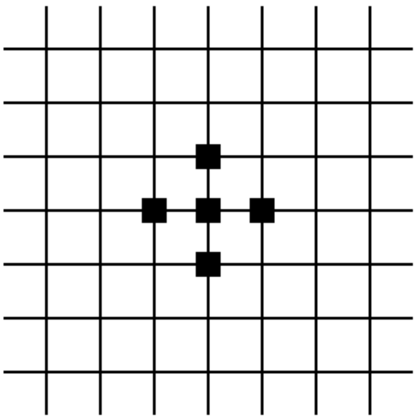

Figure 5. SDSP

Search procedure of DS motion estimation is as follow:

(1) Through the matching operation using the LDSP in the center of search area and around the 8 points, if MBD point is located in the center point. Then jump to (3), or jump to (2).

(2) We calculate with the new LDSP on the center point we found last time. If MBD point is located in the center point, jump to (3), otherwise repeat (2).

(3) We transform LDSP into SDSP to find MBD point by calculating 5 points. So the point location is the best vector.

Figure 6 shows an example which a DS algorithm is used to search the motion vector (-

4,-2). In this example, there are 5 steps. MBD points respectively are (-2,0),(-3,-1),(-4,-2) by using LDSP four times and SDSP once, totally 24 points are searched.

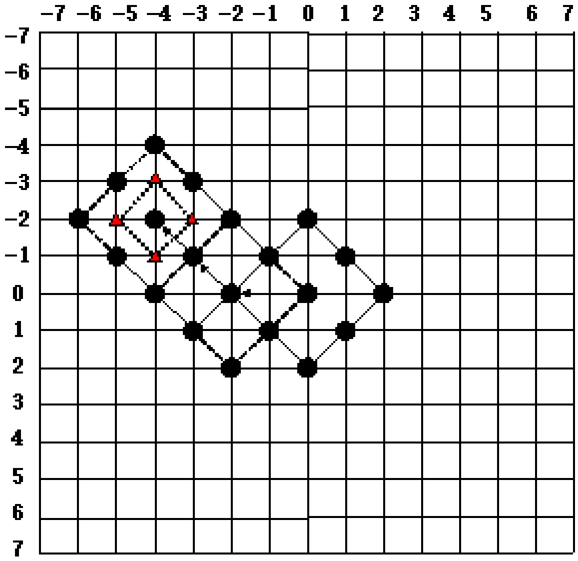

Figure 6. The DS Search Algorithm

The characteristic of DS algorithm is that it analyzes the basic laws of motion vectors in video image. It takes two types of templates which are LDSP and SDSP. We search with LDSP at first. Because of the greater steps and the wider search range, we can carry out the coarse positioning so that the search process will not fall into the local minimum value. After rough position, we can consider the optimal point on the diamond area enclosed in 8 points around LDSP, and then use the SDSP to do accurately positioning, so that the search is not hugely up and down [16-17]. So its performance is better than other algorithms. In addition, there is a strong correlation among the various steps of DS search. Therefore, only need to perform the matching calculations when the template moves, so that it can improve the search speed.

4. Improved Search Algorithm

To solve the problems of search performance and speed of the search algorithm, an Improved Diamond Search Algorithm (IDS) is proposed. The total computation in template matching is decided by the number of search and the computing of matching strategies itself. Based on guaranteeing the matching performance, it is extremely necessary to reduce the number of search. The objective function of search algorithm is following:

$$E(u,v) = \sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\left| f_{t-r}(i+u, j+v) - f_t(i,j)\right| \quad -W \le u,v \le W \tag{8}$$

(u,v) is the candidate motion vector, $f_{t-r}(\bullet,\bullet)$ and $f_t(\bullet,\bullet)$ are corresponding to the pixels of previous frame and current frame; W is the scale of search template in this function and it is equal to 7.

The DS algorithm can approximately achieve the best performance. One reason is that the iterative process of LDSP increases the possibility and the density of the search number. On the other hand, the core property of the motion field is considered: only when the optimal solution appears in the search template center point, the DS algorithm transforms from LDSP to SDSP, or it would end the process. But the speed of the algorithm is limited by the distance of the diamond template[18-19]. In order to improve the speed, we would adopt the idea of coarse-to-fine which is used in TTS algorithm to position direction approximately. For this reason, this paper puts forward to the IDS algorithm. And its basic principle is shown in Figure 7 and the specific description of the algorithm is as follows:

(1) Rough position of search direction

In the current frame three-step of square search template is used; we find the optimal solution of the object function method:

1) If the optimal solution occurs in the center of the search template, we choose 3×3 square search pattern near the center point and find the optimal solution, then the search terminates.

2) If the optimal solution occurs at the edge of the search template and the similarity is greater than a given threshold, we set the optimal solution point in the center of new search template, and search 3×3 square search pattern near center point, until find the optimal solution.

3) Otherwise, we transfer the center point of the search template to the optimal point, the coarse position end, jump to 2);

(2) The search template turns LDSP to find the optimal solution of objective function;

1) If the optimal solution appears in the center of the search template or at the edge of search range, jump to 3);

2) Otherwise, we replace the optimal solution with the center point of search template, and use LDSP again, then jump to 3);

(3) We transform search template to SDSP to find the optimal solution of objective function, the solution is MV we find, the search process ends.

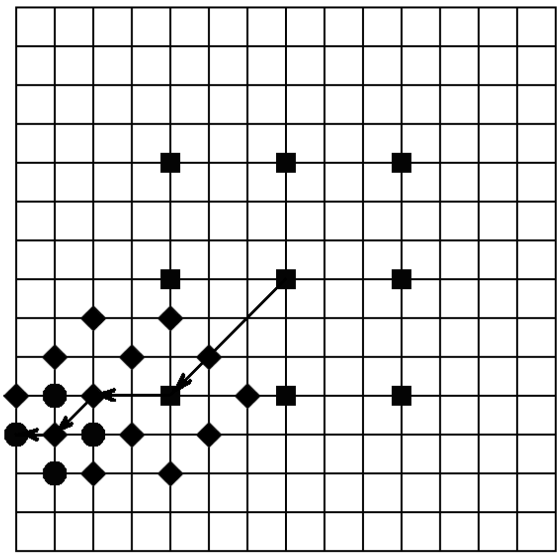

Figure 7. The IDS Search Process

Considering the characteristics of the fast convergence of TSS and the center distribution of NTSS, the paper sets the initial search template to the square. Differently, the search steps reduce to 3. The step of this template is between 2 in diamond search template and 4 in initial search steps of TSS. It can not only conduct coarse positioning for the diamond search, but also reduce the possibility of the error of search direction because of considering the center distribution of the motion field. If the optimal solution occurs in the center of search template, the possibility of the MV occurred near point is higher, so we can implement a fortified halfway stop operation (that is to get target by calculating the objective function again). If the optimal solution occurs at the edge of the search template and the similarity is greater than the given threshold (the critical point is set according to the actual needs and based on the center distribution of the motion vector field), as well as in accordance with the characteristic of center distribution[20]. The probability that MV occurred in the near point is higher, so we implement the operation of the halfway stop. Otherwise, we transfer the central point of search template to the point of the optimal solution, and the rough positioning ends. Research shows that the probability which MV occurred in the range of 5×5 of the center of the search point is higher. Thus to the new points, MV is likely less than 2. In order to avoid excessive search, this paper uses LDSP up to twice. Finally this process ends by using a complementary SDSP.

In summary, the characteristics of the center distribution of the motion field are considered. IDS algorithm is a good simulation for the state on the target region and ensure the search performance. At the same time, if some search bit can meet one of the two requirements: (1) when the similarity of some search bit near the central point is high; (2) the current optimal solution has been on the edge of search range, which means the current MV is larger than 7; Considering the characteristics, IDS adopts the fortified halfway operation and improves the search speed by timely stopping the search process. In addition, in order to improve the performance of motion estimation, IDS adopts the fortified halfway stop operation in 3×3 template twice while identifying the MV candidate region.

5. Experimental Results and Analysis

We conduct a series of experiments by selecting the typical test sequence with different characteristics with 352×288 resolution of Common Intermediate Format (CIF) and analyze the performance of IDS this paper put forward. The size of macro block selected by the experiment is 16×16 and the search range is 15×15, which means the pixel is 7.

5.1. Comparison of Average Matching Points with Different Algorithms

We conduct the experimental tests with different motion estimation algorithms based on the block matching in common standard video sequences. The fast algorithms include TSS, NTSS, FSS, BBGDS, DS, HEXBS, CDS and IDS proposed by this paper. We search the motion vector of every frame relative to previous frame in motion sequences. The compared performance parameters are as follows:

(1)MAD means the mean absolute difference of blocks between the whole sequence or frame and correspondingly prediction.

(2)MSE means the mean square sum of error of blocks between the whole sequence or frame and correspondingly prediction.

(3)NSP means the average matching points of each block during the motion estimation of the whole sequence or frame.

Table 1 lists the average performance parameters of the salesman, football and coastguard in various algorithms, respectively.

Table 1. Comparison of Average Performance Parameters with different Algorithms in Video Sequences

Salesman Sequences

	MAD	MSE	NSP
FS	2.749	18.634	225
TSS	2.793	19.323	25
NTSS	2.756	18.851	17.949
FSS	2.784	19.259	17.373
BBGDS	2.752	18.869	9.994
DS	2.782	19.191	13.793
HEXBS	2.817	19.768	11.434
CDS	2.765	18.972	9.767
IDS	2.764	19.047	8.451

Football Sequences

	MAD	MSE	NSP
FS	10.464	381.593	255
TSS	11.019	425.622	25
NTSS	10.823	414.121	22.314
FSS	10.984	427.372	19.676
BBGDS	11.125	446.173	15.284
DS	10.962	429.074	17.372
HEXBS	11.359	456.330	13.373
CDS	11.044	437.771	15.636
IDS	11.348	463.440	10.815

Coastguard Sequences

	MAD	MSE	NSP
FS	4.612	60.625	255
TSS	4.720	65.132	25
NTSS	4.656	62.446	21.292
FSS	4.683	64.752	19.798
BBGDS	4.702	65.105	14.542
DS	4.665	63.735	17.668
HEXBS	5.674	86.613	12.126
CDS	4.672	63.695	16.858
IDS	4.690	64.852	10.685

It can be seen clearly from the tables that no matter in which sequence, the search points of the IDS algorithm are greatly reduced. Take the coastguard sequences as an example, the speed of IDS has achieved 36.62% higher than CDS, which has a good comprehensive performance before.

5.2. Comparison of Saved Points

In this paper, we calculate the required matching points of the motion vector in some position searched by each algorithm in an ideal condition. The definition of the ideal condition is that the matching difference of blocks $Diff\ (P)$ is proportional to the Euclidean distance between the current matching point $P(x_p, y_p)$ and the location of the motion vector $mv(x_{mv}, y_{mv})$, namely,

$$Diff\ (P) = k \times ((x_p - x_{mv})^2 + (y_p - y_{mv})^2)^{\frac{1}{2}} \quad (9)$$

The proportionality constant k in the formula is taken as 1, in order to set the value of matching difference to the Euclidean distance between two points. We achieved the results in Table 2 from various fast algorithms of motion search applied in this ideal situation.

Table 2. Search Points of Motion Estimation Fast Algorithm in the Ideal Situation

NTSS

	0	1	2	3	4	5	6	7
0	18	21	21	34	34	34	34	34
1	20	22	22	34	34	34	34	34
2	20	22	22	22	34	34	34	34
3	34	34	22	34	34	34	34	34
4	34	34	34	34	34	34	34	34
5	34	34	34	34	34	34	34	34
6	34	34	34	34	34	34	34	34
7	34	34	34	34	34	34	34	34

DS

	0	1	2	3	4	5	6	7
0	13	13	18	18	23	23	27	27
1	13	16	16	21	21	26	26	27
2	18	16	19	19	24	24	28	28
3	18	21	19	22	22	27	27	28
4	23	21	24	22	25	25	29	29
5	23	26	24	27	25	28	28	29
6	27	26	28	27	29	28	29	29
7	27	27	28	28	29	29	29	27

IDS

	0	1	2	3	4	5	6	7
0	8	8	11	11	14	14	16	16
1	10	12	13	16	16	19	18	19
2	10	16	13	19	16	21	18	22
3	13	16	16	19	18	21	22	22
4	13	19	21	21	24	24	27	26
5	16	19	19	21	21	24	24	27
6	16	21	24	23	26	26	28	27
7	17	22	24	24	27	26	28	26

As can be seen from Table 2, in almost every position on the search window, the points searched by the proposed algorithm are less than the previous NTSS algorithm and DS algorithm. As described above, the total amount of computation of template matching are jointly determined by the amount of computation of the adoptive matching strategy itself and the number of search bits. Therefore, the proposed algorithm can effectively reduce the number of search bits, reduce the amount of computation, and improve search efficiency.

5.3. Comparison of Peak Signal To Noise Ratio (PSNR) with Different Algorithms

The average peak signal to noise ratio in the different typical test sequences is presented in Table 3 at different bit rates of different algorithms.

$$PSNR = 10 \lg(\frac{x_{max}^2}{MSE}) \quad (10)$$

x_{max} is The maximum gradation value in the original image

Table 3. Average PSNR Comparison of Proposed Algorithm and Other Algorithms

Test Sequences	QP	PSNR/dB				
		TSS	FSS	DS	NTSS	IDS
Akiyo	8	38.15	38.64	38.66	38.76	38.79
	24	32.79	32.93	32.82	32.96	33.12
Bus	12	30.42	30.65	30.52	30.54	30.58
	24	26.68	26.76	26.66	26.75	26.78
Coastguard	8	32.96	33.18	33.11	33.18	33.23
	24	27.53	27.59	27.47	27.62	27.66
Container	18	30.51	30.72	30.60	30.65	30.73
	24	29.07	29.27	29.17	29.26	29.29
Flower	8	32.42	32.76	32.67	32.03	32.82
	18	26.90	26.92	26.86	26.71	26.95
Football	18	30.40	30.47	30.46	30.60	30.61
	24	29.04	29.09	29.09	29.21	29.31
Foreman	8	34.53	34.79	34.75	34.78	34.89
	24	29.51	29.55	29.46	29.60	29.65
Hall_monitor	12	33.72	34.43	34.22	34.37	34.49
	18	31.74	32.24	32.07	32.24	32.33
Harbour	8	32.23	32.53	32.39	32.38	32.54
	12	29.85	30.00	29.88	29.94	30.02
Husky	18	25.15	25.19	25.14	25.00	25.27
	24	23.23	23.24	23.20	23.17	23.29
Ice	18	33.22	33.34	33.35	33.53	33.54
	24	31.86	31.94	31.94	32.09	32.12
Mobile	8	31.51	31.86	31.74	31.52	31.88
	18	26.37	26.48	26.37	26.37	26.55
News	12	33.99	34.39	34.29	34.39	34.48
	24	30.19	30.47	30.32	30.46	30.51
Soccer	18	31.11	31.17	31.13	31.25	31.27
	24	30.03	30.06	30.03	30.15	30.18

As can be seen from the table, compared with the sub-optimal algorithm at the middle bit rate (QP=8), the value of DSNR is 0.10dB higher than the maximum value (Foreman test sequence); compared with the sub-optimal algorithm under the low bit rate (QP=24), the value of DSNR is 0.16dB higher than the maximum value (Akiyo test sequence). At the same time, in a variety of different bit rates, the value of DSNR of the algorithm proposed in this paper is higher than that of other algorithms in all the test sequences, showing good reliability and stability.

5.4. Compensating Frame Results of Different Motion Estimation Fast Algorithm

In Figure 8, we select the Football sequence with a greater range of motion and a greater compensating difference of the different algorithms in video sequence. We also give the results of the 119th compensating frame after different motion estimation fast algorithms are employed.

a. The Original Image

b. FS Compensating Frame

c. NTSS Compensating Frame

d. DS Compensating Frame

e. IDS Compensating Frame

Figure 8. The 119th Compensating Frame of Football Sequence in different Algorithms

As shown in Figure 8, there is no difference between the compensating frame given by this paper and by the full search algorithms in subjective feelings. From the view that NTSS, DS, CDS and so on can achieve better subjective feelings. It is urgent need in current motion algorithm that the algorithm proposed in this paper can improve the search speed greatly.

6. Conclusions

We propose the Improved Diamond Search Algorithm (IDS) by adopting fast convergence of TSS, central distribution characteristics of NTSS and the approximate global optimal characteristics of DS algorithm, combining with the coarse positioning and the fortified halfway stop technology. The experimental results show that because of the fortified halfway stop technology, approximate 60% of the search process can be stopped at the first step, so the minimum search points (eight points) can be achieved in this part of the process. Therefore, the search efficiency is improved effectively. In a variety of bit rates, compared with other algorithms, IDS algorithm achieves a higher average peak signal to noise ratio in each test sequence. The computational complexity of this algorithm is significantly reduced on the basis of ensuring image quality and similar to the full search algorithm. The algorithm proposed in this paper only uses more effective templates to improve the search speed. And the obtained data can provide a reliable theoretical foundation for the subsequent research. If combined with other methods which could enhance the speed, such as the prediction of the search start point, the sampling of search lattice, *etc.*, the speed can be further improved on this basis, and eventually reach the practical and real-time level. The further work will continue to focus on these aspects.

Acknowledgments

This work is proposed in National Natural Science Foundation of China (No.61440048) and Natural Science Foundation of Hainan Province (No.614227).

References

[1] Y. Sun and K. Lee, “Video compression coding effects on motion detection”, Information technology, vol. 2, no. 3, **(2013)**, pp. 126-129.
[2] R. Mokry, “Minimal Error Drift in Frequency Scalability for Motion-Compensated DCT Coding”, IEEE Trans. on Circuits and System for Video Technology, vol. 4,no. 4, **(1994)**.
[3] K. Sauer and B. Schwartz, “Efficient block motion estimation using integral projections”, IEEE Trans. Circuits Syst. Video Technology, vol. 6, no. 5, **(1996)**, pp.513-518.
[4] L. H. Ye and L. M. Jun, “Motion Estimation Algorithms Based on H.264”, Journal of University of JiNan (Sci. &Tech.), vol. 23, no. 1, **(2009)**, pp. 72-75.
[5] Y. Baek, H. S. Oh, H. K. Lee, “An efficient block-matching criterion for motion estimation and its VLSI implementation”, IEEE Trans. Consumer Electronics, vol. 42, no. 4, **(1996)**, pp. 885-892.
[6] Z. J. Yi, L. X. Ling and H. Z. Gang, “Optimization of multi-mode motion estimation algorithm in H.264”, Journal of Beijing University of A aeronautics and A stronautics, vol. 34, no. 2, **(2008)**, pp. 140-143.
[7] X. Quan, L. J. Lin and W. Song, “Fast motion estimation with mode decision”, Journal of Zhejiang University (Engineering Science), vol. 39, no.6, **(2005)**, pp. 878-881.
[8] L. W. Sheng, L. Qin and D. L. Bo, “LBT Based Multi-Node Cooperative Image Compression Algorithm for WMSNs”, Chinese Journal of Sensors and Actuators, vol. 21, no. 9, **(2008)**, pp. 1600-1604.
[9] D. G. Guang and G. B. Long, “New Fast Motion Estimation Algorithm Based on Line Search”, vol. 38, no. 2, **(2004)**, pp. 136-139.
[10] H. Li, “Research on a new photoelectric detection method to anti-muzzle's flame or light disturbance and projectile's information recognition in photoelectric detection target”, Optoelectronics and advanced materials-rapid communications, vol. 8, no. 7-8, **(2014)**, pp. 653-658.
[11] Y. Peng, W. Hua and Y. S. Qiang, “Fast motion estimation algorithm for H.264”, J Tsinghua Univ. (Sci. & Tech.), vol. 45, no. 4, **(2005)**, pp. 501-504.
[12] C. Chun, Y. Zhi, B. J. Jun and M. L. Jian, “A Fast Variable -block –size Motion Estimation Algorithm for H. 264”, Journal of Image and Graphics, vol. 12, no. 2, **(2007)**, pp. 272-276.
[13] G. Li, G. S. Rong and C. Z. Ming, “A Fast Global Motion Estimation Algorithm”, Journal of Image and Graphics, vol. 12, no. 12, **(2007)**, pp. 2019-2026.
[14] Li H. S. and Lei Z. H. Y., “Measurement of Projectile Burst Coordinates by Using Multi-Screen Optical Method and Its Error Analysis”, ACTA OPTICA SINICA, vol. 32, no. 1, **(2012)**, pp. 02120031-6.
[15] W. Tao, “Study of Some Key Problems in Design and Implementation of Video Codec”, Dissertation of Shanghai Jiao Tong University, **(2008)**.
[16] L. H. Feng, “Study of the Fast Block Matching Algorithms for Motion Estimation. Dissertation of master degree of Xi'an Electronic and Science University, **(2002)**.

[17] D. Tao, R. M. Tian and L. W. Jiang, "Improvement and Optimization of Content-based Arithmetic Coding in MPEG-4", Computer Engineering, vol. 32, no. 8, **(2006)**, pp. 229-231.
[18] H. S. Li, Z. M. Wang and Z. Y. Lei, "Analysis and calculation object detection capture rate in multi-sky-screens across measurement system", vol. 25, no. 1, **(2014)**, pp. 4369-4373.
[19] Y. Wang, J. Ostermann and Y. Q. Zhang, "Video processing and communications", Tsinghua University Press, **(2003)**.
[20] Lanterman A. D., Sullivan J. A. O. and Miller M. I., "Kullback-leibler distances for quantifying clutter and models". Optical engineering, vol. 38, no. 2, **(1999)**, pp. 2134-2146.

Authors

Hong Chen, was graduated from Harbin University of Science and Engineering. She is now a lecturer in School of Electronic Information Engineering, Xi'an Technological University. Her main research area is video encoding.

Jia Ren, is with the Department of Information Science & Technology, Hainan University, Haikou, China (E-mail: renjia@hainu.edu.cn).