GRAPH THEORETICAL OPTIMIZATION OF PREDICTION STRUCTURE IN MULTIVIEW VIDEO CODING

Je-Won Kang[†], *Suk-Hee Cho*[‡], *Nam-Ho Hur*[‡], *Chang-Su Kim*[§], *Sang-Uk Lee*[†]

[†] Signal Processing Lab., School of Electrical Engineering and INMC,
Seoul National University, Korea
[‡] Electronics and Telecommunications Research Institute, Daejeon, Korea
[§] School of Electrical Engineering, Korea University, Seoul, Korea

ABSTRACT

An algorithm to construct the optimal prediction structure in multiview video coding (MVC) is proposed in this work. We employ the graph theory as a framework. By considering each frame as a vertex and the motion compensation or disparity compensation as an edge, we represent a prediction structure as a spanning tree. Then, we obtain the optimal structure by finding the minimum spanning tree using the Prim's algorithm. Simulation results demonstrate that the proposed algorithm provides about 0.2-0.4 dB better PSNR performance than the conventional prediction structure, and about 1.5 dB better performance than the simulcast.

Index Terms— Multiview video coding, prediction structure, graph theory, minimum spanning tree.

1. INTRODUCTION

A multiview video sequence is acquired by multiple cameras simultaneously, which are located at different view points. Using the images from multiple viewpoints, 3D information can be reconstructed, even if not perfectly. Thus, multiview videos can be used in various applications. For example, in free-view video applications, users can switch their view points interactively, and in 3D-TV applications, users can perceive objects or scenes with depth [1]. However, multiview video sequences often require a huge amount of data, which is an obstacle to their widespread use. Therefore, it is essential to develop an efficient MVC algorithm to store and transmit multiview video data.

ITU-T and ISO/IEC are extending their joint video coding standard H.264/AVC to compress multiview video sequences. In most video coding standards for single view sequences, motion compensated prediction is employed to exploit high temporal correlations. In addition to the time axis, a multiview video sequence has the camera axis for different views. Thus, it has two dimensional structure of frames as shown in Fig. 1. Therefore, in MVC, it is important to exploit inter-view correlations as well as temporal correlations.

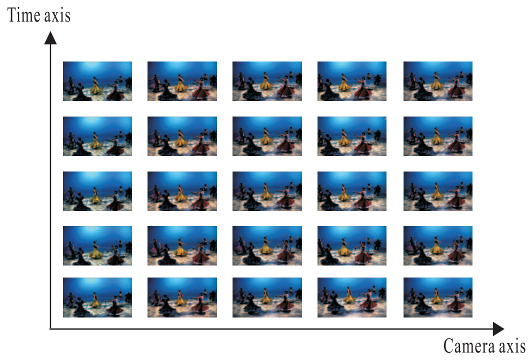

Fig. 1. A two dimensional structure of frames in a multiview video sequence (KDDI flamenco sequence).

Various schemes have been proposed to achieve a high coding gain using inter-view correlations. In the MPEG-2 multiview profile [2], the stereoscopic coding was proposed, which uses both disparity compensated prediction in addition to motion compensated prediction. In [3], the disparity estimation with a mesh-based segmentation scheme was proposed. Also, to optimize the quantization parameters in MVC, the Viterbi algorithm was developed in [4]. Recently, many research groups have proposed various prediction structures for multiview video sequences empirically [5, 6].

In this work, we propose an algorithm to obtain the optimal MVC prediction structure. In the single view video coding, it was shown that the rate-distortion performance is significantly affected by the choice of the prediction structure [7]. In MVC, we have a higher degree of freedom in selecting the prediction structure, and thus there is a bigger room for performance improvement. We adopt the graph theory to find the optimal structure. Specifically, we show that

[†] This research was supported by the MIC, Korea, under the ITRC (Information Technology Research Center) program supervised by the IITA.

[§] The research of C.-S. Kim was supported by the MIC (Ministry of Information and Communication), Korea, under the ITFSIP (IT Foreign Specialist Inviting Program) supervised by the IITA (Institute of Information Technology Assessment).

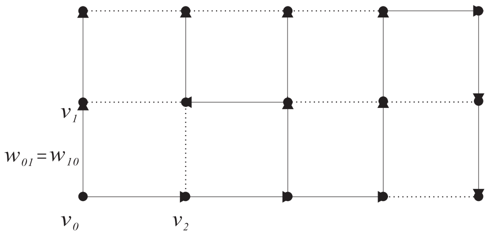

Fig. 2. A spanning tree, which represents a prediction structure in MVC

the optimal structure can be found by solving the minimum spanning tree problem. Simulation results demonstrate that the proposed algorithm provides better rate-distortion performance than the conventional prediction structures.

2. GRAPH THEORETICAL DESIGN OF PREDICTION STRUCTURE

We represent the frame structure in MVC as a graph, and formulate the construction of the optimal prediction structure as the minimum spanning tree problem in the graph theory [8].

A graph ***G*** is defined as (***V***, ***E***), where ***V*** is a set of vertices and ***E*** is a set of edges. An edge $e_{ij} = (v_i, v_j)$ in ***E*** connects two vertices v_i and v_j in ***V***. Each edge is labeled with a weight w_{ij}. In this work, a vertex represents a frame, and two frames are connected by an edge if they are adjacent in the time or camera axis. Also, the edge is labeled by the difference between the corresponding frames, which will be defined later.

From the graph ***G***, we form a spanning tree, which connects all vertices with the smallest number of edges. Then, the root of the tree is encoded in the intra mode. Except the root vertex, each vertex is encoded in the inter mode by employing its parent as the reference frame. For example, in Fig. 2, the solid lines depict a spanning tree. If v_0 is selected as the root, it is encoded in the intra mode, while v_1 and v_2 are encoded using v_0 as the reference frame, respectively.

Given a graph, there are a large number of possible spanning trees. In this work, the objective is to find the optimal prediction structure. The efficiency of the predictive coding becomes higher, as the difference between a frame and its reference frame gets smaller. Therefore, we choose the minimum spanning tree that has the least sum of weights (or frame differences).

We obtain the minimum spanning tree using the Prim's algorithm [9] as follows. We divide ***G*** into the set of coded frames ***S*** and its complement ***S***^{c}. Initially, an arbitrary frame is selected as the single vertex in ***S***. Then, at each step, a vertex in ***S***^{c}, which is adjacent to one of the vertices in ***S***, is selected and connected to ***S***. This is repeated until ***S*** has all vertices in ***V***. The connecting edges from all the steps form the spanning tree.

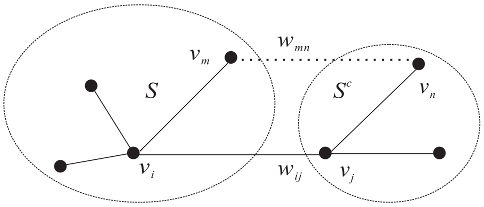

Fig. 3. Illustration of the optimal path condition for constructing the minimum spanning tree.

To guarantee that the spanning tree has the least sum of weights among all possible spanning trees, at each step, the Prim's algorithm selects the connecting edge based on the path optimal condition [9]. For example, in Fig. 3, the edge e_{ij} is selected if

$$w_{ij} \leq w_{mn} \tag{1}$$

for all edges e_{mn} that connect ***S*** and ***S***^{c}. The Prim's algorithm proceeds in a greedy and successive way, but it provides the minimum spanning tree as a global minimum. Moreover, it has a much lower complexity than the exhaustive search.

3. DESIGN OF CODEC

In this section, we describe several issues, related to the implementation of the proposed MVC algorithm.

First, we do not consider the multiple reference frame mode, and encode each frame as an I (intra) or P (inter) mode only. Second, we use an undirected graph when applying the Prim's minimum spanning tree algorithm. Third, the weight of an edge in the graph, which describes the difference between two frames, is defined based on the motion compensated sum of squared differences (MCSSD). Let F_i and F_j denote the frames corresponding to vertices v_i and v_j, respectively. Then, MCSSD from F_i to F_j is given by

$$\text{MCSSD}(F_i, F_j) = \sum_{x,y} (F_i(x + v_x, y + v_y) - F_j(x, y))^2,$$

where (v_x, v_y) is the motion vector of a pixel (x, y), which is computed blockwise using the block matching algorithm. Then, the undirected weight w_{ij} is defined as the average of the two directed MCSSD's

$$w_{ij} = \frac{\text{MCSSD}(F_i, F_j) + \text{MCSSD}(F_j, F_i)}{2}.$$

Before generating the minimum spanning tree, a vertex needs to be selected as the root, which is encoded as an I frame. In general, as the I frame is more similar to the other frames, it can be used more effectively as the reference for encoding the other frames. Therefore, we select F_i as the I frame, if

$$\sum_{k}^{N-1} MCSSD(F_i, F_k) \leq \sum_{k}^{N-1} MCSSD(F_j, F_k) \text{ for all } j,$$

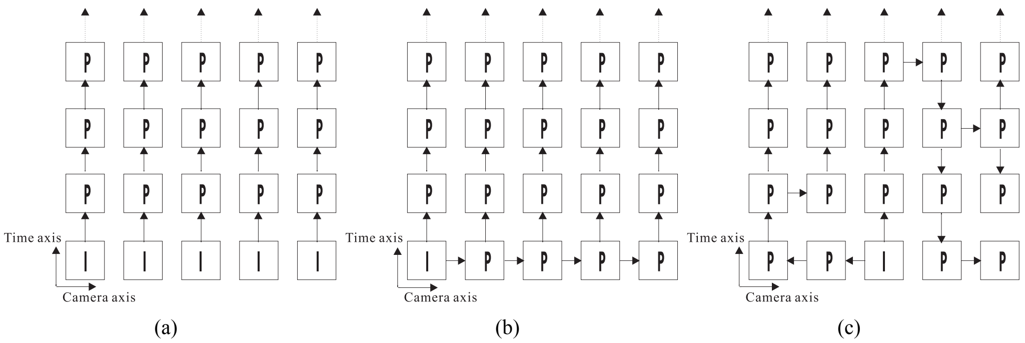

Fig. 4. Prediction structures for MVC: (a) simulcast structure, (b) anchor structure, and (c) irregular structure.

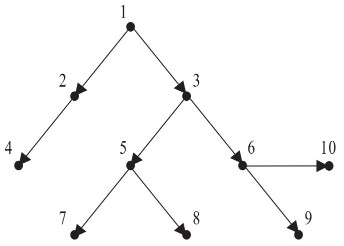

Fig. 5. After encoding frame 1, we should encode frame 2 or 3. In this work, we select frame 2 to reduce the maximum memory requirement.

where N is the number of frames at a temporal base line in the graph. Then, the other frames are encoded as P-frames, according to the spanning tree.

The proposed algorithm generates a tree of irregular shape, as shown in the example of Fig. 4 (c). When adopting this prediction structure, we should select the coding order carefully to reduce the amount of memory, which is required to store frames at both the encoder and the decoder. Fig. 5 shows an example. Let us assume that frame 1 has been encoded already, and we should select its left child 2 or right child 3 as the next frame to be encoded.

- Suppose that frame 2 is selected first. After encoding frame 2, we encode frame 4 and free the memory for frames 2 and 4. Then, we encode vertex 3, and free the memory for vertex 1 that is no more necessary. In this way, the maximum memory requirement is three frames. Note that we need to store frames 1, 2 and 4, when we encode frame 4.
- Suppose that vertex 3 is selected first. In this case, it can be shown that the maximum memory requirement is four frames.

Thus, it is advantageous to encode frame 2 first. This is because the subtree with the root at vertex 2 has a shorter depth than the subtree with the root at vertex 3. In general, after encoding the current frame, we compare the depths of the subtrees of the children. Then, the child with the minimum depth subtree is encoded next. Also, the memory for a frame is freed, when all its children are encoded.

Table 1. Properties of the test sequence and the encoding parameters.

Sequence name	race1	No. of frames	400
Frame rate (fps)	7.5	GOP size	10
No. of cameras	8	Camera array	1-D parallel

Last, since the proposed algorithm uses the adaptive prediction structure, the prediction structure itself should be encoded and transmitted to the decoder. We encode this information using the reserved syntax of H.264/AVC standard [10]. However, the size of this additional information is negligible as compared with the whole bitstream.

4. SIMULATION RESULTS

We evaluate the performance of the proposed algorithm on various multiview video sequences. In this paper, we present the results on one of the sequences, called "race1." Table 1 summarizes the properties of the sequence and the encoding parameters. For comparison, we use two prediction structures: the simulcast structure in Fig. 4 (a), and the anchor structure in Fig. 4 (b).

Fig. 6 compares the PSNR plot of the proposed algorithm with that of the anchor algorithm. The bit rate for the proposed algorithm (363 Kbps) is slightly higher than that for the anchor algorithm (354 Kbps). The test sequence has very fast motions in the earlier part, while the fast moving objects move away from the camera in the later part. Thus, the earlier part exhibits lower temporal correlations. In such a case, the propose algorithm adapts the prediction structure to use interview predictions, and provides better performance than the anchor algorithm, which uses the temporal predictions only except at the start of the sequence. On the other hand, the later part has higher temporal correlations, and the anchor algo-

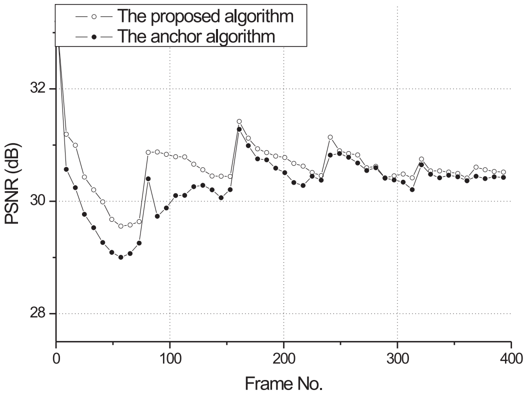

Fig. 6. The comparison of the PSNR performances on the "race1" sequence

rithm and the proposed algorithm yield similar performances.

We present the R-D performances in Fig. 7. We see that the proposed algorithm provides about 0.2-0.4dB better performance than the anchor algorithm, and about 1.5 dB better performance than the simulcast algorithm. These simulation results indicate that the proposed algorithm is an efficient technique for the compression of multiview video sequences.

5. CONCLUSION

In this work, we proposed an efficient compression algorithm for multiview videos. By considering each frame as a vertex and the motion compensated prediction or the disparity compensated prediction as an edge, we converted the optimization of the prediction structure to the minimum spanning tree problem. Simulation results demonstrated that the proposed algorithm provides about 0.2-0.4 dB better PSNR performance than the anchor structure.

As a future research topic, we plan to include the hierarchical B prediction mode in H.264/AVC to the graph theoretical framework. It is expected that the rate-distortion performance will be improved further, if we can select the optimal prediction structure using I, P and B modes.

6. REFERENCES

[1] A. Smolic, K. Mueller, P. Merkle, C. Fehn, P. Kauff, P. Eisert, and T. Wiegand, "3D video and free viewpoint video - technologies, applications and MPEG standards," in *Proc. ICME*, July 2006, pp. 2161–2164.

[2] ISO/IEC/JTC1/SC29/WG11, "ISO/IEC 13818-2 AMD 3: MPEG-2 multiview profile," 1996.

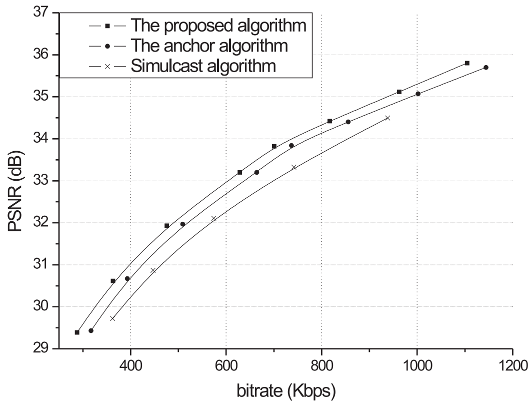

Fig. 7. The rate-distortion performances of the proposed algorithm, the anchor algorithm, and the simulcast algorithm on the "race1" sequence

[3] R.-S. Wang and Y. Wang, "Multiview video sequence analysis, compression, and virtual viewpoint synthesis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 3, pp. 397–410, Apr. 2000.

[4] J. H. Kim, J. Garcia, and A. Orthega, "Dependent bit allocation in multiview video coding," in *Proc. ICIP*, Sept. 2005, vol. 2, pp. 293–296.

[5] ISO/IEC/JTC1/SC29/WG11, "Description of core experiments in MVC," Doc. N7798, Jan. 2006.

[6] B. Bai, P. Boulanger, and J. Harms, "An efficient multiview video compression scheme," in *Proc. ICME*, July 2005, pp. 836–839.

[7] J. Lee and B. W. Dickinson, "Rate-distortion optimized frame type selection for MPEG encoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 501–510, June 1997.

[8] J. Gross and J. Yellen, *Graph theory and its applications*, CRC Press, 1999.

[9] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows - Thoery, Algorithms, and Applications*, Prentice Hall.

[10] ITU-T and ISO/IEC JTC1, "Advanced video coding for generic audio-visual services," ITU-T Recommendation H.264 and ISO/IEC 14496-10 AVC, 2003.