

A* PRUNE MODIFIED ALGORITHM IN VIDEO COMPRESSION

S Verma IIT,
India
sverma@iita.ac.in

Amit Kant Pandit †
SMVDU , India
amit.pandit@smvdu.ac.in†
† Corresponding author

ABSTRACT

Block matching technique is the most significant tool in motion estimation and compensation in video compression. Block matching is either fixed size block matching (FSBM) techniques or variable size block matching (VSBM). Rate distortion optimization problem is related with it. The R-D optimization, NP hard problem, is solved using Lagrange's parameter to find a constrained path, where a given PSNR (distortion) and bit rate is achieved. In this paper, we modify and study the A*Prune algorithm used in QoS routing in network ,to solve the R-D optimization problem of video compression .we cast the R-D optimization problem as KMCSP (K multiple constrained shortest path problem). The modification presented in this paper is applied to DVF and DFD both and are constrained simultaneously to find any optional number of constrained shortest paths.

Keywords: A* PRUNE, VIDEO COMPRESSION, QUAD TREE

1 INTRODUCTION

Block matching is widely used method for stereo vision, visual tracking and video compression. At present, most of the techniques used for block matching in video compression are either fixed size block matching (FSBM) or variable size block matching (VSBM) techniques. In FSBM blocks of only one size are used. When a larger block size is used, the number of motion vectors to be encoded or the motion vector field (MVF or DVF) is low. So less number of bits is required for transmission in a channel. However, it results in a higher prediction error or displaced frame difference (DFD) which affects the quality of reconstructed frame. On the other hand, when a smaller block size is used, the DFD is quite low, but more motion vectors need to be encoded resulting in higher transmission rates. There is Rate distortion (R-D) optimization problem associated with video compression

The optimal solution to the fundamental problem of splitting coding bits between *DVD* and *DVF* is closely related to the size of the block which in turn is dependent on the scene content of video. Variable size block matching (VSBM) provides a better solution of above optimization problem as compared to fixed size block matching (FSBM). The constrained R-D problem is solved using shortest path algorithms like graph search or viterbi algorithm [1]. It is a multiple constraint shortest path problem

(MCSP); universal solution is not possible to arrive at. Solving the R-D using the graph search is shown to be NP- hard [2].Lagrangian bit allocation technique is most popular and widely accepted, for efficient bit allocation at some distortion level. Its popularity is due to its effectiveness and simplicity. It provides an optimized constrained path .The decision is based on minimizing the sum of distortion of block and λ times bits needed to code it, where λ is the Lagrangian parameter.

Lagrangian methods have many limitation and problems. Firstly, we do not have any control on individual contribution of DVF and DFD. Secondly, due to presence of temporal and spatial dependencies of the rate -distortion costs, the complexities increases, when applied to block based hybrid video codec such as H.264/AVC or MPEG-1/2/3/4. Thirdly, it cannot adjust speedily, according to variation of different constrains and bit rate requirement.

Considering, the bandwidth available for transmission or bit rate required as dynamic parameter, since its value may change at any time. Whenever the dynamic parameter is changed, it is expected to call MCSP (multiple constraints shortest path) procedure each time to find best feasible solution. It is time consuming process. To adjust speedily according to the dynamic parameter, we require K-MCSP (K-multiple constraints shortest path) method. This is speedier since we are selecting a feasible path from multiple precomputed paths. We

require a MCSP method which can control contribution of different constraints individually and provide with K paths to choose, according to the variation of dynamic parameter

2. PROBLEM DEFINATION

The RD optimization problem using Lagrangian technique is NP-hard, and optimal solution is not guaranteed. We convert it to KMCSP (K- multiple constrained shortest paths) problem or NP complete. We search k-shortest paths instead of a single shortest path and use the best one of the pre calculated k-shortest paths to predict optimal solution as per the need or value of dynamic parameter.

Consider a network that is represented by a graph $G=(V, E)$, where V is the set of nodes and E is the set of links. Each link $(i,j) \in E$ is associated with R nonnegative and additive constraints values: $W_r(i,j)$, where $r = 1,2,3,\dots,R$.

Given a source node, 's' and a target node 't' and R constraints $C_r(s,t)$, where $r = 1,2,3,\dots, R$. The problem is to find K shortest path from source node s to target node t such that

$$W_r(p(s,t)) \stackrel{\text{def}}{=} \sum_{(i,j) \in p(s,t)} w_r(i,j) \leq C_r(s,t) \quad \forall r \in \{1,2,\dots,R\}. \quad (2)$$

Here, K multiple shortest paths are found, which satisfies the equation 2.

3. PROPOSED SOLUTION

We use an existing k-shortest path algorithm called A* prune algorithm [3], used in QOS routing, as a base algorithm to find the k-shortest paths subject to multiple constraints. We adopt it here for solving DVF-DFD optimization problem and both are constrained simultaneously.

A* prune algorithm gives k - multiple constrained shortest paths between a pair of nodes in a digraph in which each is associated with a several Quality of service metrics. It constructs paths starting at the source and going towards the destination. But at each iteration the algorithm gets rid of all paths that are guaranteed to violate the constraints, thereby keeping only those partial paths that have the potential to be turned into feasible paths, from which the optimal paths are drawn. The choice of which path to be extended first and which path can be pruned depend upon a projected path cost function, which is obtained by adding the cost already incurred to get to an intermediate node to an admissible cost to go the

remaining distance to the destination. The Dijkstra's shortest path algorithm is used to calculate the admissible cost.

Though, the A*prune algorithm successfully calculates K constrained shortest paths and gives optimal solutions, it is not viable to use it in its present form in our application for video compression. The problem is due to its computational complexity. Though it uses inadmissible head path pruning and admissible distances, the complexity still increases significantly with the number of nodes. Since our application in DVF-DFD optimization involves large no of nodes (~ 4096), the algorithm takes very huge time in its original formulation. The time also increases if the costs of the admissible distances (predicted) are very low compared to actual costs.

To make A*prune algorithm feasible for use in video compression. We propose the following adaptation to the original algorithm:

1. Since A* prune algorithm stores list of all possible admissible head paths. We limit the number of head paths to be stored to a pre assigned maximum value. Whenever, a new head path is created, if the limit was already reached, then the new head path displaces an existing head path with the maximum cost and takes its place. In this way we limit the size of Path list.
2. The existing algorithm uses a linear function to calculate the projected cost. However the rate-distortion curve between the DVF and the DFD is a decreasing function. It is proposed to use non- linear weight function. Non -linear weight function converges on the solution quickly compared to linear weight function. This reduces the complexities of the algorithm and the added advantage is that it tries to find a solution towards the center of RD curve. This helps in equally distributed (almost in the ratio of constraints) bit allocation between DVF and DFD. We use the non-linear weight function given in [2].

The equation (3) describes the non linear weight function:

$$\max(C(P)/\Delta_c, D(P)/\Delta_d) \quad (3)$$

Where $C(P)$ gives the path cost for DVF, $D(P)$ gives the path cost for DFD and Δ_c and Δ_d gives the maximum constraints for DVF and DFD respectively

4. IMPLEMENTATION AND SIMULATION DETAIL

The proposed approach was simulated on

MATLAB7.0 platform. In the first step we calculate the of Displacement vector field (DVF) and displaced frame difference (DFD) for each of the block sizes 16 x 16, 8 x 8 and 4 x4. The motion estimation is done using exhaustive search approach. The resulting motion vectors were coded using differential pulse code modulation (DPCM) technique. The quantized DFD values are coded using Huffman entropy coding. We calculated the PSNR values for each of the block sizes.

We used an adjacency matrix representation to construct the resulting graph structure, DVF and DFD values as the two link metrics for each link. The DVF and DFD values for each of the block sizes are populated in the graph in the order of Hilbert scan. This ensures that the quad-tree structure is maintained in the graph, i.e. when a macro block is divided into four smaller sized blocks, the metrics of the smaller sized blocks are just below that of the larger macro block. Hilbert scan order also ensures that the blocks are scanned in such ways that for each block, both its predecessor and successor share an edge with a block.

Once the graph structure is constructed we run the proposed k-multiple constrained shortest path algorithm over the graph. We compute the number of bits required for DFD and DVF for each of the k-shortest paths. We also compute the corresponding PSNR values. We select the best of the shortest paths and use it to reconstruct our frame.

We used mother and daughter frames for simulation. Each frame is clipped to be of size 256 x 256. We initially predict the second frame from the first frame. Thereafter, each of the subsequent frames will be predicted from the previously reconstructed frame. Table: 1 shows the values of different parameters for different block size of frame-2. Frame 1 is the reference frame.

Table 1: DVF-DFD values for frame 2

| block size | 16 | 8 | 4 |
|--------------|--------|--------|-------|
| MVF | 512 | 3944 | 16380 |
| DFD | 5082 | 3749 | 2050 |
| total | 5594 | 7693 | 18430 |
| PSNR | 38.467 | 38.539 | 38.71 |

Now we apply our modified algorithm with following constraints:

Maximum list size: 10;

K= 5;

MVF constraint : 3000;

DFD constraint : 4000;

Here K refers to the number of shortest paths, and maximum list size is the maximum number of head

paths that can be stored at a time. Table-2 list the various optimized path with different parameters

Here no. of nodes is nodes used in resulting quad tree. It is clear from above that path 2 is best with respect to PSNR and constrained bit rate. Out of five optimal paths found with the given constraints.

Now the frame-3 is predicted from the reconstructed frame 2 .The frame-3 has following parameter with respect to different block sizes. (Table :3)

Table 2: MVF-DFD values for K paths for frame2

| K | 1 | 2 | 3 | 4 | 5 |
|-----------------|-------|-------|-------|-------|-------|
| No.nodes | 923 | 926 | 923 | 940 | 925 |
| MVF | 2372 | 2370 | 2378 | 2380 | 2374 |
| DFD | 3406 | 3408 | 3414 | 3414 | 3424 |
| total | 5778 | 5778 | 5792 | 5794 | 5798 |
| PSNR | 39.56 | 39.57 | 39.55 | 39.56 | 39.56 |

Table 3: MVF-DFD values for frame 3

| Block size | 16 | 8 | 4 |
|--------------|------|-------|-------|
| MVF | 521 | 5090 | 20022 |
| DFD | 8658 | 6452 | 3118 |
| total | 9179 | 11542 | 23140 |
| PSNR | 32 | 32.06 | 31.99 |

Now we calculate the parameters with our algorithm with following constraints

Maximum list size = 10;

K = 5;

MVF constraint = 1000

DFD constraint = 8000; Count = 3232

The resulting paths are as shown in Table :4

Table 4: MVF-DFD values for K paths for frame3

| K | 1 | 2 | 3 | 4 | 5 |
|--------------|-------|-------|-------|------|-------|
| nonodes | 392 | 392 | 392 | 390 | 392 |
| MVF | 948 | 950 | 950 | 948 | 951 |
| DFD | 7960 | 7964 | 7972 | 7970 | 7976 |
| total | 8908 | 8914 | 8922 | 8918 | 8927 |
| PSNR | 32.55 | 32.52 | 32.59 | 32.6 | 32.53 |

Other frames are predicted in the same way. The figure-1 shows the rate obtained for different frames at different block levels. Figure -2 shows the five

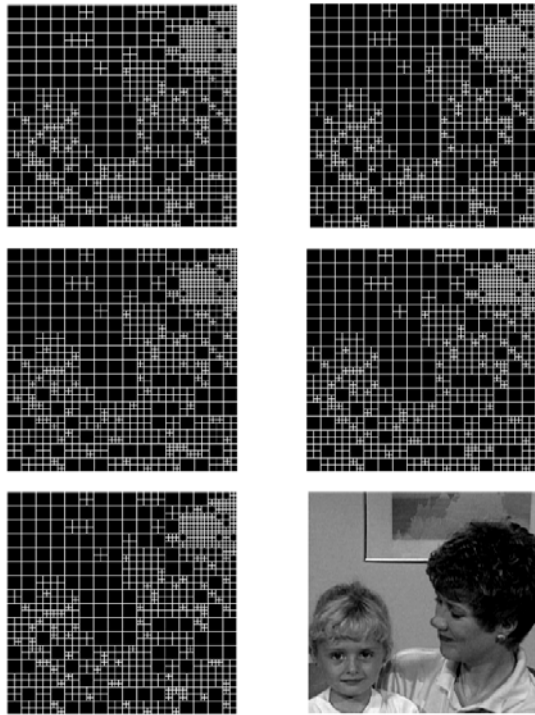


Figure 2: Quad tree structure for five calculated shortest path for frame 2

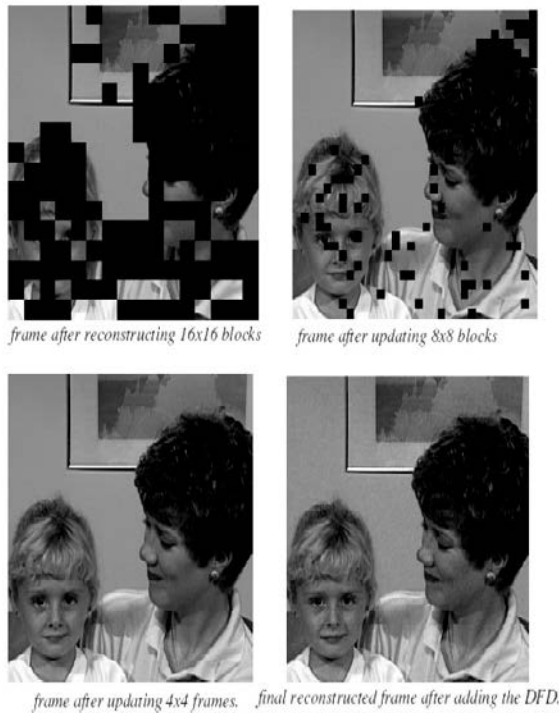


Figure 3: Stages of reconstruction for frame 2

different shortest path quad tree structures obtained. Figure-3 shows the stages of reconstruction of frame 2 after adding blocks of different sizes and there DFD, obtained by selecting the best path from different path calculated.

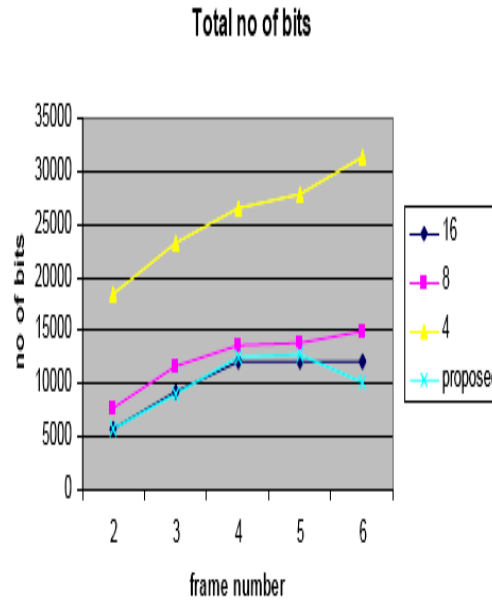


Figure 1: comparison of total bits for encoding

5. CONCLUSION:

Variable size block matching technique was developed based on constrained shortest path algorithm, which gives lower overall bit rates, at the same time satisfying and taking into account both the DVF and the DFD constraints simultaneously. The total allocation of bits was comparable to that of block size 16, which was significantly lower than the rates for other two blocks. Since, to reduce complexities the algorithm is designed sub optimally, and does not guarantee to give best possible solution. Still, depending on the requirement of dynamic parameter, one of the different constrained paths obtained can be selected.

6. REFERENCE

- [1] G.M. Schuster and A.K. Katsaggelos, An optimal quad-tree based motion estimation and motion compensated interpolation scheme for video compression, *IEEE Transactions on Image Processing*, Vol. 7, No. 11, November 1998
- [2] Hans De Neve and Piet Van Mieghem, "A multiple quality of service routing algorithm for PNNP", *Proceedings 1998 IEEE ATM Workshop*, May 26-29, Fairfax, VA, USA, pp.324-328, 1998.
- [3] Gang Liu, K. G. Ramakrishnan, "A*Prune: An Algorithm for Finding K Shortest Paths Subject to Multiple Constraints", *0-7803-7016-3, IEEE INFOCOM*, 2001