A FAST MULTIVIEW VIDEO TRANSCODER FOR BITRATE REDUCTION

Bing Wang, Xiaopeng Fan, Shaohui Liu, Yan Liu, Debin Zhao, Wen Gao

Dept. of Computer Science and technology, Harbin Institute of Technology, Harbin, China
{bingwang, fxp, shliu, yanliu, dbzhao}@hit.edu.cn, wgao@jdl.ac.cn

ABSTRACT

Video transcoding is an efficient way to reduce the bitrate or convert the format of the original video stream to meet the requirements of different applications and various channel capacity. In this paper, we propose a fast multiview video transcoder (MVT) for bitrate reduction. Different from the H264 transcoder, the inter-view prediction information in the input video stream is utilized to reduce the complexity of transcoding. Besides, we also utilize the mode and selected reference frame information in original stream to accelerate RD optimization calculations. Experimental results show that the proposed transcoder can achieve significant computation reduction while maintaining close RD performance compared to the fully decode and re-encode transcoder (FDET).

Index Terms—multiview video transcoder, bitrate reduction, inter-view reference.

1. INTRODUCTION

Multiview video is a group of video sequences captured by a set of same cameras on different positions at the same time instance and from the same scene. To improve the coding efficiency of multiview video coding (MVC), a prediction structure is proposed by HHI [1] as shown in Fig.1, where S is the view order and T is the time order. The hierarchical B prediction structure is used for each view and other views are taken as reference if possible. View 0, called base view, is coded without taking other views as reference. The other views which only have one direction inter-view reference are called P-views (such as S2, S3), while views with bi-direction inter-view reference are called B-views (i.e. S1).

Often, the compressed video needs to be transcoded to meet the requirements of different applications and/or fits for various storage or channel conditions. Generally, a video transcoder converts a video from one format into another format including bitrate, frame rate, spatial resolution and coding syntax [2-3]. In previous works, many transcoding techniques have been proposed for H.264. The FDET is the most simple and time consuming transcoder by fully decoding and re-encoding the input stream. Besides, there exist other three transcoding architectures for the bitrate transcoding: open-loop transcoder [4-5], cascaded pixel domain transcoder (CPDT) [5] and DCT-domain transcoder (DDT) [6]. The main disadvantage of open-loop algorithms is that they introduce error drift in video sequence. The CPDT is similar to FDET except that CPDT reuses the motion vectors along with other information extracted from the input stream directly and the computation for motion estimation (ME) is omitted. However, simply reusing the information may lead to a large bitrate increment and PSNR loss in output stream. The DDT is derived based on the assumption that DCT, IDCT, and motion compensation (MC) are linear operations, which are not strictly true [7] so that some drift may be caused in output video.

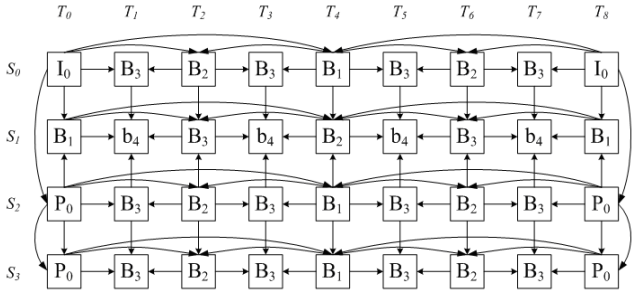

Fig. 1 Basic coding scheme of MVC

Since the MVC scheme of HHI is based on H.264, most of approaches in H.264 transcoding can be used in MVT. The most important issue in MVT is utilizing the inter-view prediction information from the input stream. Bai [8] proposes a MVT which converts multiple compressed synchronized video streams into an encoded stream. Liu [9] proposes a MVT for transcoding any viewpoint in multiview video streams encoded by MVC to bit streams that can be decoded by H.264 decoders. But to our best knowledge, MVT for bitrate reduction has not been studied before and the transcoder for H.264/AVC bitrate reduction[10-11] don't apply to MVC. In this paper, we propose a MVT by refining and utilizing the coding information in original stream instead of reusing directly. Experiment results show

This work was supported in part by the Major State Basic Research Development Program of China (973 Program 2009CB320905), the Program for New Century Excellent Talents in University (NCET) of China (NCET-11-0797), and the National Science Foundation of China (NSFC) under grants 60803147 and the Fundamental Research Funds for the Central Universities (Grant No. HIT.BRETIII.201221)

that the proposed transcoder can reduce the complexity greatly while maintaining a high RD performance.

The rest of this paper is organized as follows. In section 2, firstly we propose our transcoding scheme, and then we analyse the refinement part of proposed MVT in detail. Experimental results of our transcoder are shown in section 3. Finally we conclude this paper in section 4.

2. MULTIVIEW VIDEO TRANSCODER

An important issue in video transcoding is to utilize the information in the input video including motion vector (MV), macroblock (MB) mode and selected reference pictures. For MVT, more information can be reused, including disparity vector (DV) and other inter-view prediction information. By utilizing the original information, a video transcoder is often more efficient than the traditional video encoder.

2.1 Multiview Video Transcoder Structure

The proposed MVT structure is shown in Fig. 2, where (I)DCT means (inverse) discrete cosine transform, F means reference picture buffer, MC/DC means motion/disparity compensation. The most important part of our MVT is the refinement of coding information in original stream and will be stated in detail in the following section.

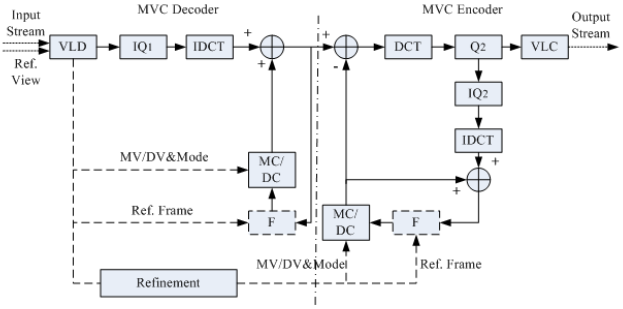

Fig. 2 Proposed MVT structure

In MVC encoder, the raw video data is encoded view by view in reference order. Corresponding inter-view reference pictures are added into reference picture lists to implement inter-view prediction. Our MVT is designed based on this scheme. For example, an 8-views video is transcoded in the following order: view0, view2, view1, view4, view3, view6, view5, view7. For base view and P-view, the original bit stream is decoded and reconstructed in order to perform inter-view prediction for B-view and P-view.

2.2 MB Mode Refinement

In MVC, the MB modes are as same as H.264/AVC. For intra mode, Intra_4x4 and 16x16 modes can be used. For inter mode, MBs can be coded in SKIP, 16x16, 16x8, 8x16 and 8x8 mode. In the Inter_8x8 prediction mode, each MB can be further partitioned into sub-MBs: 8x8, 4x8, 8x4 and 4x4 sub-MB. For each MB, a RD Optimization (RDO) contains ME, disparity estimation (DE) and transform is performed among all available modes to select the best mode, which is the most time consuming part in MVC.

In requantization transcoder, since a larger quantization parameter (QP) is used to transcode the input video stream, some regions in original stream may be more homogeneous in transcoded video. Therefore, a lager MB partition may be selected as the best mode compared to the original video [12-13]. We encode different views from several video sequences (*ballroom*, *ballet* and *breakdancers*) with QP = 27, and then transcode with QP = 32, the statistical results of input and output modes can be found in Table I.

Table I. MB modes variation after requantization

QP27 \ QP32	SKIP	Inter 16x16	Inter 16x8	Inter 8x16	Inter 8x8	Intra 16x16	Intra 8x8
SKIP	3244	434	70	48	2	16	0
16x16	1531	726	110	89	22	62	4
16x8	840	352	161	63	24	24	4
8x16	717	322	53	170	31	24	2
8x8	426	270	120	118	192	11	6
16x16I	686	148	12	20	4	235	5
8x8I	897	639	220	185	55	549	809

In Table I, the first column means MB modes in original stream and the following columns means modes selected when transcoding. From Table I we also find that over 50% MBs are transcoded with SKIP or Inter_16x16 regardless of the original modes.

Furthermore, since the video contents are similar between adjacent views, the prediction mode of a MB in current view is most similar to the mode of the corresponding MB in neighbor view. Therefore, the mode of the corresponding MB in neighbor view shall be utilized to accelerate the mode decision of current MB when transcoding [14-15].

Based on the above analysis, we propose our MB mode refinement scheme which utilizes the mode information in original stream and neighbor view as follows.

First, it has been well-recognized that for MBs in regions with complex motion or rich texture, their prediction mode sizes are diverse, usually Inter_8x8 or Intra mode. On the other side, for MBs in region with homogeneous motion and texture, their prediction mode sizes are usually larger, such as SKIP or Inter_16x16 mode. That is, there exists a strong correlation between prediction mode and MB complexity. In this paper, we define the MB complexity as MBC, which is calculated using formula (1).

$$MBC = \lambda \cdot W_{input} + (1-\lambda) \cdot \sum_{i=0}^{N-1} \alpha_i \cdot W_i \qquad (1)$$

In (1), W means MB mode factor, which is defined in Table II. The MBC is calculated by the linear combination of the input mode factor and the mode factors of corresponding MBs in neighbor view, which is located using global disparity vector (GDV) [16] as shown in Fig. 3. Since GDV is measured by MB-size of unit and is not exactly the disparity between current MB and the corresponding one in neighbor view, the modes of the corresponding MB (0) and N-1 (N is set to 9 in this paper)

of its neighbor MBs (1~8) as shown in Fig.3 are used to estimate mode characteristic of MB A'. The parameter α is the weight factor of the neighbor MBs. The closer the neighbor MB to A', the larger weight factor should be assigned. The weights are determined through our extensive experiments and documented in Table III. The model parameter λ is set to 0.6 through our experiments.

Table II. MB mode factors

MB mode	SKIP Inter16x16	Inter16x8 Inter8x16	Inter8x8	Intra16x16 Intra8x8
W	0	1	2	3

Table III. Weight factors of neighbor MBs

MB Index	0	2,4,5,7	1,3,6,8
Weight factor	0.25	0.125	0.0625

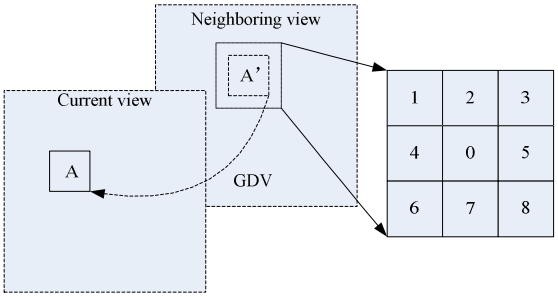

Fig. 3 Corresponding MBs in neighboring view

Second, all MBs are classified into four groups in terms of MBC. For each group, only part of modes are selected as candidate modes and considered during RDO loop when transcoding. The four groups and corresponding candidate modes are listed in Table IV.

Table IV. All groups and candidate modes

Group	Candidate modes
Simple	SKIP,Inter16x16
Medium simple	SKIP,Inter16x16,Inter16x8,Inter8x16
Medium complex	All Inter modes
Complex	All modes

The criterion for MB classification is as follow

$$\begin{cases} MBC \le T_1 & \text{MB in simple group} \\ T_1 < MBC \le T_2 & \text{MB in medium simple group} \\ T_2 < MBC \le T_3 & \text{MB in medium complex group} \\ MBC > T_3 & \text{MB in complex group} \end{cases} \quad (2)$$

In (2), T_i (i = 1, 2, 3) is threshold for MBC. Through our extensive experiments, T_1 T_2 and T_3 are set to 0.60, 1.20 and 1.80 respectively.

Experiments show that the accuracy of proposed mode refinement scheme is over 90%, and for sequences with low motion, most MBs are classified into simple or medium simple group so that fewer modes are considered and the complexity of RDO loop for transcoding is reduced greatly.

2.3 MV and DV Refinement

For transcoding, MV/DV extracted from original stream is usually more accurate than the one predicted by neighboring MBs. Therefore, MV/DV in original stream is used as the motion or disparity search center for corresponding MB in transcoding and the optimal MV/DV usually lies in a relatively small range around the search center.

However, due to requantization in transcoding, the best mode used to encode a MB in transcoding may be different from that in original stream as stated in section 2.2 and MVs/DVs cannot be reused directly. In this case, we propose an algorithm to select the best MV/DV from original stream as the start MV/DV according to the predicted MV/DV by neighboring MBs.

For simplicity, we suppose that a MB coded with Inter_8x8 mode in original stream is transcoded with Inter_16x16 mode as shown in Fig. 4. In this case, the MV/DV for each 8x8 sub-block may be different and taking average of four vectors as start vector of block B may introduce a mistake when a large discrepancy exists among them. Here, we calculate the start vector for B (denoted as V_B) using formula (3).

$$V_B = \arg \min_{i=\{00,01,10,11\}} Dis(V_i, Pv). \quad (3)$$

In (3), Pv means the MV/DV predicted by neighboring MBs and $Dis(V_i, Pv)$ means the distance between V_i and Pv which is calculated using formula (4).

$$Dis(V_i, Pv) = \left|Hor(V_i) - Hor(Pv)\right| + \left|Ver(V_i) - Ver(Pv)\right|. \quad (4)$$

In (4) Hor(V) and Ver(V) denote the horizontal and vertical component of vector V respectively.

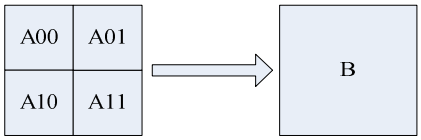

Fig. 4 MV/DV refinement between different modes

In H.264/AVC transcoding, motion vectors extracted from the incoming bit stream are usually reused. However, due to the requantization in transcoder, MVs extracted from input stream are usually not optimized. To overcome the loss of quality without performing a full motion re-estimation, MV refinement schemes are proposed in [17].

A DV refinement scheme for MVT is proposed in this paper. Typically, the search window used for MV and DV refinement is relatively small compared with the original search window. To determine the search range for MV and DV refinement, we select 300 frames from three sequences: b*allroom* (640x480), *exit* (640x480) and *breakdancers* (1024x768) and transcode them from QP=20 to QP=27. Experiment results are shown in Table V.

In Table V, P(n) means the probability (%) of the distance between actual MV/DV and the predicted one is not larger than n (n = 0,1,2,…,6). We find that over 95% MVs/DVs lie within 4 pixels around the predicted MV/DV for sequences with resolution 640x480 and a larger search range is not necessary. For sequences with a higher

resolution, a relatively larger search range is needed to reach a high enough probability. Through our extensive experiments, we set search range to 4 and 6 for sequences with resolution 640x480 and larger sequences respectively.

Table V. MV and DV Search Range Statistics(%)
(1 ballroom 2 exit 3 breakdancers)

Seq.	MV/DV	P(0)	P(1)	P(2)	P(3)	P(4)	P(5)	P(6)
1	MV	79	93	96	97	98	98	98
	DV	69	93	96	97	98	98	98
2	MV	76	91	94	96	97	97	98
	DV	74	89	94	95	96	97	97
3	MV	54	80	88	90	92	94	96
	DV	65	85	91	91	93	95	96

2.4 Reference Frame Refinement

In MVC, inter-view prediction is usually used for regions with complex motion, such as deformation, rotation and zooms because these regions are usually predicted with small block size and large MV, which decrease the coding efficiency. Similarly, regions with low motion are usually encoded using intra-view prediction. The properties of video sequence mainly remain the same except some variation in texture detail after transcoding. Therefore, there is strong correlation in reference frame before and after transcoding.

We transcode three sequences in section 2.2 for test and notice that almost all MBs are transcoded using the same reference frame as they used in the original stream (over 95%). Based on this, we select reference frames as follows:

(1) If the current MB mode is same as that in original stream, then we select the same reference picture as in original stream for transcoding.

(2) If MB mode is changed, e.g. Inter_8x8 changed to Inter_16x16; we select the dominant reference frame for the original MB as reference frame for the new MB. Here the dominant reference frame means the most frequently used reference frame by sub-blocks in a MB as shown in Fig. 5. If the dominant reference frame cannot be determined, i.e. each reference frame in the reference list is used equally by sub-blocks, the residual of each sub-block is calculated and the reference picture of the sub-block with the least residual is selected as the best reference for the new MB.

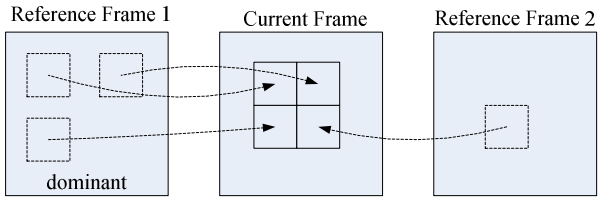

Fig. 5 Dominant reference frame selection
(Reference frame 1 is selected as the dominant one because three of four sub-blocks use it as reference)

Besides, experiments also demonstrate that if the original MB is encoded using uni-direction prediction, it possess a high probability (over 96%) to remain uni-direction prediction, otherwise, if the original MB is encoded using bi-direction prediction, both bi-direction and uni-direction prediction are possible. Therefore, for blocks coded using uni-direction prediction, the iterative search for bi-direction prediction can be omitted.

For a frame in B view, there are four reference frames by default in MVC, including two intra-view and two inter-view reference frames. By utilizing reference information in original stream, about 70% calculation for reference selection can be saved.

3. EXPERIMENTS AND RESULTS

In our experiments, seven multiview video sequences shown in Table VI are tested. Views 0 to view 2 of each sequence are transcoded, among them, view 0 is base view and considered as the neighbor view of view 1 and view 2, view 1 is B-view and view 2 is P-view. All sequences are first encoded with QP_1 and then transcoded with QP_2. We test various values for QP_1 and QP_2 in our experiments. The simulation results for $QP_1 = 20$ and $QP_2 = 25$ are listed in Table VII. Table VIII shows the BD-PSNR and BD-Bitrate compared to FDET for $QP_1 = 20$ and $QP_2 = \{22, 27, 32, 37\}$. All values listed are the average value of three views. The implementation is based on the version 8.5 of reference software JMVC with GOP size = 8. The performance of CPDT (all information reuse) is also listed for comparison.

Table VI Test Sequences

Number	Sequence	Resolution
1	Ballroom	640x480
2	Vassar	640x480
3	Exit	640x480
4	Race1	640x480
5	Flamenco2	640x480
6	Ballet	1024x768
7	Breakdancers	1024x768

The performance of proposed MVT is calculated using (5)-(7) and for CPDT the calculation is similar.

$$\Delta \mathrm{T}ime = \frac{Time_{FDET} - Time_{MVT}}{Time_{FDET}} \times 100\% \quad (5)$$

$$\Delta Bitrate = \frac{Bitrate_{MVT} - Bitrate_{FDET}}{Bitrate_{FDET}} \times 100\% \quad (6)$$

$$\Delta PSNR = PSNR_{MVT} - PSNR_{FDET} \quad (7)$$

Table VII Experiment results for $QP_1 = 20$, $QP_2 = 25$

Seq	ΔBitrate(%)		ΔPSNR(dB)		ΔTime(%)	
	CPDT	Proposed	CPDT	Proposed	CPDT	Proposed
1	+11.30	+1.01	+0.001	-0.026	94.41	90.91
2	+13.41	+0.99	-0.085	-0.007	93.50	89.01
3	+32.15	+1.59	+0.050	-0.023	94.95	90.54
4	+31.43	+0.83	-0.014	-0.013	96.40	90.97
5	+17.53	+1.55	-0.039	-0.025	95.88	90.16
6	+35.78	+1.92	+0.073	-0.029	95.37	91.84
7	+29.43	+1.23	-0.045	-0.051	96.77	90.90
Avg	+24.43	+1.30	-0.008	-0.025	95.33	90.62

Table VIII BD-Bitrate and BD-PSNR

Sequence	BD-Bitrate(%)		BD-PSNR(dB)	
	CPDT	Proposed	CPDT	Proposed
1	+22.40	+2.57	-0.925	-0.111
2	+32.16	+2.68	-0.877	-0.081
3	+50.10	+3.31	-1.475	-0.110
4	+35.21	+2.06	-1.620	-0.100
5	+15.77	+3.03	-0.832	-0.166
6	+51.67	+5.39	-1.785	-0.218
7	+49.21	+5.66	-1.090	-0.156
Avg	+36.65	+3.53	-1.227	-0.135

As demonstrated in Table VII, the PSNR degradation of the proposed MVT is only 0.025 dB while the bitrate increment is 1.30% and the transcoding time is reduced by over 90% on average. In the CPDT, the RDO loop is omitted by reusing the original information so that much time is saved. However, the MB mode and reference frame used in CPDT is not always the best one so that the PSNR of some sequence may improve but with a severe bitrate increment.

From Table VIII, we conclude that the proposed MVT significantly outperforms the CPDT in coding efficiency with slightly higher computation cost. Besides, the R-D performance of proposed MVT is most similar to FDET. The R-D curves of FDET, CPDT and proposed MVT for *ballroom* sequence are shown in Fig. 6. The R-D curves of FDET and the proposed MVT are so close that cannot be distinguished from each other in the figure.

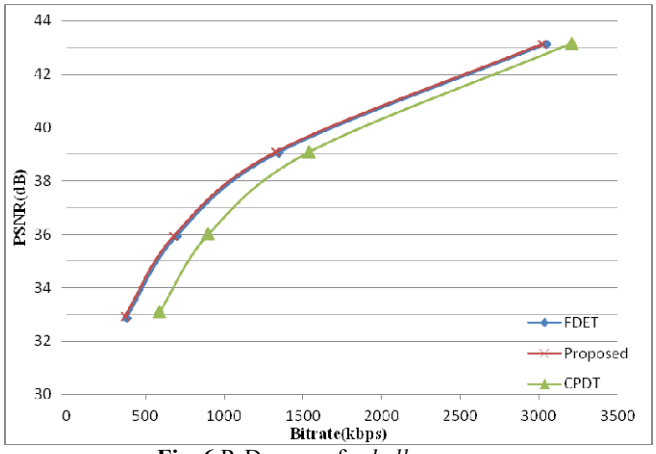

Fig. 6 R-D curves for *ballroom* sequence

4. CONCLUSION

In this paper, we propose a multiview video transcoder for bitrate reduction. By utilizing the MV/DV, MB mode and reference frame information in original multiview video stream, the proposed transcoder can transcode the input stream to lower bitrate efficiently. Experiment results show that the complexity of transcoding is reduced by over 90% compared with FDET, while the PSNR degradation and the bitrate increment are negligible. Compared with CPDT which reuse all coding information in original stream, the improvement in coding performance is quite remarkable.

5. REFERENCES

[1] "Description of core experiments in MVC," ISO/IEC JTC1/SC29/WG11, MPEG2006/W7798, January 2006.

[2] A. Vetro, C. Christopulos, and H. Sun, "Video transcoding architectures and techniques: An overview," *IEEE Signal Process. Mag.*, vol. 20, no. 2, pp. 18–29, Mar. 2003.

[3] I. Ahmad, X. Wei, Y. Sun, and Y.-Q. Zhang, "Video transcoding: an overview of various techniques and research issues," *IEEE Trans. Multimedia*, vol. 7, no. 5, pp. 793–804, Oct. 2005.

[4] A. Eleftheriadis and D. Anastassiou, "Constrained and general dynamic rate shaping of compressed digital video," in Proc. *IEEE Int. Conf. Image Processing*, vol. 3, 1995, pp. 396–399.

[5] H. Sun, W. Kwok, and J. W. Zdepski, "Architectures for MPEG compressed bitstream scaling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 2, pp. 191–199, Apr. 1996.

[6] D. G. Morrison, M. E. Nilson, "Reduction of the bit-rate of compressed video while in its coded form," in *Proc. 6th Int. Workshop Packet Video*, 1994, pp. D17.1–D17.4.

[7] J. Youn and M.-T. Sun, "Video transcoding with H.263 bit streams," *J. Visual Commun. Image Represent.*, vol. 11, pp. 385–404, Dec. 2000.

[8] B. Bai, P. Boulanger and J. Harms, "A multiview video transcoder," in Proc. 13th annual *ACM Int. Conf. on Multimedia*, 2005.

[9] S. Liu, "Multiview Video transcoding: From multiple views to single view," *Picture Coding Symposium*, 2009.

[10] Kwee-Li Cheng, Naofumi Uchihara, "Analysis of Drift-Error Propagation in H.264/AVC Transcoder for Variable Bitrate Streaming System", *IEEE Transactions on Consumer Electronics*, vol. 57, pp. 888-896, May, 2011.

[11] M. Li, B. Wang, "Hybrid Video Transcoder for Bitrate Reduction of H.264 Bit Streams", *International Conferences on International Assurance and Security*, vol .1, pp. 107-110, Aug. 2009.

[12] Nam, Hyeong-Min, etc; "Low Complexity H.264 Transcoder for Bitrate Reduction," *Communications and Information Technologies*,2006. ISCIT '06. International Symposium on Oct. 18 2006-Sept. 20 2006, pp.679 – 682.

[13] J. Jiang, Y. Lin, "Efficient Mode Decision for H.264/AVC Frame skipping transcoding," *TENCON 2010 - 2010 IEEE Region 10 Conference,* pp.2062-2065, Nov. 2010.

[14] H. Q. Zeng, C. H. Cai, and K.-K. Ma, "Fast Mode Decision for Multiview Video Coding Using Mode Correlation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 11, pp. 1659-1666, Nov. 2011.

[15] L. Shen, T. Yan, Z. Liu, Z. Y. Zhang, P. An, "Fast mode decision for multi-view video coding", *IEEE Int. Conf. Image Processing (ICIP)*, Nov. 2009.

[16] H. S. Koo, Y. J. Jeon, and B. M. Jeon, "MVC motion skip mode," ISO/IEC JTC1/SC29/WG11 and ITU-T Q6/SG16, Doc. JVT-W081, Apr. 2007.

[17] J. Youn, M.T. Sun, and C.W. Lin, "Motion vector refinement for high performance transcoding," *IEEE Trans. Multimedia*, vol. 1, pp. 30-40, Mar. 1999.