RESEARCH
Open Access

Early disparity estimation skipping for multi-view video coding

Jungdong Seo and Kwanhoon Sohn[*]

Abstract

Multi-view video (MVV) and multi-view plus depth video became typical formats for 3D video systems which offer more realistic visual effects for consumers. The standard for encoding of MVV, known as multi-view video coding (MVC), was produced by ISO/IEC and ITU-T and has exhibited significant compression efficiency for MVV signals. However, the high computational complexity of MVC leads to excessive power consumption and limits its applications. In this study, new fast algorithms for MVC are proposed by skipping disparity estimation prediction. According to the characteristics of the motion and disparity vectors, unnecessary disparity estimation is skipped based on the amount of motion that can be measured through the use of motion activity and SKIP mode on the temporal axis. Experimental results showed that by skipping the disparity estimation, the proposed method reduced the encoding computational complexity by nearly 73% for inter-view prediction and 38% in the overall encoding process of B-views. In addition, no noticeable visual quality degradation was observed.

Keywords: multi-view video, multi-view video coding (MVC), inter-view prediction, early skipping algorithm, motion activity.

1. Introduction

Recently, 3D video has received increased attention because of the great success of 3D movies. Other mediums such as three-dimensional TV (3DTV) and 3D digital multimedia broadcasting (3D DMB) are also representative applications of 3D video [1]. 3DTV provides vivid and realistic scenes to users with its feeling of depth [2,3]. The principles of 3DTV are based on a stereoscopic vision system. When left and right views of stereoscopic video are shown in a user's left and right eyes, respectively, users can perceive a feeling of depth due to binocular parallax. A 3DTV system contains not only stereoscopic video, but also a multi-directional 3D scene by multi-view. 3D DMB is a broadcasting system which offers 3D video to users in mobile circumstances [4]. Because the system can present more realistic video and information than conventional DMB, it is expected to be widely used in e-commerce and news media.

Among the video formats that support 3DTV and 3D DMB, multi-view video (MVV) and multi-view plus depth video (MVD) are typical formats. MVV is simultaneously acquired by two or more cameras placed at a certain distance from each other so that its viewpoint number is fixed as the number of cameras. MVV is very simple and intuitive, but it has difficulties in responding to various types of 3D displays because it provides limited viewpoints. On the other hand, MVD contains multi-view color and depth video which are used to synthesize virtual views in the decoder side. The MVD format can be applied to various 3D displays because of virtual view synthesis, which leads to a high degree of computational complexity for the decoder. However, MVV and MVD have one aspect in common: they receive an enormous amount of data from multiple viewpoints.

Multi-view video coding (MVC) is a video coding standard that reduces a large amount of MVV data. It was developed by the Joint Video Team (JVT), an organization jointly created by the Video Coding Expert Group and the Motion Picture Expert Group (MPEG) [5]. In general, the computational complexity of a video codec for MVV proportionally increases with the number of views. However, the complexity of MVC is far beyond the independent encoding of each view because of the exhaustive search for inter-view prediction, a key

* Correspondence: khsohn@yonsei.ac.kr
C129, School of EE, Yonsei University, 134 Shinchon-dong Seodaemoon-gu, Seoul, Korea

Springer

technique in MVC, and it leads to excessive power consumption and works as an obstacle for practical use. Therefore, it is necessary to decrease the computational complexity of MVC.

Some research has been performed to reduce the computational burden of MVC [6-10]. Kim et al. [6] proposed an adaptive search range control algorithm for motion and disparity estimation. The researchers reduced the computation time for motion and disparity estimation by controlling the search ranges. The reliability of the predicted vectors for the current block was used to control the range, and it was calculated based on the difference between two predicted vectors by two ways. Since the search ranges of motion and disparity estimation are reduced, the algorithm showed time saving performance of about 70%. However, the algorithm needs camera parameters for vector prediction, and it is difficult to apply to MVC because it was based on the different coding structure from the structure of MVC. Ding et al. [7] proposed a fast motion estimation algorithm that finds an initial motion vector based on the macroblock partitioning information of the reference view. The algorithm then refines the motion vector in a small search range near the initial motion vector. This approach simplified the mode selection process and reduced the search range of the motion estimation so that the overall processing time of MVC was reduced. However, there are many conventional fast mode decision and search range adaptation algorithms for motion estimation, Ding's algorithm overlaps with some of these conventional algorithms. Peng et al. [8] presented a hybrid fast macroblock mode selection algorithm for MVC. In base view, the algorithm partly stops the block mode selection process through multi-thresholding of the rate-distortion cost. In other views, the macroblock mode can be predicted from the frames of neighboring views. The fast algorithm decreased the number of cases in the mode selection process. It covered most of the pictures in the coding structure of MVC, but it is difficult to implement into hardware due to the complicated construction of the algorithm. Li et al. [9] proposed a fast disparity and motion estimation technique based on multi-view geometry. The algorithm reduced the search range for disparity estimation based on a correlation between neighboring cameras and selectively skipped the motion estimation process through the use of the relationship between the motion and disparity vectors. Li et al.'s algorithm showed a time saving performance of about 65%, but bit rate was increased by about 3%. Shen et al. [10] proposed reduced mode selection, search range adaptation, and view-adaptive disparity estimation (VDAE) for MVC. They were based on the mode complexity and motion homogeneity of the neighboring views. The algorithms do not overlap each other

and show relatively high performance, but the reliability of the algorithms is not high because they use a global disparity vector to obtain the information of the encoded neighboring views.

In this article, new fast algorithms that skip the disparity estimation process for inter-view prediction without the information of neighboring views are proposed. While disparity vectors are rarely selected for the results of MVC, the disparity estimation process consumes much time. The proposed algorithms reduce the encoding time of MVC by skipping the unnecessary disparity estimation process. For skipping of disparity estimation, the motion activity and SKIP mode are introduced. Since these methods are not influenced by multi-view geometry, the proposed algorithms can provide consistent results regardless of the camera arrangement. It can also be applied with the conventional fast algorithms of other types such as search range adaptation and early mode determination, because the proposed algorithms do not theoretically overlap with the conventional fast algorithms.

This article is organized as follows. An overview of the MVC prediction structure is presented in Section 2. In Section 3, two new methods for skipping disparity estimation are proposed. Experimental results and analysis are given in Section 4 and conclusions are followed in Section 5.

2. Overview of MVC prediction structure

MVC reference software was chosen at the 75th MPEG meeting, and it was based on version 3.5 of the JSVM (Joint Scalable Video Model) software [11]. It used reordered MVV as a 2D sequence because conventional 2D video codec was used for MVC without syntax modification. After the MVC standardization activity was moved to JVT, a new version of MVC software considered the characteristics of MVV was developed. It included two special properties such as coding structure for inter-view prediction and hierarchical B-picture structure.

Figure 1 shows the acquisition and coding structure for MVV. The MVV is simultaneously acquired by two or more cameras as shown in Figure 1a, and then it is encoded by the MVC software with coding structure as shown in Figure 1b. The MVC standard adopted inter-view prediction to remove the redundancy among neighboring views. The full inter-view prediction is applied to every other view, i.e., S1 and S3 in Figure 1b, because excessive estimation process for the inter-view prediction causes high computational complexity. However, for all the pictures at T0 and T8, the inter-view prediction is performed regardless of view order. These pictures are called 'anchor picture,' and are used as stamps of synchronization and random access. According to the picture type of the anchor pictures, a view
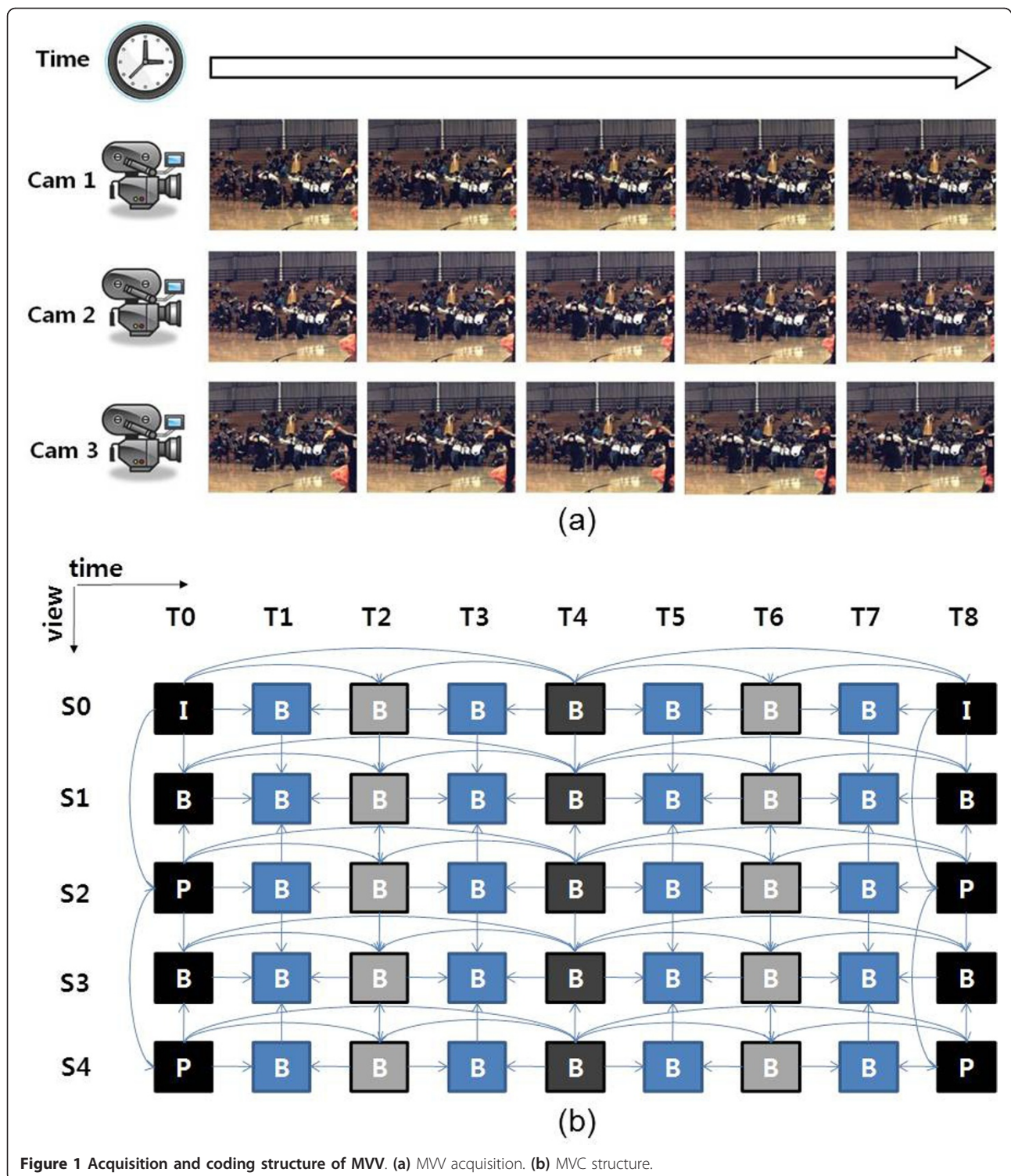
Figure 1 Acquisition and coding structure of MVV. (a) MVV acquisition. **(b)** MVC structure.

type is determined. For instance, the view type of S0 is I-view, and the view types of S2 and S1 are P-view (predictive coded view) and B-view (bi-directional predictive coded view), respectively.

A hierarchical B-picture structure was adopted for coding performance regardless of the characteristics of the MVV. In contrast to the conventional structure such as 'IPPP' or 'IBBP', the predicted picture has its own

hierarchy level in the hierarchical B-picture structure and the pictures are encoded sequentially according to their level [12]. This concept provides the benefit of increased flexibility at the picture/sequence level through the availability of the multiple reference picture technique [13]. In general, the maximum hierarchy level is four because of the encoding complexity. For the multi-view test sequences, the hierarchical structure is determined based on the group of picture (GOP) length, as shown in Figure 2.

The inter-view prediction technique enhances the coding efficiency of MVC, but leads to a high degree of computational complexity. The technique yielded coding gains of up to 3.2 dB and an average coding gain of 1.5 dB [14]. However, its complexity was much higher when compared to that of view-independent coding due to the exhaustive search for inter-view prediction. The computational complexity for B-views was twice that of a single view coding. The proportion of motion and disparity estimation time of the B-views are shown in Table 1. Disparity estimation occupies almost one half of the total processing time. Nevertheless, disparity vectors are rarely selected. Merkle et al. [14] found that the proportion of selected inter-view prediction is about 13% in MVC for several sets of multi-view test data. Because temporal prediction was selected for the inter-coding, the inter-view prediction process is unnecessary for the other blocks. Thus, the computational complexity of MVC can be reduced by skipping the unnecessary process of inter-view prediction.

3. Fast algorithm for MVC by skipping inter-view prediction

To determine the unnecessary disparity estimation process, we observed motion and disparity estimation processes. It was found that motion vectors are most likely to be selected in static or slow motion areas and disparity vectors are used only in fast motion areas. Table 2 shows the average magnitude values of motion vectors for two regions, encoded by temporal prediction and inter-view prediction. The average magnitude values of the motion vectors in blocks encoded by inter-view prediction are far larger than the values in the temporally predicted blocks as shown in Table 2. This is due to the

characteristics of temporal and inter-view prediction. In general, the performance of temporal prediction is superior to inter-view prediction, because inter-view prediction has inherent disadvantages such as a perspective effect and a color imbalance problem between views. The same objects may be seen as different shapes and pixel values in each view. However, temporal prediction exhibits poor performance in fast motion areas, because the correlation between the previous picture and the current picture decreases and its rate-distortion cost increases due to large motion vectors. Thus, by skipping the disparity estimation process in motionless or slow motion areas, the computational complexity of MVC can be decreased with no degradation in video quality. To measure the amount of motion and to simplify inter-view prediction, two methods are proposed in the following sections.

3.1. Motion activity method

The first proposed method to determine the motionless or slow motion area is based on a thresholding technique. Motion activity is defined to represent the amount of motion in a macroblock as follows:

$$\text{Motion Activity} = |MV_x| + |MV_y| \tag{1}$$

where MV_x and MV_y are the horizontal and vertical components of the motion vector, respectively. There are some vision techniques used to accurately measure the amount of motion, such as optical flow and feature tracking. However, these techniques have a high degree of computational complexity and are not suitable for a codec. The information from motion vectors is appropriate for measuring the amount of motion in a video codec, because the vectors roughly reflect the amount of motion in a block. The information of the motion vectors can be obtained after the motion estimation. The motion activity has a physical meaning similar to the magnitude of the motion vectors and exhibits more hardware-friendly characteristics than the magnitude. Thus, the motion activity was used as a variable for the thresholding.

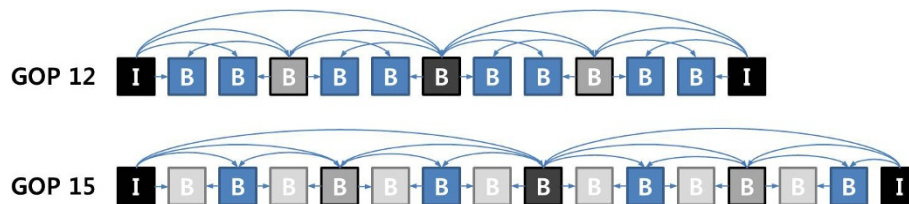To formalize the calculation process of the motion activity in various macroblock partition modes, an 8×8

Figure 2 Hierarchical B-picture structures for multi-view sequences.

Table 1 Processing time ratio of motion and disparity estimation in B-view

Sequence	Motion estimation processing time (%)	Disparity estimation processing time (%)
Ballroom	52.02	46.98
Breakdancers	50.30	48.84
Exit	52.41	46.71
Uli	48.10	50.76
Average	50.71	48.32

block was selected as a basic unit for the motion activity. In the case of inter 16×16 block mode, the motion activity is calculated by multiplying the sum of the motion vector components by 4, which is the number of 8×8 blocks in the 16×16 block. Since intra mode is usually selected in very fast motion areas or new object areas, the motion activity is defined as infinite in intra mode. Equations for calculating the motion activity for all of the cases in this study are given in Table 3.

It is possible to select various threshold values of the motion activity based on the video data and quantization parameters (QPs) for performance. However, in this study, only one threshold value was selected for every case due to practical use. To observe the motion activity values under various conditions, MVC was performed in four QPs and eight video sequences. The average motion activity values for temporal prediction and inter-view prediction are shown in Table 4. The average motion activity of the temporally predicted region was 17.3 for the test sequences. Based on this result, the proposed algorithm was performed with threshold values of the motion activity in the range of 10 to 30. The peak signal-to-noise ratio (PSNR) values were similar to all the threshold values we used. However, as the threshold value increased, the bit rate increased and the processing time decreased, as shown in Table 5. From the results, 22 was selected as the threshold, because it maintained a bit rate increment of about 0.5% and time was sufficiently saved.

The overall process of the disparity estimation skipping algorithm based on the motion activity is shown in Figure 3. At first, only motion estimation is performed for a target macroblock for the temporal prediction. The results of the motion estimation include motion information such as the block mode and the motion vectors.

This motion activity is calculated based on the information. Then, the motion activity is compared with an empirically predefined threshold value to determine whether disparity estimation was performed or not. If the motion activity is less than the threshold, disparity estimation is skipped, because the macroblock is located in a motionless or slow motion area. Otherwise, disparity estimation will be performed. When disparity estimation is performed for the macroblock, the rate-distortion cost of the disparity estimation is compared with that of motion estimation in order to select the best block mode and vectors.

3.2. Method used by SKIP mode in temporal prediction

Another method to determine skipping of the disparity estimation process refers to the selection of SKIP mode in temporal prediction. SKIP mode is one of the inter prediction modes in H.264/AVC, and a 1-bit flag is transmitted for the mode. The blocks encoded as the mode are reconstructed by reference indexes and motion vectors derived by causal blocks. SKIP mode is usually selected in stationary regions or slow motion regions. Figure 4 shows the distribution of selected SKIP, inter and intra modes in a B-picture. The moving object (the dancer in the middle of the scene) is encoded as intra or inter mode, but the background and static objects are encoded as SKIP mode. Table 6 also shows the motion activity values in blocks encoded as SKIP mode and non-SKIP modes. In case of SKIP mode, the motion activity values are smaller than the values of blocks encoded as non-SKIP modes. It presents that blocks selected as SKIP mode are located in slow motion regions or stationary regions. Thus, SKIP mode can be used to determine whether disparity estimation is skipped or not.

Table 2 Average magnitude of the motion vectors in two regions

Sequence	Average magnitude of MVs in blocks encoded by temporal prediction	Average magnitude of MVs in blocks encoded by inter-view prediction
Ballroom	3.7	44.5
Breakdancers	9.1	46.6
Exit	2.3	52.7
Uli	4.5	64.6
Average	4.9	52.1

Table 3 Motion activity in block modes

Block mode	Motion activity calculation
16×16	$(|MV_x| + |MV_y|) \times 4$
16×8	$(|MV_x| + |MV_y|) \times 2$
8×16	$(|MV_x| + |MV_y|) \times 2$
8×8	$|MV_x| + |MV_y|$
8×4	$(|MV_x| + |MV_y|)/2$
4×8	$(|MV_x| + |MV_y|)/2$
4×4	$(|MV_x| + |MV_y|)/4$
Intra	Infinite

The overall process of the proposed algorithm based on SKIP mode is shown in Figure 5. At first, motion estimation is performed in the temporal axis. If SKIP mode is employed for the macroblock, the disparity

Table 4 Average value of the motion activity in B-view

QP	Sequence	MA of temporal prediction	MA of inter-view prediction
24	Ballroom	13.46	180.69
28		14.39	175.03
32		14.23	161.81
36		13.98	162.74
24	Breakdancers	46.60	208.93
28		40.08	202.62
32		30.89	174.56
36		21.57	140.13
24	Exit	10.13	225.86
28		9.34	229.23
32		9.23	267.85
36		7.86	185.43
24	Uli	17.37	220.42
28		16.49	202.50
32		15.80	227.07
36		13.10	169.39
24	Balloons	15.25	159.45
28		13.59	154.07
32		7.96	149.68
36		6.00	128.16
24	Bookarrival	6.01	350.69
28		5.89	291.48
32		5.52	327.70
36		4.33	281.75
24	Kendo	50.33	250.93
28		47.92	241.55
32		46.58	238.49
36		40.17	236.70
24	Newspaper	2.90	84.10
28		2.73	99.76
32		2.37	92.73
36		1.92	97.61
Average		17.31	197.47

Table 5 Simulation results for various motion activity thresholds

Sequence	MA threshold	PSNR	Bit rate	Total time	DE time
Ballroom	10	36.73	739.37	825.9	175.1
	12	36.74	739.86	828.7	171.7
	14	36.73	740.45	820.2	170.6
	16	36.73	740.42	818.6	168.9
	18	36.73	740.55	816.9	166.8
	20	36.73	741.06	815.7	165.3
	22	36.74	741.48	815.6	164.8
	24	36.73	741.42	814.3	163.1
	26	36.73	741.55	828.5	166.1
	28	36.73	742.09	815.9	161.1
	30	36.73	742.54	813.6	162.5
Breakdancers	10	38.44	427.88	3182.0	1148.6
	12	38.44	428.43	3161.2	1115.3
	14	38.44	428.53	3156.3	1113.1
	16	38.44	428.27	3146.4	1094.8
	18	38.44	428.36	3137.6	1092.9
	20	38.44	428.45	3120.8	1073.9
	22	38.44	428.60	3119.3	1075.0
	24	38.44	429.08	3100.2	1055.9
	26	38.44	429.04	3112.5	1055.1
	28	38.44	429.22	3079.3	1027.7
	30	38.44	429.57	3080.4	1027.0

estimation process is skipped. Otherwise, disparity estimation is processed for inter-view prediction. The result of the disparity estimation is then compared to the result of the motion estimation in order to obtain the best macroblock mode and vectors.

4. Experimental results

Implementation of the proposed algorithm was based on JMVM 8.0 to verify its performance [15]. All of the experiments have been performed on a desktop computer with a 2.4-GHz CPU and 2 GB of memory. The test conditions for the simulation are shown in Table 7 and information on the four test sequences are presented in Table 8[16]. Each view image of the multi-view test sequences are shown in Figure 6.

Three metrics were defined for evaluating the performance of the proposed algorithm: ΔBDPSNR, ΔBDBR, and ΔT. The metric ΔBDPSNR is defined as the change between the average PSNR of the conventional MVC and the proposed method by the Bjonteggard delta measurement [17]. As the performance improves, this criterion approaches zero. The metric ΔBDBR is the bit rate difference, as a percentage, between the compared methods by a Bjonteggard delta measurement. When the proposed algorithm exhibits the same performance as the conventional method, ΔBDBR approaches zero. If ΔBDBR has a negative value, the proposed algorithm is
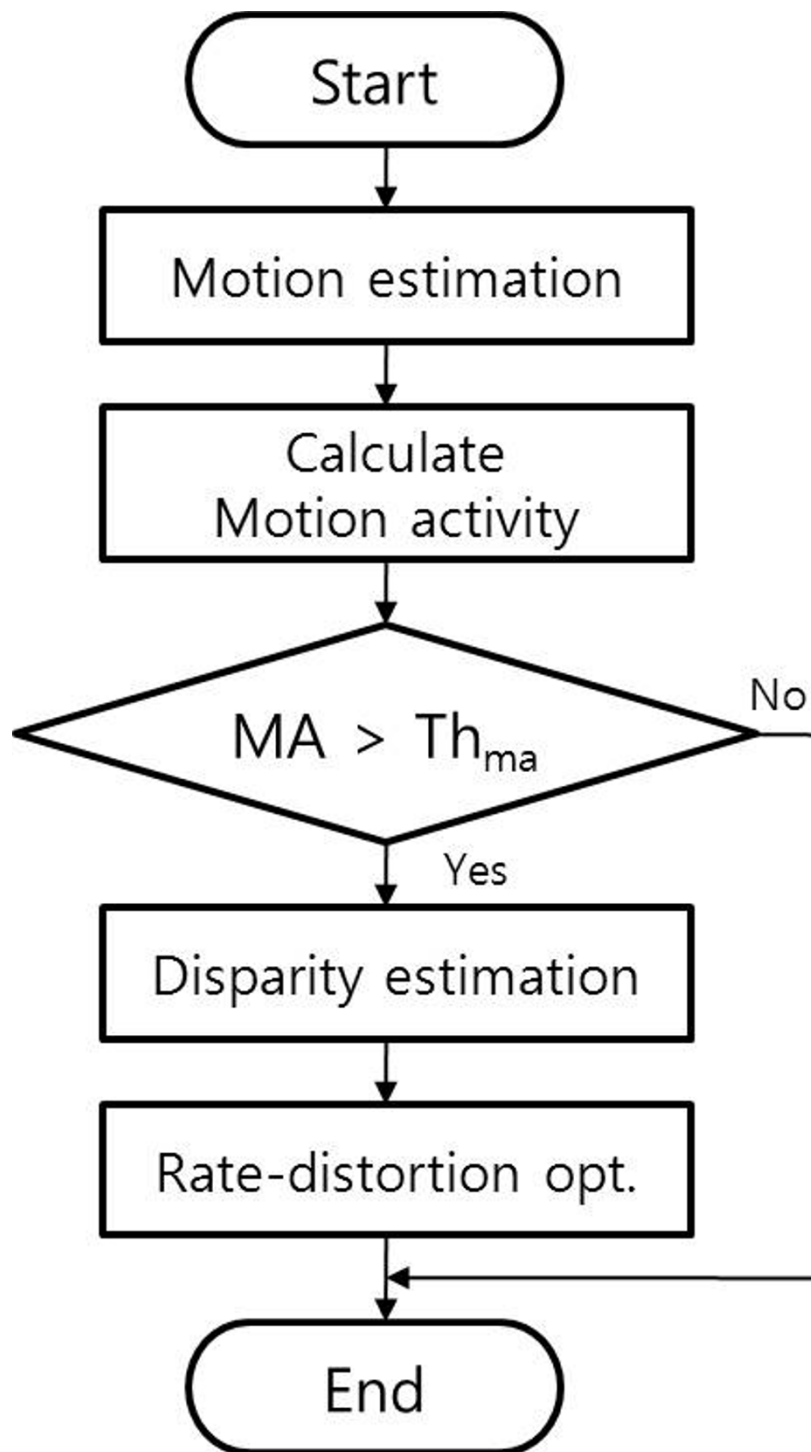
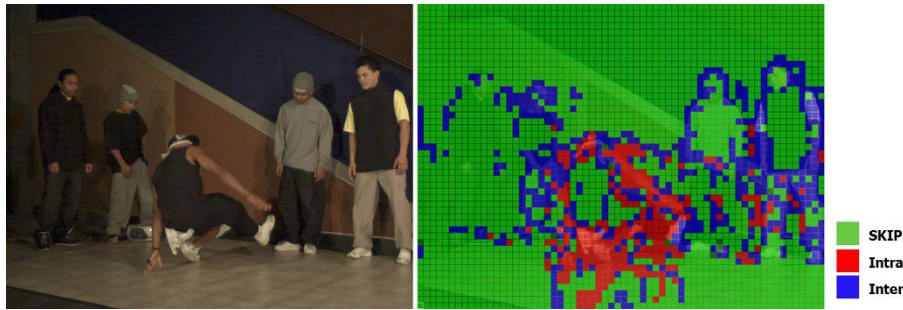**Figure 3 The overall process of the proposed algorithm based on the motion activity**.

Figure 4 Macroblock type status in the B-picture of the Breakdancers sequence.

better than the conventional method. The parameter ΔT is the processing time saving factor and is defined as:

$$\Delta T = \frac{T_{\text{conventional}} - T_{\text{proposed}}}{T_{\text{conventional}}} \times 100(\%) \qquad (2)$$

As ΔT increases in a negative direction, the performance speed is increased. With the predefined threshold, the proposed algorithm was performed for B-views of the test sequences. The experimental results of the proposed algorithm are shown in Table 9. For the B-views, the encoding speed was improved by about 73% in the disparity estimation for inter-view prediction and about 38% in the total encoding process. When compared with VADE [10], the proposed algorithm shows better performance by 14% in disparity estimation and 8% in total encoding time. As shown in Figure 7, no noticeable degradation in quality was observed. The ΔBDPSNR decrease was less than 0.05 dB and the

Table 6 Average value of the motion activity in SKIP mode and non-SKIP mode

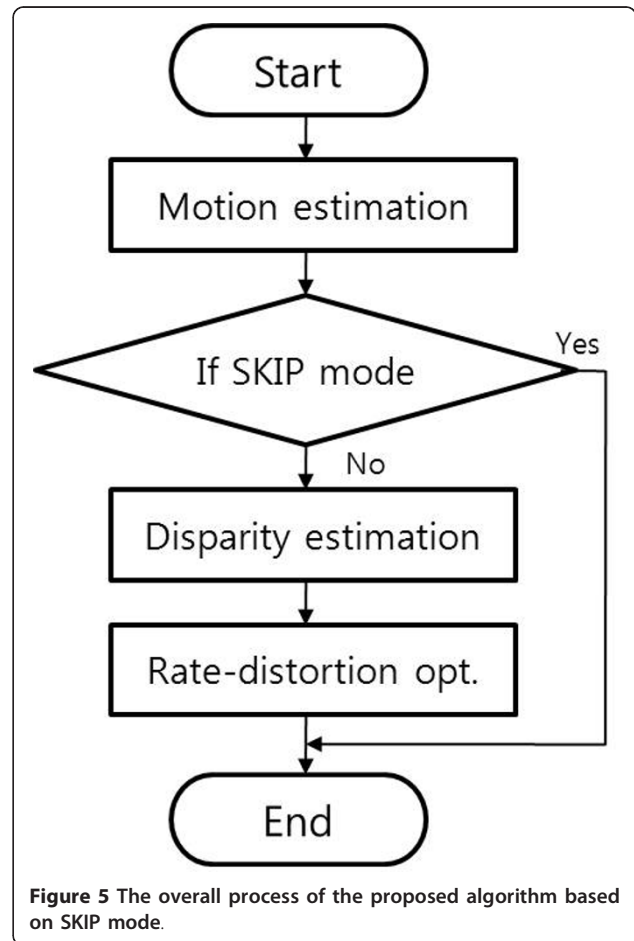| QP	Sequence	MA of SKIP mode	MA of non-SKIP mode
24	Ballroom	11.83	54.47
28		11.72	69.56
32		11.89	74.66
36		10.93	84.81
24	Breakdancers	47.02	109.91
28		40.77	109.10
32		31.72	100.14
36		20.69	97.19
24	Exit	6.79	50.34
28		6.02	66.29
32		6.06	74.53
36		4.23	77.51
24	Uli	6.09	31.82
28		6.05	34.62
32		6.49	38.84
36		4.90	42.98
Average		14.48	69.47

Figure 5 The overall process of the proposed algorithm based on SKIP mode.

Table 7 Test conditions for the experiments

Encoder	JMVM8
QP	24, 28, 32, 36
Loop filter	Enable
Encoding frames	For 1 s
Prediction structure	Hierarchical B-picture

Table 8 Parameters of the test sequences

Sequences	Ballroom	Breakdancers	Exit	Uli
Resolution	640 × 480	1024 × 768	640 × 480	1024 × 768
Camera space (cm)	19.5	20	19.5	20
Frame rate (f/s)	24	15	24	30
GOP length	12	15	12	15
Number of views	8	8	8	8
Camera arrangement	1D/parallel	1D/arc	1D/parallel	1D/parallel

ΔBDBR increase was about 1% when compared to the original results. For the Breakdancers and Uli sequences, the time saving performance was a little worse than in the other sequences because the Breakdancers sequence includes a dancer with fast movements and the Uli sequence includes many moving objects in the scene. The motion activity method presents a smaller bit rate increment and lower PSNR degradation than those of the method used by SKIP mode. The average encoding time of the motion activity method was slightly faster. This is due to the characteristics of the method used by SKIP mode. The method indirectly measures the amount of motion. SKIP mode roughly reflects the tendency of motion in a scene, but it can be selected in a fast motion area and not selected in a slow motion area. Table 10 shows the ratio of false alarms for two algorithms. The false positive affects the coding efficiency of the fast algorithm, and the false negative affects the time saving performance. The motion activity method showed more skipping of the disparity estimation process and fewer false alarms than method used by SKIP mode. However, the method used by SKIP mode is

Table 9 Performance of the proposed algorithm in B-views

Sequence	Method	ΔBDPSNR (dB)	ΔBDBR (%)	ΔT_{Total} (%)	ΔT_{DE} (%)
Ballroom	MA	-0.032	0.794	40.46	**77.34**
	SKIP	-0.074	1.845	38.04	72.66
	VADE [10]	-0.080	2.000	31.54	61.03
Break-dancers	MA	-0.038	1.625	34.57	66.31
	SKIP	-0.107	4.686	37.27	**71.60**
	VADE	-0.147	6.637	29.36	56.95
Exit	MA	-0.006	0.219	40.43	77.31
	SKIP	-0.018	0.616	41.50	**79.41**
	VADE	-0.018	0.633	31.12	60.35
Uli	MA	0.000	-0.002	38.21	**72.96**
	SKIP	0.000	0.003	34.34	65.60
	VADE	0.000	-0.012	31.06	60.10
Average	MA	-0.019	0.659	38.42	**73.48**
	SKIP	-0.050	1.788	37.79	72.32
	VADE	-0.061	2.315	30.77	59.61

Entries in bold present the largest time saving performance among the three methods for each test sequence.

simpler than the motion activity method in terms of implementation. It does not need threshold values and a calculation process to determine disparity estimation skipping. The motion activity method is suitable for software implementation because it can provide better performance through the use of an additional process for selection of the threshold values. On the other hand, the method used by SKIP mode is appropriate for a limited hardware development environment, such as mobile
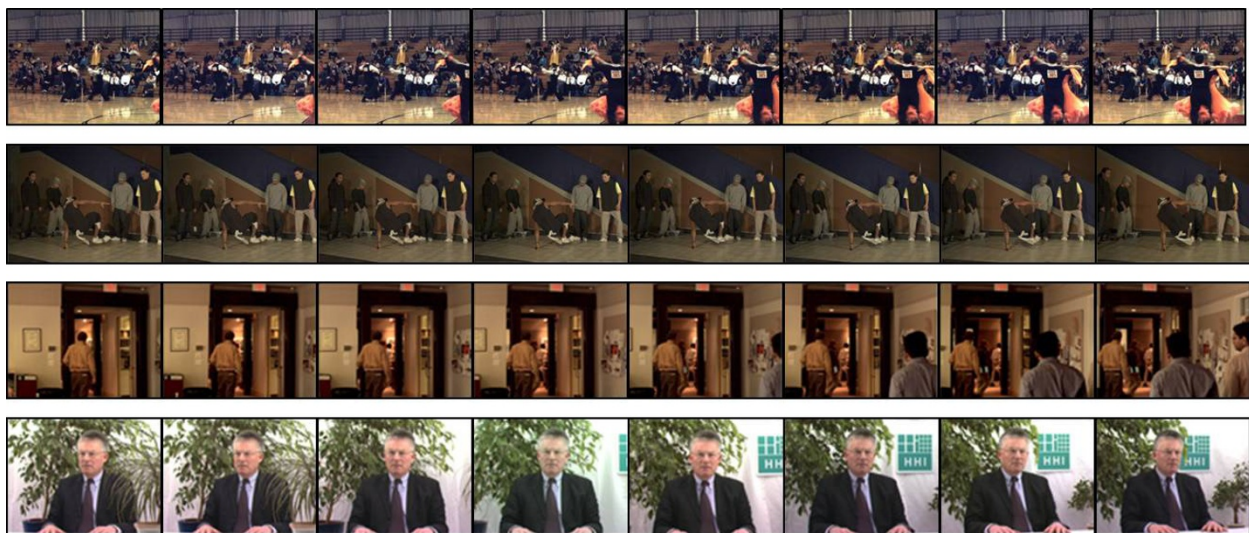

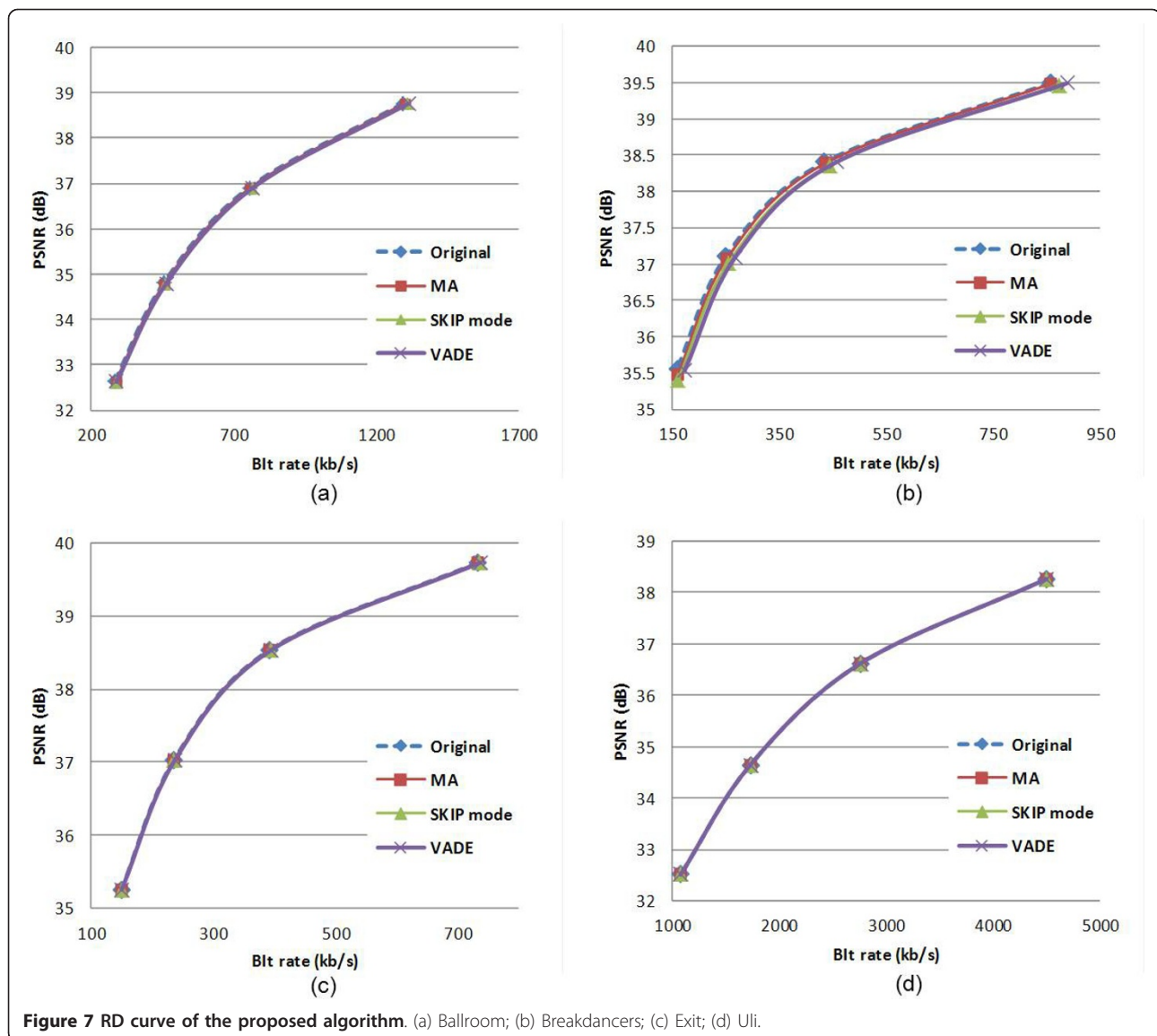**Figure 6 Test sequences (top to bottom: Ballroom, Breakdancers, Exit, and Uli)**.

Figure 7 RD curve of the proposed algorithm. (a) Ballroom; (b) Breakdancers; (c) Exit; (d) Uli.

and embedded hardware, due to its hardware-friendly characteristics.

The proposed algorithm and other conventional fast algorithms can simultaneously be applied without a loss in time saving performance, because the proposed algorithm does not theoretically overlap with the conventional fast algorithms such as search range reduction and early mode determination. Table 11 shows the performance of the proposed algorithm and the TZ-search algorithm [18], which is used in JMVM software as a fast search method. Results of the proposed algorithm with the TZ-search algorithm showed improvement of about 45% in encoding time of the B-views when compared to the results of the TZ-search algorithm alone. About 70% of the disparity estimation process was skipped by the proposed algorithm in B-view, although

the conventional fast algorithm was simultaneously applied with the proposed algorithm. This amount of time saving is similar to the results obtained by the proposed algorithm shown in Table 9.

5. Conclusion

New fast algorithms for MVC using disparity estimation skipping for inter-view prediction were proposed. The proposed algorithms determine whether disparity estimation is performed based on the amount of motion or not. To measure the amount of motion in a macroblock level, two methods were employed. The motion activity method uses motion activity as a variable for thresholding. The method used by SKIP mode omits the disparity estimation process when SKIP mode is selected in temporal prediction. The proposed methods are simple and

Table 10 The comparison of two proposed algorithms

QP	Sequences	Skip ratio of MA (%)	False positive of MA (%)	False negative of MA (%)	Skip ratio of SKIP (%)	False positive of SKIP (%)	False negative of SKIP (%)
24	Ballroom	86.81	0.60	6.86	74.18	1.33	20.22
28		87.10	0.73	7.23	80.16	1.36	14.80
32		87.68	0.87	7.23	83.90	1.20	11.35
36		88.90	0.75	6.70	86.68	0.96	9.14
24	Breakdancers	58.76	1.37	23.37	65.52	4.23	19.46
28		65.23	1.48	21.36	74.21	3.50	14.40
32		72.93	1.75	16.79	80.85	3.06	10.18
36		79.29	2.08	13.15	85.76	2.44	7.03
24	Exit	87.57	0.36	10.20	83.85	0.62	14.18
28		88.83	0.17	9.52	89.75	0.32	8.76
32		89.80	0.10	8.93	91.67	0.19	7.15
36		91.46	0.17	7.62	93.00	0.16	6.07
24	Uli	75.75	0.02	21.70	53.90	0.00	43.53
28		77.34	0.02	20.41	61.55	0.00	36.19
32		79.13	0.01	18.98	69.10	0.01	29.01
36		82.12	0.04	16.38	76.71	0.01	21.76
Average		81.17	0.66	13.53	78.17	1.21	17.08

Table 11 Performance of the proposed algorithm simultaneously performed with TZ search algorithm

Sequence	Method	ΔBDPSNR (dB)	ΔBDBR (%)	ΔT_{Total} (%)	ΔT_{DE} (%)
Ballroom	MA	-0.034	0.843	50.88	79.16
	SKIP	-0.078	1.929	46.68	72.84
Break-dancers	MA	-0.039	1.634	34.48	61.79
	SKIP	-0.112	4.906	36.66	65.86
Exit	MA	-0.004	0.161	50.55	75.39
	SKIP	-0.014	0.541	52.13	77.38
Uli	MA	0.000	-0.002	49.94	71.23
	SKIP	0.000	0.007	42.85	60.81
Average	MA	-0.019	0.659	46.46	71.89
	SKIP	-0.051	1.846	44.58	69.22

have hardware-friendly characteristics. The encoding complexity was decreased by about 73% in disparity estimation for inter-view prediction and about 38% in the total encoding process of B-views without noticeable degradation in quality. Because the proposed algorithm uses an approach that is different from the conventional fast algorithms, it exhibits consistent time saving performance when it is simultaneously applied with the conventional fast algorithms.

Competing interests
The authors declare that they have no competing interests.

References
1. A Kubota, A Smolic, M Magnor, M Tanimoto, T Chen, Multiview imaging and 3DTV. IEEE Signal Process Mag. **24**(6), 10–21 (2007)
2. S Pastoor, M Wopking, 3-D displays: a review of current technologies. Displays. **17**(2), 100–110 (1997). doi:10.1016/S0141-9382(96)01040-2
3. P Benzie, J Watson, P Surman, I Rakkolainen, K Hopf, H Urey, V Sainov, C Kopylow, A survey of 3DTV displays: techniques and technologies. IEEE Trans Circ Syst Video Technol. **17**(11), 1647–1658 (2007)
4. S Cho, N Hur, J Kim, K Yun, S Lee, Carriage of 3D audio-visual services by T-DMB, in *Proceedings of IEEE ICME 2006*, Toronto, Ontario, Canada, pp. 2165–2168 (9–12 July 2006)
5. A Vetro, P Pandit, H Kimata, A Smolic, Y Wang, 14496-10:200X/FDAM 1 Multiview Video Coding. ISO/IEC JTC1/SC29/WG11, Doc. N9978 (July 2008)
6. Y Kim, J Kim, K Sohn, Fast disparity and motion estimation for multi-view video coding. IEEE Trans Consum Electron. **53**(2), 712–719 (2007)
7. L Ding, P Tsung, W Chen, S Chien, L Chen, Fast motion estimation with inter-view motion vector prediction for stereo and multiview video coding, in *Proceeding of IEEE ICASSP 2008*, Las Vegas, Nevada, pp. 1373–1376 (31 March 2008)

8.   Z Peng, G Jiang, M Yu, Q Dai, Fast macroblock mode selection algorithm for multiview video coding. EURASIP J Image Video Process (2008). doi:10.1155/2008/393727

9.   X Li, D Zhao, X Ji, Q Wang, W Gao, A fast inter frame prediction algorithm for multi-view video coding, in *Proceeding of IEEE ICIP 2007*, San Antonio, Texas, pp. III-417–III-420 (16–19 September 2007)

10. L Shen, Z Liu, T Yan, Z Zhang, P An, View-adaptive motion estimation and disparity estimation for low complexity multiview video coding. IEEE Trans Circ Syst Video Technol. 20(6), 925–930 (2010)

11. K Mueller, P Merkle, A Smolic, T Wiegand, Multiview coding using AVC. ISO/IEC JTC1/SC29/WG11, Doc. M12945 (January 2006)

12. H Schwarz, D Marpe, T Wiegand, Analysis of hierarchical B-pictures and MCTF, in *Proceeding of IEEE ICME 2006*, Toronto, Ontario, Canada, pp. 1929–1932 (9–12 July 2006)

13. T Wiegand, H Schwarz, A Joch, F Kossentini, GJ Sullivan, Rate-constrained coder control and comparison of video coding standards. IEEE Trans. Circ. Syst. Video Technol. 13(7), 688–703 (2003)

14. P Merkle, A Smolic, K Muller, T Wiegand, Efficient prediction structures for multiview video coding. IEEE Circ Syst Video Technol. 17(11), 1461–1473 (2007)

15. A Vetro, P Pandit, H Kimata, A Smolic, Y Wang, Working draft 1 of multiview video coding reference software. ISO/IEC JTC1/SC29/WG11, Doc. N9761 (April 2008)

16. Y Su, A Vetro, A Smolic, Common test conditions for multiview video coding. ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, Doc. JVT-T207 (July 2006)

17. G Bjonteggard, Calculation of average PSNR differences between RD-curves. ITU-T Q.6, Doc. VCEG-M33 (March 2001)

18. J Reichel, H Schwarz, M Wien, Joint Scalable Video Model 8. ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, Doc. JVT-U202 (July 2006)