COMPLEXITY-DISTORTION OPTIMIZED MOTION ESTIMATION ALGORITHM WITH FINE-GRANULAR SCALABLE COMPLEXITY

Li Zhang, Wen Gao

Institute of Computing Technology, Chinese Academy of Science
{zhangli, wgao}@jdl.ac.cn

ABSTRACT

Video encoding now is being implemented in various computing platforms with different computing capability, the requirement on the encoding complexity is also different according to different applications. As the most computation-intensive part of video encoding, the ME (motion estimation) should have a scalable complexity. This paper proposes a ME algorithm with fine-granular scalable complexity, a more important feature of the proposed algorithm is that it seeks for the complexity-distortion optimization. The given computation budget will be allocated to each MB (macroblock) in one frame. Each MB will consume its allocated computation by a hybrid search pattern. Experimental results show that the proposed algorithm can get a better computation-distortion performance than the existing ME algorithms.

1. INTRODUCTION

The ME (motion estimation) is the most computation-intensive part of a video codec, that is why there are so many fast ME algorithms such as [1]-[4] in literature. These algorithms aim at reducing the complexity while trying to keep the search accuracy as possible, but the complexity of these algorithms is not scalable. Nowadays the computing capability of different computing platforms differs greatly, even in the same computing platform, the constraint on encoding complexity is also different for different applications. For the conventional ME algorithms, if the actually spent complexity has exceeded the computation budget, the ME process has to be terminated ungracefully with search efficiency lost. On the contrary, if the computing capability is greater than that needed by the ME algorithm, the redundant computing capability is wasted without gain in search accuracy.

There have been some ME algorithms which have scalable complexity such as [5]-[7]. In these algorithms, the complexity scalability is achieved by adjusting some parameters or thresholds. As a result, the complexity of these algorithms is only "layered" scalable. But they can not adjust their complexity accurately according to the computation budget.

Different from the above algorithms, the proposed algorithm can accurately control its complexity to be equal with any given computation budget, thus the fine-granular scalable complexity is achieved. More important, the proposed algorithm seeks for the complexity-distortion optimization. The complexity-distortion optimization is to get better distortion result with a given computation budget. To achieve this goal, a computation allocation is used to allocate computation to different MBs in one frame, then a hybrid search pattern is used by each MB to fully utilize the allocated computation.

The rest of the paper is organized as follows: section 2 is to present the computation allocation strategy, section 3 describes the hybrid search pattern, experimental results are in section 4 and conclusions in section 5.

2. COMPUTATION ALLOCATION FOR COMPLEXIYT-DISTORTION OPTIMIZATION

In the following of this paper, we will use the number of searching points to evaluate the complexity of a ME algorithm. The MSE (mean square error) between the current MB (macroblock) and the reference MB is used to evaluate the distortion. Suppose the MSE of the initial searching point is MSE0, after checking all the searching points, the MSE of the best point is MSE1, then the distortion benefit brought by ME is (MSE0-MSE1). Suppose the searching points budget for one frame is C and there are N MBs in one frame. The complexity-distortion optimization targets at the following goal:

$$\max \sum_{i=0}^{N-1} D(MB_i) \qquad subject \sum_{i=0}^{N-1} C(MB_i) < C$$

$D(MB_i)$ is the distortion benefit of the ith MB, $C(MB_i)$ is the searching points that allocated to the ith MB.

Due to the variety of motion field, the distortion benefit differs greatly for different MBs in one frame. For those MBs which belong to a stationary background, the distortion benefit is usually zero since the MV is often zero, but those MBs on the moving foreground object can get larger distortion benefit if they can successfully find the optimal MV (motion vector). If more computation is

allocated to the MBs which have larger potential distortion benefit, the overall distortion benefit of the whole frame can be larger. But the distortion benefit of one MB is not known before taking its ME.

Previous work such as [8]-[9] has used an on-line strategy to allocate computation, take [8] for example, all the MBs in one frame are sorted into a list in descending order with their current MSE, at each allocation step, the computation is always allocated to the MB that has largest MSE. There exist some problems in this on-line strategy: the ME of all the MBs in this frame is actually processed in an interleaving way, a MB's ME may be interrupted by other MB's ME. Considering that the practical encoder always processes the MBs one by one, this interleaving ME process is not suitable for practical application. So we are using a heuristic method to pre-allocate computation before one frame's encoding.

For the ME in MPEG or H.26x, the MV prediction technique is used, one MB's MV is predicted according to the neighboring MBs' motion vectors. The initial searching point of the ME is the predicted MV. Suppose the MVD (motion vector difference) between the predicted MV and the best MV is (MVDx, MVDy). If the MVD is zero, there will be no distortion benefit. To further explore the relation between the MVD and the distortion benefit, we define the variable MVD_MB:

$$MVD\_MB = abs(MVDx) + abs(MVDy) \qquad (1)$$

Note that the abs(MVDx) is to get the absolute value of MVDx. If the MVD_MB is large, the best MV is far away from the predicted MV, the difference between the best candidate MB and the initial candidate MB is larger, thus the difference between the smallest MSE and the initial MSE is larger. So the MVD_MB can be used as an indicator for the potential distortion benefit. In the proposed strategy, the computation is allocated in proportion to the MVD_MB.

But the MVD_MB of one frame's MBs are still not known before this frame's ME, fortunately, the MVD_MB can be predicted by using the context motion field information. Since there exists similarity between two successive frames' motion fields, the distribution of the MVD_MB in two successive frames should also be similar, the distribution of MVD_MB in the previous frame can be used to predict the next frame's.

The MVD_MB of the co-located MB in the previous frame can be used to predict the current frame's MVD_MB, but according to our experimental result, this MB to MB prediction is not so accurate since the two successive frame's motion fields are not totally same, so we use a window-based prediction, a 3x3 MB window in the previous frame is selected (the center MB of this window is co-located to the current MB), the average value of the 9 MBs' MVD_MB is used to predict the current MB's MVD_MB.

To describe the details of the proposed computation-allocation strategy, we assign each MB an index (t, i, j), t is the frame index, i and j means that the MB is in the ith MB-row and jth MB-column of frame t., the variable MVD_region(t,i,j) is defined as:

$$MVD\_region(t,i,j) = 1 + \sum_{m=i-1}^{i+1} \sum_{n=j-1}^{j+1} MVD\_MB(t,m,n) \quad (2)$$

Supposing one frame consists of MxN MBs, the given total searching points of each frame is C, the searching points allocated to MB(t,i,j) is C(t,i,j). In the proposed strategy, C(t,i,j) is allocated in proportion to MVD_regoin(t,i,j). As mentioned above, the MVD_region(t,i,j) is replaced by MVD_region(t-1,i,j):

$$C(t,i,j) = \frac{C * MVD\_region(t-1,i,j)}{(\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} MVD\_region(t-1,m,n))} \quad (3)$$

In (2), a constant value 1 is added so that *MVD_region(t,i,j)* will always be non-zero, so the C(t,i,j) will not be zero in (3), i.e. each MB's can at least get a non-zero computation budget.

3. HYBRID SEARCH PATTERN

After the computation allocation is completed, each MB should use a search pattern to fully utilize the allocated computation without exceeding or wasting. There are many search patterns for ME, they can be classified into three types:

Local Search: these are the most popular search patterns such as [1]-[2], these search patterns assume that the ME's distortion surface is monotonic in a local region around the optimal MV, so these search patterns are compact and local-based. They can get optimal MV with very few searching points when the motion is slow and small, but are easy to be trapped into local minimal when the motion is intensive.

Global Search: in the global search such as [4], the searching points compose of sparse grids which span the whole search window. The local minimal can be avoided under global search, but the computation waste may happen when the best MV is very near to the initial searching point.

Full Search: the full search checks all the searching points in the search window, it can get the best search accuracy with the highest complexity.

We evaluate a ME search pattern by its CD performance, the CD performance is the ratio of the distortion benefit and the computation result that is needed to get this distortion benefit. The CD performance order should be local search > global search > full search, but when the computation budget is enough, the distortion benefit order should be full search > global search > local search.

To get the best distortion result under any computation budget, we propose a hybrid search pattern, local search pattern is firstly used, global search is then used, full search

is finally used. The search at any stage should be terminated if the allocated computation budget is exhausted.
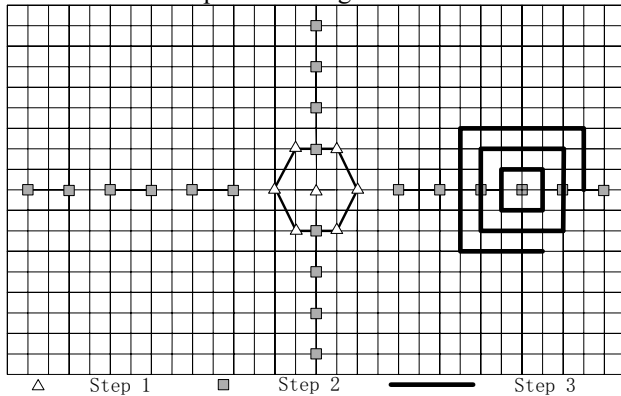

Figure.2 Hybrid search pattern

Fig.2 shows the detail of the proposed hybrid search pattern, it can be divided into 3 steps:

Step 1: Local search. The HEXS in [2] is used for the local search, the center of the initial hexagon is the predicted MV. The HEXS is terminated whenever the number of searching points has been equal to the budget.

Step 2: Global search. The distribution of the searching points should conform to the distribution of the optimal MV. As previous work such as [3]-[4] has pointed out: the distribution of the optimal MV is cross-center biased, so the cross-shape is used for the global search, the center of the cross is the best point in step 1, the distance between two successive points is 2. Suppose the remaining searching points budget after step 1 is C, and there are W searching points in the horizontal direction and H searching points in the vertical direction, then we have the following equation:

$$W + H = C \qquad (4)$$

It should be pointed out that a MB's motion may be either horizontal biased or vertical biased, so the computation should be unequally allocated to horizontal/vertical direction of the cross. We can predict the current MB's motion trend with the motion information of the co-located MB in the previous frame. If current MB is MB(t, i, j), the MVD of MB(t-1, i, j) is (MVDx, MVDy), W and H can be decided as follows:

$$W = \min(\ C * MVDx / (MVDx + MVDy)\ ,\ T\ )$$
$$H = \min(\ C * MVDy / (MVDx + MVDy)\ ,\ T\ ) \qquad (5)$$

In (5), W and H are constrained by an upper bound T because a huge search range of the cross is not reasonable in practical ME.

Step 3: Spiral search. Since the search range of the cross search is limited, there may be redundant computation after cross search when the budget is plenty. The full search uses a spiral pattern, the center of the spiral is the best point in step 2. When the remaining computation budget for the spiral search is very large, the spiral search will cover the whole search window and the optimal MV will not be missed.

4. EXPERIMENTAL RESULTS

To verify the performance of the proposed algorithm, we take two experiments. Experiment 1 is to compare the proposed algorithm with the full search. The full search searches all the searching points in a square window with the search range being [-R,R], if there are N MBs in one frame, the total searching points of one frame is $(2R+1)^2 \times N$. In experiment 1, R varies from 1 to 16, so the computational complexity fluctuates in a wide range. With each R, the equal computation budget is assigned to the proposed algorithm. The PSNR is used to evaluate the resulted distortion of ME. Three typical CIF sequences are chosen: Stefan (acute motion), Foreman (relatively fast motion), Paris (stationary background), the T in (5) is 32.
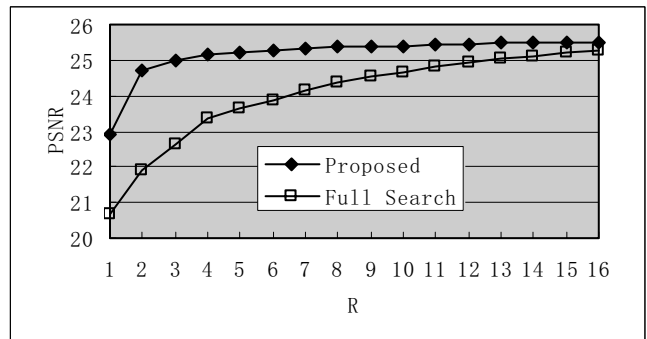

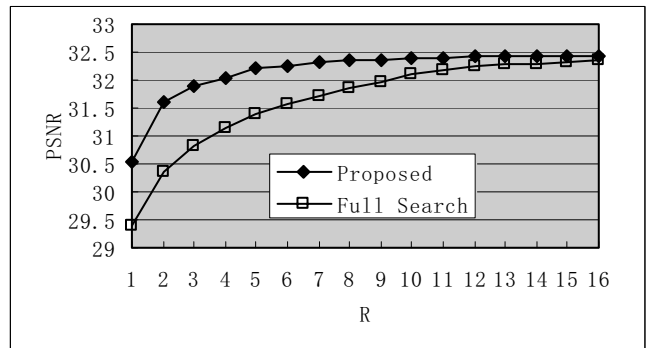Figure.3 Result on "Stefan" in experiment 1
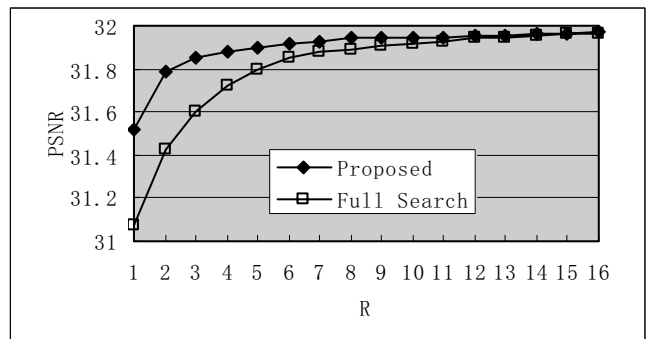

Figure.4 Result on "Foreman" in experiment 1

Figure.5 Result on "Paris" in experiment 1

The above 3 figures show the results of experiment 1. It can be seen that the proposed algorithm's complexity is fine-granular scalable in a wide range, it also can be seen that the proposed algorithm can get a higher PSNR than full search at each comparing point set by R.

As mentioned above, the full search has the worst CD performance among the existing ME algorithms, so experiment 1 can not demonstrate the optimized CD performance brought by the computation allocation. In experiment 2, the proposed algorithm is compared with HEXS in [2] which has a high CD performance.

Stefan		
Searching points per frame	Proposed	HEXS
4000	23.0808	22.5035
5000	23.5611	23.0461
6000	23.9976	23.3080
7000	24.3420	23.4298
8000	24.4959	23.4947
Foreman		
Searching points per frame	Proposed	HEXS
4000	30.7624	30.0971
5000	31.0454	30.7533
6000	31.1913	31.0527
7000	31.3737	31.2423
8000	31.5044	31.3396
Paris		
Searching points per frame	Proposed	HEXS
4000	31.5848	30.9507
5000	31.7315	31.4810
6000	31.7624	31.6248
7000	31.7886	31.6549
8000	31.7974	31.6746

Table.1 Result of experiment 2

Since the computation spent by the HEXS is small, the computation budget in experiment 2 only fluctuates in a range of small value: from 4000 searching points per frame to 8000 searching points per frame. In HEXS, the computation budget is uniformly allocated to each MB, a MB terminates its search when its computation budget is exhausted. The experimental result is in Table 1. It can be seen that the PSNR of the proposed algorithm outperformed the HEXS in all the three sequences. The PSNR improvement is more obvious in "Foreman" and "Stefan", this is because that there exists fast moving foreground object in the two sequences, more computation will be allocated to the MBs which are located in foreground object, these MBs can take global search to avoid the local minimal. On the contrary, nearly all the MBs of "Paris" belong to stationary background, the potential distortion benefit of these MBs are equally small, thus the effect of computation allocation is not so obvious.

5. CONCLUSIONS

To meet variable requirement on the complexity of ME, this paper presents a ME algorithm with fine-granular scalable complexity. To get an optimized CD performance, a computation allocation strategy is used to spend the computation at where it is needed. The context information of motion field is used as the clue of allocation. A hybrid search pattern which consists of local search, global search and full search is proposed. This search pattern can fully utilize any computation budget. Experimental results show that the proposed algorithm is not only fully scalable in a wide complexity range and but also has better CD performance than the existing ME algorithms.

6. REFERENCES

[1] J. Y. Tham, S. Ranganath, M. Ranganath, and A. A. Kassim: "A novel unrestricted center biased diamond search algorithm for block motion estimation", IEEE Trans, CSVT, vol. 8, pp. 369–377, Aug. 1998.
[2] Ce Zhu, Xiao Lin, and Lap-Pui Chau: "Hexagon-Based Search Patten for Fast Block Motion Estimation", .IEEE Trans, CSVT, pp. 349-355, Vol.12, No.5, May 2002.
[3] Chun-Ho Cheung, Lai-Man Po, "A Novel Cross-Diamond Search Algorithm for Fast Block Motion Estimation. IEEE Trans, CSVT, December, 2002.
[4] Zhibo Chen, Zhou Peng, Yun He: "Fast integer pel and fractional pel motion estimation for JVT", JVT-F017 December 2002.
[5] K. Lengwehasatit and A. Ortega, "Computationally scalable partial distance based fast search motion estimation," Proc ICIP'2000, vol. 1, pp. 824–827, Sept. 2000.
[6] S. Mietens, P.H.N. de With, and C. Hentschel, "Computational-complexity scalable motion estimation for mobile MPEG encoding," IEEE Transactions on Consumer Electronics, vol. 50, no. 1, pp. 281–291, Feb. 2004.
[7] Deepak S. Turaga, Mihaela, van der Schaar, Beatrice Pesquet-Popescu, "Complexity Scalable Motion Compensated Wavelet Video encoding". IEEE Trans, CSVT, August, 2005.
[8] Pol-Lin Tai, Shih-Yu Huang, Chii-Tung Liu, and Jia-Shung Wang, "Computation-Aware Scheme for Software-Based Block Motion Estimation", IEEE Trans, CSVT, VOL, 13, NO.9, September 2003.
[9] Zhi Yang, Hua Cai , and Jiang Li. "A Framework for Fine-Granular Computational-Complexity Scalable Motion Estimation", 2005 IEEE International Symposium on Circuits and Systems.