FAST RATE-CONSTRAINED N-STEP SEARCH ALGORITHM FOR MOTION ESTIMATION

Muhammed Z. Coban　　Russell M. Mersereau

Center for Signal & Image Processing (CSIP),
School of Electrical & Computer Engineering,
Georgia Institute of Technology,
Atlanta, GA 30332-0250, USA.
e-mail: (coban, rmm)@ee.gatech.edu

ABSTRACT

A fast N-step search algorithm for rate-constrained motion estimation is presented. The motion vectors are selected from a search window based on a rate-distortion criterion by successively eliminating the search positions at each step. The performance of the proposed algorithm is identical to the performance of the conventional rate-constrained N-step search algorithm, with considerable reduction in computation. Computational savings increase in parallel with the increases in the rate constraint and the number of steps.

1. INTRODUCTION

Motion estimation is an essential part of conventional video coding systems. Motion estimation and compensation techniques are used to remove the temporal redundancies that exist in video sequences. Block matching algorithms (BMAs) are commonly used for motion estimation. The full search BMA is widely used for motion estimation in video coding. It exhaustively searches for the best matching block within a search window and finds the optimal motion vector that minimizes the distortion. The disadvantage of the full search algorithm is its computational cost. In order to reduce the computational complexity without degrading the estimation performance significantly, various fast search algorithms have been proposed, such as the three-step search, the 2-D-logarithm search and the cross search algorithms [1, 2]. These fast search algorithms reduce the computational cost by reducing the number of tested motion vectors at the expense of the accuracy of the motion estimate. The N-step search (NSS) algorithm, which is a modified version of the three-step search algorithm, evaluates the matching criterion only at a subset of the candidate MV positions at each step.

In conventional motion estimation systems the mean absolute difference (MAD) and the mean squared error (MSE) are commonly used as the matching criteria. We prefer the MAD as the matching criterion because it requires no multiplication and gives similar performance as the MSE. Motion estimates based on these matching criteria do not necessarily give the best rate-distortion (R-D) performance and lead to irregular motion fields which in turn lead to an increase in the MV bit rate. For low bit rate video coding applications, the MVs take up a substantial amount of the available bit rate budget. For these severely rate-constrained applications, optimal bit allocation among the MV rate and the prediction error rate subject to a rate constraint becomes an important task [3, 4, 5].

In this paper, we propose a fast rate-constrained N-step search algorithm which finds the R-D optimized MVs by successively eliminating the search positions at each step of the N-step search algorithm according to an inequality condition that limits the number of matching evaluations at each step, similar to the successive elimination algorithm of Li and Salari [6], which was proposed for exhaustive search. The proposed algorithm is based on the N-step search algorithm. The number of search positions to be tested for a match at each step is reduced by utilizing the successive elimination algorithm.

2. R-D OPTIMIZED BMA

In BMAs where no constraint on the MV rate is imposed, the motion vector $\vec{d} = (d_x, d_y)$ is chosen based on minimizing the distortion. Efficient motion estimation algorithms must minimize MV rate as well as distortion.

THIS WORK WAS SUPPORTED BY THE NATIONAL SCIENCE FOUNDATION, CONTRACT MIP-9205853.

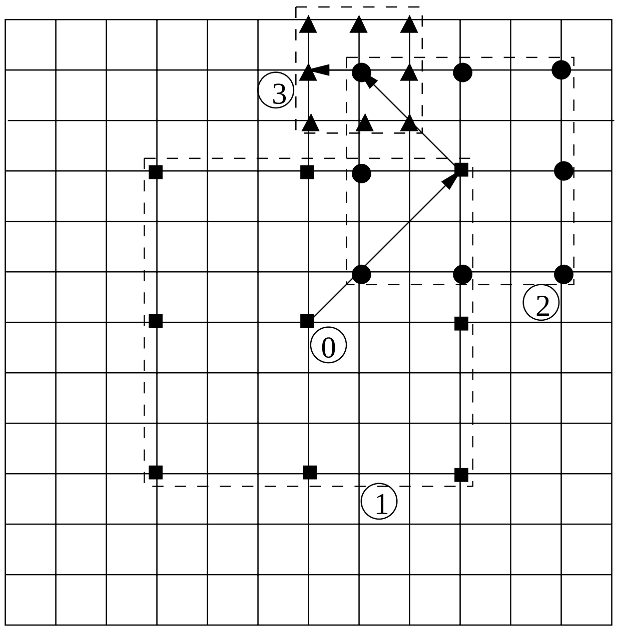

Figure 1: An example of the proposed NSS (N=3) algorithm. The cost function is first evaluated at position 0, then inequality (5) is checked at grid points on 1 until a minimum is found. This procedure repeated on grids 2 and 3.

Consider a frame which is partitioned into K blocks. Let $\vec{d}_k \in \mathcal{S}$ be the MV selected for block k. Then MV field assigned to the frame is given by the K-tuple, $\mathcal{D} = (\vec{d}_1, \cdots, \vec{d}_K) \in \mathcal{S}^K$. For inter-frame coding, the problem of finding the MV field that minimizes the distortion for a given rate constraint can be formulated as finding points on the convex hull of all possible R-D points which can be found by Lagrangian minimization [7],

$$J_{min}(\lambda) = \min_{\mathcal{D} \in \mathcal{S}^K} \sum_{k=1}^{K} D_k(\vec{d}_k) + \lambda \; R_k(\vec{d}_k), \qquad (1)$$

where λ is the Lagrange multiplier, and D_k and R_k are the distortion and the number of bits associated with the MV for block k, respectively. As a result, the cost function $J(\lambda)$ is used as a matching criterion instead of the MAD criterion. If each block is coded independently, the solution to equation (1) can be reduced to minimizing the Lagrangian cost function of each block, i.e.,

$$J_{min}(\lambda) = \min_{\vec{d} \in \mathcal{S}} D_k(\vec{d}) + \lambda \; R_k(\vec{d}). \qquad (2)$$

In most motion estimation algorithms, MVs are differentially encoded. Hence, the blocks are not independently coded. In order to simplify the problem, although the MVs are coded differentially, the blocks will be treated as if they are being coded independently.

3. PROPOSED ALGORITHM

In this section, we establish an inequality relation to limit the search process at each step of the N-step search while preserving the R-D optimized solution for the motion vector, given the Lagrange multiplier λ. N-step search algorithm operates in a hierarchy of decreasing search distances, at each step matching most closely to one of nine locations (grid points), including the origin and the eight neighboring positions. At each step the distance of the search pixels from the new center is decreased by one to obtain a finer-resolution estimate.

The MV components (d_x, d_y) can take values $-w \leq d_x, d_y \leq w$. Assuming the predicted MV for the current block is $(\hat{d}_x, \hat{d}_y)$, the number of bits spent on the MV for block k is $R_k(\vec{d}) = -\log_2 p(d_x - \hat{d}_x) - \log_2 p(d_y - \hat{d}_y)$ where $p(\cdot)$ is the probability function of the differential MV components. From the triangle inequality, we obtain

$$\begin{aligned} &\left| \overbrace{\sum_{\vec{r} \in \mathcal{W}_k} |I(\vec{r}, t)|}^{R} - \overbrace{\sum_{\vec{r} \in \mathcal{W}_k} |I(\vec{r} + \vec{d}, t-1)|}^{M(\vec{d})} \right| + \lambda \; R_k(\vec{d}) \\ &\quad \leq \underbrace{\sum_{\vec{r} \in \mathcal{W}_k} |I(\vec{r}, t) - I(\vec{r} + \vec{d}, t-1)|}_{MAD(\vec{d})} + \lambda \; R_k(\vec{d}) \qquad (3) \end{aligned}$$

where $I(\vec{r}, t)$ is the intensity at position $\vec{r}$ of frame t. $\mathcal{W}$ is the matching block of size $M \times M$ and $\mathcal{S}$ is the search window of size $(2w+1) \times (2w+1)$. R is the sum norm of the reference block in the current frame, whereas, $M(\vec{d})$ is the sum norm of a candidate block in the previous frame.

Assuming we have a motion vector which yields a cost function value $J_{min}(\lambda) = MAD(\vec{d}_{min}) + \lambda \, R_k(\vec{d}_{min})$ at a position on the grid of the nth step, we should search for motion vectors with lower cost function value, i.e.,

$$\left| R - M(\vec{d}) \right| + \lambda \; R_k(\vec{d}) \leq J_{min}(\lambda), \qquad (4)$$

on the same grid. This leads to the result that we should only evaluate the cost function at matching positions which satisfy

$$\begin{aligned} R - J_{min}(\lambda) &\leq M(\vec{d}) - \lambda \; R_k(\vec{d}) \\ M(\vec{d}) + \lambda \; R_k(\vec{d}) &\leq R + J_{min}(\lambda) \qquad (5) \end{aligned}$$

Table 1: Performance Evaluation of the Proposed NSS Algorithm

		$\lambda = 0$			$\lambda = 50$			$\lambda = 100$		
		N=3	N=4	N=5	N=3	N=4	N=5	N=3	N=4	N=5
Miss America	Rate	326.6	336.5	346.7	233.9	234.1	234.6	224.8	225.1	225.3
	PSNR (dB)	37.5	37.39	37.36	37.42	37.42	37.38	37.25	37.24	37.21
	ANMC	19.5	23.4	26.9	11.0	12.0	12.7	7.4	8.1	8.6
Car phone	Rate	395.2	410.1	421.8	306.1	308.8	311.3	283.4	285.6	287.8
	PSNR (dB)	31.23	31.24	31.18	31.17	31.19	31.16	31.03	31.04	31.09
	ANMC	16.4	19.3	21.8	13.2	14.9	16.3	11.1	12.4	13.5
Suzie	Rate	393.2	411.8	420.4	333.8	344.6	348.3	307.1	314.2	315.9
	PSNR (dB)	29.60	30.58	30.87	29.53	30.50	30.84	29.40	30.32	30.57
	ANMC	19.8	23.7	27.1	16.1	18.8	21.1	14.4	16.6	18.4
Foreman	Rate	490.0	563.3	598.9	400.3	447.2	463.5	354.4	387.9	395.4
	PSNR (dB)	28.14	28.13	28.07	28.13	28.11	28.10	28.00	28.07	28.08
	ANMC	20.4	24.9	28.9	19.1	23.1	26.5	17.8	21.4	24.4
Mthr & Dotr	Rate	309.9	317.9	327.6	250.7	253.2	254.8	240.7	242.0	243.43
	PSNR (dB)	32.13	32.34	32.46	32.11	32.34	32.48	32.01	32.23	32.39
	ANMC	15.9	18.6	20.8	11.4	12.7	13.8	9.6	10.6	11.4

simultaneously on the grid points of the nth step(Figure 1). When these inequalities are satisfied at a search position, the cost function is evaluated at that particular position and if its value is less than the current value, the cost function value $J_{min}(\lambda)$ is updated. This method progressively confines the search space; hence the solution can be found in fewer matching evaluations at each step. Once the minimum value for the nth step is found, it becomes the initial minimum cost function for the next step of the N-step search. This goes on until the Nth step. Testing the inequalities in (5) is not computationally demanding compared to the evaluation of $J_{min}(\lambda)$ at each position.

3.1. Computational Complexity

In the evaluation of the inequality (5), for a frame of size $H \times W$, the sum norms of the reference blocks, R's, and the matching blocks, $M(\vec{d})$'s, must be known before any search can be done. For matching of a single block, one sum norm must be calculated for the reference block. This takes $M^2 - 1$ operations. The calculation of the sum norms of the matching blocks can be efficiently performed by the method described in [6]. In this method, first absolute sum norms over a window of size M on the columns of the previous frame is computed, then the sum norms over a window of size M on the resulting rows is computed. This method yields approximately $4M^2$ additions per block. This is twice the number of additions needed to perform a single block matching evaluation, which is $2M^2 - 1$ additions. The total computation overhead for each reference block including the computation of the sum norm of the reference block is approximately 2.5 times the computation required to perform a single block matching evaluation. In addition to this overhead, there is of course the computational cost of searching eight positions that can satisfy inequality (5) at each step. The worst case computational complexity of the proposed algorithm becomes approximately 2.5 block matches more than the conventional NSS algorithm. In an N-step search algorithm, there are total of $(N \times 8 + 1)$ search positions.

3.2. Implementation Issues

The number of bits, $R(\vec{d})$, used to code the differential MVs are computed using the Huffman table defined in the draft H.261 standard [8] for coding MVs. All possible $\lambda\, R_k(\vec{d})$ values are calculated and stored before the search process begins. The MV prediction is taken as the median of the previous, above, and the above right MVs [9]. For each block, the initial cost function $J_{min}(\lambda)$ is chosen as the one which corresponds to the zero MV. The remaining motion vector positions are evaluated at the grid positions at each step.

The value of λ affects the speed of the estimation process. The $\lambda = 0$ condition corresponds to the unconstrained rate case, which finds the minimum MAD. As λ gets larger, i.e., more rate-constrained, the number of matching evaluations decreases. The reason for the decrease in the number of matching evaluations is that the sum norm surface of a matching block, i.e., the values of $M(\vec{d})$ for all search positions, is smoother than the $\lambda\ R_k(\vec{d})$ surface. Due to the fact that the variation of the $M(\vec{d})$'s is much smaller than the variation of the the $\lambda\ R_k(\vec{d})$'s over the search window, in the $M(\vec{d}) \pm \lambda\ R_k(\vec{d})$ term of (5), $\lambda\ R_k(\vec{d})$ term dominates. As λ gets larger, $\lambda\ R_k(\vec{d})$ values vary considerably from the bottom of the surface to the top of the $\lambda\ R_k(\vec{d})$ surface. If the estimate is an MV which corresponds to the bottom of the $\lambda\ R_k(\vec{d})$ surface, not many search positions where inequality (5) holds can be found. As a result, the solution can be found in fewer steps.

4. PERFORMANCE EVALUATION

The performance of the proposed algorithm, is studied using 150 frames of the "Miss America," "Car phone," "Suzie," "Foreman," and "Mother & Daughter" QCIF (176×144) sequences at 10 frames per second, with full pixel accuracy. Its performance is compared with that of conventional rate-constrained N-step search (NSS) algorithm for different N values. The block size is chosen as 16×16 for $N = 3, 4, 5$. The average MV bit rate, peak-SNR (PSNR) and number of matching calculations per block (ANMC) including the overhead are used as comparison parameters. These comparisons are tabulated in Table I.

For conventional rate-constrained NSS algorithm, the PSNR values and the rate values are exactly the same as their corresponding values for the proposed search algorithm. The ANMC values for the conventional algorithm for $N = 3, 4, 5$ step searches are 25, 33, and 41, respectively. The amount of computational savings increases as the search region size gets larger, i.e., as N increases. For example, when $\lambda = 0$, the amount of savings for $N = 3$ is between $20 - 40\%$, whereas it is between $25 - 50\%$ for $N = 5$ case. As λ increases, both the number of bits spent on the MVs and the number of matching evaluations decreases for the proposed algorithm. This makes it suitable for low bit rate video applications. For example, when $\lambda = 50$, the computational complexity of the proposed algorithm is computation wise $25 - 55\%$ more efficient than the conventional rate-constrained NSS algorithm for $N = 3$ and $35 - 70\%$ more efficient for $N = 5$. As the rate constraint increases further, $\lambda = 100$, the computational savings increase to $30 - 70\%$ for the $N = 3$ case, $40 - 80\%$ for the $N = 5$ case.

5. CONCLUSIONS

A new fast N-step search block matching algorithm has been presented for rate-constrained motion estimation. Experimental results show that the performance of the proposed algorithm is equivalent to that of the conventional rate-constrained NSS, with lower computational complexity. The motion vectors are chosen from a search region based on a rate-distortion criterion. The number of block matching calculations is reduced by limiting the matching evaluation positions based on an inequality condition, which is updated whenever a motion vector with lower cost function value is found. As the constraint on the rate increases, the number of matching evaluations needed to find the best motion vector decreases, and the algorithm becomes faster.

6. REFERENCES

[1] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion compensated interframe coding for video conferencing," Proc. Nat. Telecom. Conf. (New Orleans, LA), pp. G5.3.1–5.3.5, December 1981.

[2] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding," IEEE Trans. on Communications, vol. COM-29, pp. 1799–1808, December 1981.

[3] G. Sullivan and R. L. Baker, "Rate-distortion optimized motion compensation for video compression using fixed or variable size blocks," Proc. of Global Communications Conf., pp. 85–90, November 1991.

[4] B. Girod, "Rate-constrained motion estimation," SPIE Proc. Visual Communications and Image Processing, vol. 2308, pp. 1026–1034, 1994.

[5] W. C. Chung, F. Kossentini, and M. J. T. Smith, "An efficient motion estimation technique based on a rate-distortion criterion," Proc. ICASSP'96, vol. 4, pp. 1926–1929, May 1996.

[6] W. Li and E. Salari, "Successive elimination algorithm for motion estimation," IEEE Trans. on Image Processing, vol. 4, pp. 105–107, January 1995.

[7] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," IEEE Trans. on Acoust., Speech, Signal Processing, vol. 36, pp. 1445–1453, September 1988.

[8] ITU-T Recommendation H.261, "Video Codec for Audiovisual Services at $p \times 64$ kbit/s," December 1990.

[9] ITU Telecom. Standardization Sector Study Group 15, "ITU-T Recommendation H.263 Question 2/15," Draft Recommendation H.263, July 1995.