Generalized Partial Distortion Search Algorithm for Block-Matching Motion Estimation

Chun-Ho Chung and Lai-Man Po
Email: celot@cityu.edu.hk and celmpo@cityu.edu.hk
Department of Electronic Engineering, City University of Hong Kong,
83 Tat Chee Avenue, Kowloon, Hong Kong.

Abstract—The quality against speed control for real-time video applications, such as the low-bit-rate video conferencing or the high quality video entertainment, usually absents from many traditional fast block motion estimators. In this paper, a novel block-matching algorithm for fast motion estimation named generalized partial distortion search algorithm (GPDS) is proposed. It uses half-way-stop technique with progressive partial distortion (PPD) to increase the chance of early rejection of impossible candidate motion vectors at very early stages. Simulations on PPS show that 28 to 38 times computational reduction with only 0.45-0.50dB PSNR performance degradation as compared to full search algorithm. In addition, a new normalized partial distortion comparison method is also proposed for enabling the control of searching speed against the prediction accuracy by a quality factor k. This method also generalizes the conventional partial distortion search algorithm when k is equal to 1, and the normalized partial distortion search algorithm (NPSD) when k is equal to infinity. Experimental results show that GPDS with use of PPD could provide PSNR performance very close to full search algorithm with 7 to 17 times, and to NPSD with 22 to 35 times speedup, respectively, as compared to full search algorithm.

Keywords—generalized partial distortion search algorithm, quality factor, progressive partial distortion, motion estimation.

I. INTRODUCTION

The major gain of compression ratio in many video coding like ISO MPEG-1/2/4 and ITU-T H.261/263 is achieved by motion estimation. Motion estimation efficiently removes the temporal redundancy between successive frames by block-matching algorithms (BMA). Block-based motion estimation is the most practical approach to obtain motion compensated prediction frames. It divides frames into equal-sized rectangular blocks and finds out the displacement of the best-matched block from previous frame as the motion vector to the block in the current frame within a search region. Full search (FS) algorithm is the most straightforward brute force BMA, which provides the optimal results by matching all possible candidates within a search window (±w pixels). In contrast, it is the most computational intensive as compared to other popular traditional fast BMA like three-step search (3SS) [1], new three-step search (NS3S) [2] and four-step search (4SS) [3], which reduce computational complexity by limiting the number of search points within the search window. However, traditional BMA are easily trapped into local minima as they are based on the monotonically increasing matching criteria [4] around an optimal motion displacement for iterative determination, but, there are several such local minima fake to be the global optimum solution within the search region. It results in loss of visual quality and the mean absolute distortion (MAD) is always higher than FS. Recently, some fast search algorithms perform block matching as in FS without limitation of checking points, especially the normalized partial distortion search algorithm (NPSD) [5]. NPSD introduces the normalized cumulative partial distortion criteria for early rejection of impossible candidate motion vectors (CMV). However, they all lack the control between speed and quality. In this paper, a novel fast BMA named generalized partial distortion search algorithm (GPDS) is proposed. The proposed algorithm consists of two parts. The first part is to increase the speed of NPSD by introduction of progressive partial distortion (PPD) at very early stages. The second part is a new normalized partial distortion comparison method with enabling the quality control against different speedup ratio. Experimental results of GPDS with PPD are also given to show the trade-off between searching speed and the prediction accuracy for different applications.

II. PROGRESSIVE PARTIAL DISTORTION

Suppose the block size is M × M. Let \(a_{x,y}\) and \(b_{x,y}\) be the pixel value of row \(x\) and column \(y\) of block \(a\) and \(b\), in current and previous frame, with horizontal and vertical sampling by \(q\) and \(r\) pixels, respectively. The block distortion measure (BDM) \(d(x,y,q,r;M)\) of \(b_{x+y,q+r}\), with \((u,v)\) displacement from \(a_{x,y}\), is: \(q=r=1\), using sum absolute error (SAE):

\[
d(x,y,q,r;M) = \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} |a_{i,j} - b_{i+u,j+v}| \quad (1)
\]

Conventional partial distortion search algorithm (PDS) [6] provides optimal result equal to FS with speedup ratio about twice that of FS. PDS rejects impossible CMV by means of half-way step technique with partial distortion comparison to the current minimum distortion \(D_{MIN}\) in a pixel-wise basis. To reduce the number of comparison and increase the chance of early rejection of impossible CMV, NPSD divides the matching blocks into \(P\) groups \((P=16)\), as shown in Fig. 1, and performs partial distortion comparison against the normalized current minimum distortion \(D_{MIN}\) as a group of \(M^2/P\) pixels basis. Experimental results in [5] show that NPSD outperforms other BMA such as 3SS with PSNR performance close to FS and highly reduces the computations 12-13 times. It results in a saving of multiples of 16 pixel-operations for each impossible
CMV. However, NPDS limits the maximum computational reduction to the number of pixels in the first partial distortion. If \( M^2/p = 16 \) times for \( M = P = 16 \). In order to further increase the rejection rate more efficiently, we propose to use progressive partial distortion (PPD) at the first few stages of the NPDS, as the first partial distortion, which is further divided into \( H \) smaller dimensions and equal-sized partial distortions. The maximum computational reduction of NPDS can then be increased up to \( M^2/p = 64 \) times if \( H = 4 \).

With PPD, matching blocks are firstly divided into \( P \) equal-sized partial distortions \( d_{i,p} \) by horizontal \( s \) and vertical \( t \) pixels sampling, respectively. Then, the first partial distortion \( d_1 \) is further sub-sampled by horizontal \( k \) and vertical \( l \) pixels, respectively. The dimension of the first partial distortion \( d_1 \) is therefore reduced from \( \frac{d^2}{k \times l} \) to \( \frac{d^2}{M \times N} \) in order to support \( H \) equal-sized partial distortions. Thus, \( d_1 \) is replaced by \( \sum_{p=1}^{P} d_{i,p} \) for \( n = 1 \leq k \times l \) and the two types of partial distortions are defined as in Eqn(2) and Eqn(3).

\[
d_{p,k,l} = d_{(i+p,j+l)} \quad \text{for} \quad 1 \leq k \times l \leq H
\]

\[
d_{p} = d_{(i+p,j+l)} \quad \text{for} \quad 1 \leq p \leq P
\]

Since PPD is introduced at the very early stages, the total number of partial distortions is equal to \( G \), where \( G = H \times P \times 1 \). Each group of pixels gives partial distortion \( d_{p} \) for \( 1 \leq G \leq G \). The original-p-th cumulative partial distortion \( D_p = \sum_{k=1}^{P} d_{k} \) of NPDS is rewritten into the \( G \)-th cumulative partial distortion \( D_p \) as defined in Eqn(4). The normalization of the \( p \)-th cumulative partial distortion, \( D_p = \frac{D_p}{n} \), where \( n = P \times M^2/P \times \frac{M^2}{P} \), is rewritten into the \( p \)-th normalized cumulative partial distortion \( D_p \) as defined in Eqn(5).

\[
D_p = \sum_{k=1}^{P} d_{k} \quad \text{for} \quad 1 \leq n \leq G
\]

\[
D_p = \frac{D_p}{n} \quad \text{for} \quad 2 \leq p \leq P \times \frac{M^2}{P} \times \frac{M^2}{P} \times \frac{M^2}{P}
\]

Regular sub-sampling \( \psi(\theta,\phi) \rightarrow \psi(\theta,\phi) \) with \( i \) pixel(s) taken out at a time from a group of \( S = M^2/P \) pixels has possible combinations of \( \sum_{i=1}^{S} \left( \begin{array}{c} \prod_{i=1}^{P} \left( \begin{array}{c} \prod_{i=1}^{P} \end{array} \right) \end{array} \right) \), where \( \psi(\theta,\phi) \rightarrow \psi(\theta,\phi) \). Thus, the total combination \( \psi(\theta,\phi) \rightarrow \psi(\theta,\phi) \) is very large. Regularity favors practical implementations. Fig.2(a)-(c) show grouping of 1, 8 and 4 pixels on \( d_1 \), respectively. Further sub-samplings on the \( d_1 \), such as \( (1,1,2,4,4,4) \) decimating sequence in Fig.2(d), gives hierarchical progressive structure on \( d_1 \).

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|}
\hline
Method & PS & ABS & +/– & Comp. & Total SpeedUp \n\hline
PS & 108554 & 206 & 156010 & 1.00 & 0.00 \n
PDS & 45469 & 1817 & 69294 & 2.27 & 0.00 \n
SSS & 801 & 258 & 12301 & 12.80 & 1.49 \n
SS & 6437 & 204 & 9869 & 16.07 & 0.74 \n
SNS & 7395 & 241 & 11695 & 15.07 & 0.94 \n
NPDS & 4921 & 454 & 13505 & 13.17 & 0.29 \n
PPD & 1071 & 638 & 4017 & 35.48 & 0.96 \n
PPD & 13313 & 2480 & 4455 & 35.60 & 0.71 \n
PPD & 3550 & 492 & 5672 & 27.96 & 0.50 \n
PPD & 1043 & 402 & 918 & 19.37 & 0.33 \n
PPD & 1148 & 2120 & 3987 & 30.78 & 0.86 \n\hline
\end{tabular}
\caption{Computer reduction and PSNR performance comparison on sequence "tennis".}
\end{table}

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|}
\hline
Method & PS & ABS & +/– & Comp. & Total SpeedUp \n\hline
PS & 108554 & 206 & 156010 & 1.00 & 0.00 \n
PDS & 45469 & 1817 & 69294 & 2.27 & 0.00 \n
SSS & 801 & 258 & 12301 & 12.80 & 1.49 \n
SS & 6437 & 204 & 9869 & 16.07 & 0.74 \n
SNS & 7395 & 241 & 11695 & 15.07 & 0.94 \n
NPDS & 4921 & 454 & 13505 & 13.17 & 0.29 \n
PPD & 1071 & 638 & 4017 & 35.48 & 0.96 \n
PPD & 13313 & 2480 & 4455 & 35.60 & 0.71 \n
PPD & 3550 & 492 & 5672 & 27.96 & 0.50 \n
PPD & 1043 & 402 & 918 & 19.37 & 0.33 \n
PPD & 1148 & 2120 & 3987 & 30.78 & 0.86 \n\hline
\end{tabular}
\caption{Computer reduction and PSNR performance comparison on sequence "football".}
\end{table}
III. GENERALIZED PARTIAL DISTORTION COMPARISON

GPDS is to provide generalization of the normalized cumulative partial distortion $D_n$ so that the NPDS matching criteria against $D_{MIN}$ can be adjusted to provide different speedup ratio and prediction quality. The generalized normalization of n-pixel cumulative partial distortion $D_n$ from Eqn(5) can be rewritten as:

$$
\tilde{D}_n = \frac{D_n}{f(n)}; \text{where } f(n)=n.
$$

Generalization function $f(n, k)$ is designed to replace $f(n)$ by introducing a quality factor or speedup factor $k$, which relates the PSNR performance from conventional PDS to NPDS. The generalized $\tilde{D}_n$ provides matching criteria for optimal result as in PDS when $k=1$, i.e. $f(n, k)|_{k=1}=M^2$. When $k \to \infty$, GPDS from Eqn(2) gives the performance as in NPDS, i.e. $f(n, k)|_{k=\infty}=M$. Thus, $f(n, k)$ is defined as in Eqn(7) and plotted as in Fig.4.

$$
\hat{f}(n, k) = n + \left( \frac{M^2 - n}{k} \right)
$$

In general, $f(n, k)$ starts at $M^2$ (i.e. PDS matching criteria) and going to become $f(n)=n$ (i.e. NPDS matching criteria) as $k$ increases from 1 to infinity. Thus, it is suitable for adjustment of speedup versus quality between PDS and NPDS. As $f(n, k)$ decreases with $k$ dramatically for small $n$, but gets to become $M^2$ for larger $n$. It implies that $\tilde{D}_n$ increases with $k$ dramatically at small $n$, and thus favors early rejection purposes. With the dramatical slope of $f(n, k)$, NPDS favors to give result near to that of PDS (at small value of $k$) by rejection of impossible CMV at early stages (small value of $n$). Since the proposed GPDS($v3$) gives $H=4$ equal-sized PPD with lower dimension at $d_k$, it increases the early rejection rate for the first group of $M^2/P=16$ pixels to 4 times than that of NPDS, which results in about three times of the searching speed with MSE performance degraded slightly. Thus, GPDS($v3$) is very suitable for generalization.

IV. EXPERIMENTAL RESULTS

The proposed algorithm GPDS with PPD is simulated using the luminance of the popular SIF (380×240) video sequence “tennis” and “football”. The block size and search window used are 16×16 ($M \times M$) and $\pm 7$ ($w$), respectively. In NPDS, each block is sampled evenly in both horizontal and vertically by $s=4$ and $t=4$ pixels, respectively, into $P=16$ partitions. For PPDS($v3$) and GPDS, the first partial distortion $d_1$ is sub-sampled by $k=2$ and $l=2$ into $H=4$ PPD. All SS, N4SS and N3SS are implemented to use half-way-stop BD algorithm for fair comparison. Their prediction quality in terms of PSNR and MSE, and computational complexity are compared.

Results using various PPD can be referred to second section II, Table I and II. In general, PPDS gives 19.37 to 62.39 times speedup with 0.30 to 0.97dB PSNR degradation as compared to FS. The PPDS($v3$) employs total $G=19$ partial distortions with quality-speed compromised performance and is generalized by $f(n, k)$ to form GPDS for normalized partial distortion comparison.

Table III compares the speedup ratio and PSNR performance of different BMA against GPDS of different quality factor $k$. GPDS maintains its PSNR performance very close to FS at $7.24$ and $17.21$ times speedup, for sequence “tennis” and “football”, respectively. As compared to traditional BMA, firstly, GPDS has a speedup ratio of 1.49-2.11 times with better or similar PSNR performance. Secondly, it also gives better PSNR performance by 0.21-1.41dB more with searching speed.
at 16.24(k=42) times for "tennis" and 23.56(k=31) times for "football", respectively, which are just faster than traditional BMA. As compared to NPD, GPDS has PSNR performance very close to NPD at k=122 on both sequences with computational reduction of 1.82-2.37 times faster than that of NPD. Fig.3 shows MSE performance of various BMA on "tennis" in which GPDS at k=42 almost overlaps the MSE performance of FS at 16.24 times speed up, which also outperforms all traditional BMA.

It is noted that multiplication operations exist in Eqn(6). In PPDS comparison, all multiplication operations are translated into combinations of "left-shift(LS)" and "addition(+/-)") operations since the numerator $M^2$ of the matching criteria can be simplified into power of 2 with the values of H and P, which are also power of 2 in $M^2D_{av} > f(n)D_{MIN}$. However, for generalization function in Eqn(7), smooth curve of speedup ratio and distortion against quality factor k are expected theoretically from PDS to NPD, but impractical for implementation. Thus, nineteen f(n,k) values are computed for desired k in advance by means of integer precisions and operation from k=1 to 256 such that f(n,k) saturates at k=256. It results in stepping-behavior of the speedup ratio and PSNR performance against k as shown in Fig.5 and Fig.6. Since dramatical behavior exists in the small k and the proposed working value for k ≤ 100, which provides obvious speedup ratio up to 21.60-32.04 times with 0.18-0.20dB slightly degradation as compared to FS.

V. Conclusions

In this paper, progressive partial distortion (PPD) is proposed at the first few stages of NPD to increase the early rejection rate of impossible candidate motion vectors. Simulation on NPD with different PPD provides computational reduction up to 62.39 times with less than 0.98dB degradation on PSNR performance as compared to FS. In addition, generalized partial distortion search algorithm (GPDS) is proposed. It introduces a quality factor k to the normalized partial distortion comparison so that NPD can be adjusted to provide an optimization on the trade-off between searching speed and quality from PDS to NPD. Experimental results show that the proposed GPDS has PSNR performance very close to NPD at k=122 with computational reduction up to 2.37 times as compared to NPD. Thus, GPDS is very suitable for wide range of video applications such as speed-oriented video conferencing and high quality video coding.

VI. Acknowledgment

The work described in this paper was substantially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China [Project No. 9040423].

REFERENCES