Distributed multiview video coding with 3D-DCT transform domain Wyner-Ziv codec

Distributed Multiview Video Coding with 3D-DCT Transform Domain Wyner-Ziv Codec

Jonathan K.K. Loo*[1], Zhuo Xue[1], Enrico Masala[1], Amanda P.Y. Yip[2], Dhananjay Singh[3]

[1]Computer Communications, School of Engineering and Information Sciences, Middlesex University, London, NW4 4BT, UK.
E-mail: {J.Loo,E.Masala}@mdx.ac.uk, zhuo.xue@hotmail.com

[2]School of Computer Science, University of Hertfordshire, Hertfordshire, AL109LW, UK
E-mail: P.Y.A.Yip@herts.ac.uk

[3]Division of Fusion and Convergence of Mathematical Sciences, National Institute for Mathematical Sciences (NIMS), KT Daeduk 2 Research Center, 463-1, Jeonmin-dong, Yuseong-gu, Daejeon, 305-390, South Korea E-mail: singh@nims.re.kr

*Corresponding author

Abstract—The need for efficient multiview video coding schemes is expected to strongly increase in the near future. The Distributed Multiview Video Coding (DMVC) approach seems very promising since it can achieve good compression efficiency while keeping the complexity low. The main contribution of this paper is to investigate how to improve the classic DMVC framework based on transform domain WZ video coding (TDWZ) by means of the introduction of the 3D-DCT. The main advantage of this combination resides in the limited computational complexity of the overall framework, which however does not penalize the compression performance since both the spatial and the temporal domain correlation can be exploited due to the use of the 3D-DCT. The framework is designed in a flexible way so that it can handle both traditional and residual-frame based WZ coding. The simulation results confirm the validity of the proposed framework in terms of video quality improvements, with gains up to 4.4 dB PSNR compared to a pixel-domain WZ technique, and up to 0.6 dB PSNR compared to a 2D-DCT based one.

Index Terms—Multiview video coding, Wyner-Ziv video coding, Distributed video coding, 3D-DCT

International Journal of Multimedia Intelligence and Security (IJMIS), vol. 2, n. 1, 2011, pp. 54-74, DOI: 10.1504/IJMIS.2011.040929 (ISSN: 2042-3462)

1. Introduction

In order to provide a more vivid video experience, multiview seems one of the solutions that will dominate in the near future. Therefore, it is desirable to extend the current monoview video coding schemes to the multiview scenario. Multiview video coding implies compressing video content from multiple cameras, which are placed at different locations and angles in the same scene. The collected video data can be further processed by multiview applications on the decoding side such as free viewpoint television. However, the amount of the video data is very large and thus efficient multiview video coding algorithms are required to achieve high compression. There have been numerous research achievements in the design of efficient multiview video coding algorithms. Most of them focused on the design of view synthesis structures such as the works introduced in [1-3]. In [4-6], authors focused on the development of various prediction strategies in multiview video coding. In [7, 8], the application of multiview video coding for 3DTV application scenarios is discussed. Basically, three cases of multiview video coding architectures can be envisaged. They are summarized in the following.

- Case 1. Each camera is encoded separately and decoded separately. The encoding process of each camera is independent and it is carried out with a conventional video coding standard. No communication is required between cameras during the encoding process. The architecture is illustrated in Fig. 1.

- Case 2. In this case, the video content from multiple cameras is jointly encoded and jointly decoded. The typical configuration is that one camera uses the conventional predictive video coding, and the other cameras perform motion estimation and motion compensation with respect to the content encoded by the first one and only the residual

and MV are encoded. This is typically called inter-view coding. Fig. 2 and 3 illustrate such an architecture. An overview of the inter-view coding structure is presented in [8].

- Case 3. This case is referred to as Distributed Multiview Video Coding (DMVC), in which each camera is separately encoded but jointly decoded. MVC was originally proposed in [2-6], it employs the WZ video coding [9, 10] to encode video content from multiple cameras. Its typical architecture is shown in Fig. 4

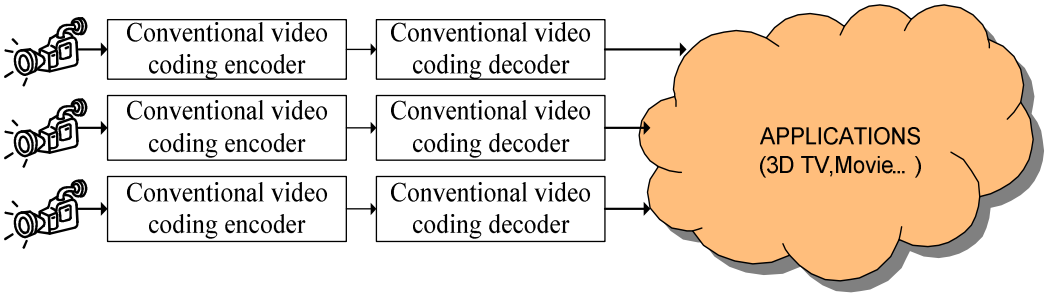

Fig. 1. Multiview video coding, Case 1 (conventional coding)

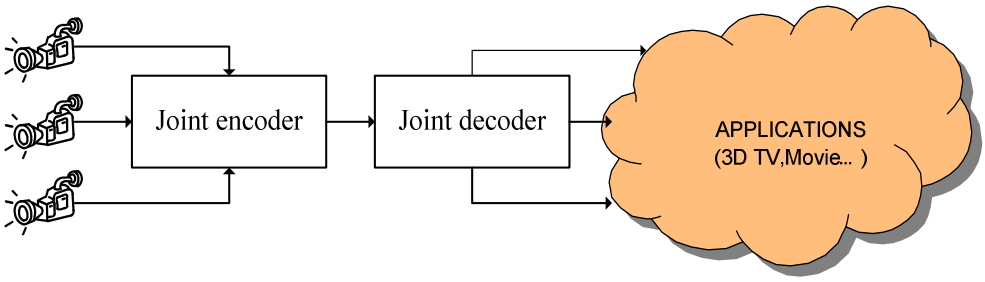

Fig. 2. Multivew video coding, Case 2 (inter-view coding)

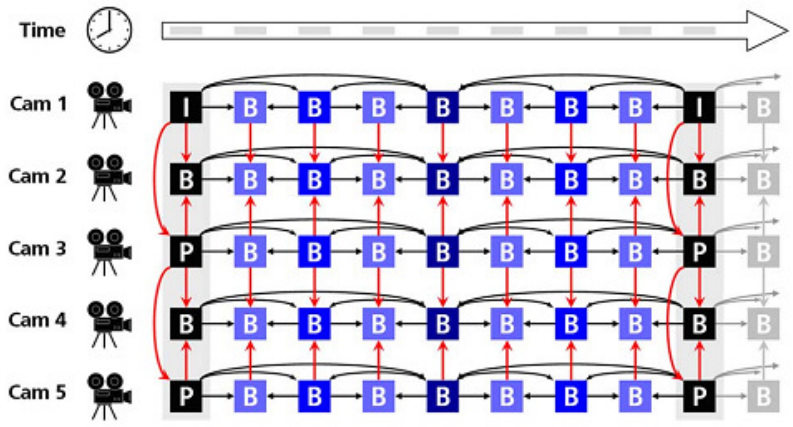

Fig. 3. A typical inter-view video coding structure [8]

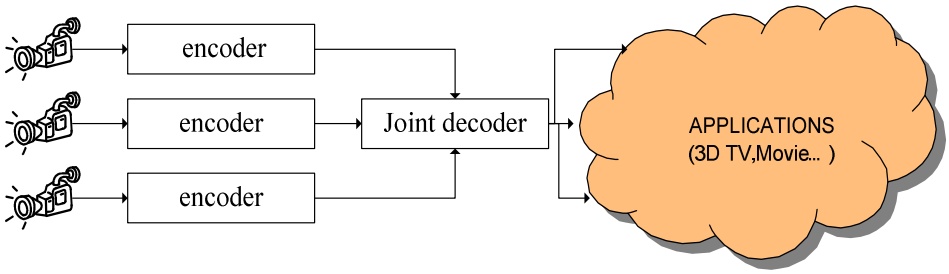

Fig. 4. Multiview video coding, Case 3 (distributed multiview video coding)

In multiview video coding, cameras are aimed at the same scene but from different angles, and therefore high correlation exists among the content captured by these cameras. For Case 1, each encoder employs the conventional video coding standard and works separately, the correlation among cameras is not exploited at all and thus the compression is not efficient. For Case 2, the correlation is exploited by the inter-view prediction and better compression can be achieved. For Case 1 and Case 2, complex encoders are required due to the predictive coding structure. The encoder of Case 2 is even more complex than that in Case 1. However, in certain multiview applications where the encoder is energy constrained and cannot afford such a heavy task, Case 1 and 2 cannot be used. Particularly, in Case 2, the prediction structure requires the cameras to be able to communicate with each other, which is difficult in real time multiview coding since communication among cameras requires exchanging a large amount of data. Case 3 provides an optimum solution for such a situation with limited computation capabilities at the encoder. First, it does not need a complex encoder due to the use of WZ video coding. Compared to traditional video coding standard, the WZ video coding reverses the asymmetry of the coding structure: the encoder is simple while the decoder is more complex. Second, WZ coding does not require cameras to communicate with each other during the encoding. Finally, it exploits the correlation among the cameras at the decoder and thus good compression efficiency can still be achieved.

The performance of DMVC highly relies on the adopted WZ video coding scheme. Although WZ video coding has evolved from the earlier architectures [10, 11] into more advanced architectures such as DISCOVER [12, 13], most of the works concerning DMVC are mainly based on the simple Pixel Domain Wyner-Ziv video coding (PDWZ) scheme introduced in [14-16]. Since the Transform Domain Wyner-Ziv video coding (TDWZ) provides better compression performance than the normal PDWZ, it is reasonable to constitute a transform domain DMVC by combining TDWZ and DMVC, as in [17, 18] which applies TDWZ to 2D-DWT or [19] which relies on 2D-DCT instead of 2D-DWT. The compression performance of DMVC strongly relies on the used WZ video coding scheme. In current WZ video coding, the residual WZ video coding shows better compression performance with a slight increase in the complexity at the encoder. In [20], Aaron explored the performance of WZ coding applied to residual frames in the pixel domain. In that work, the WZ frame is not directly encoded. Instead, the residual frame resulting from the subtraction of the WZ frame and the reference frame is encoded. The residual frame is obtained by making simple subtraction in order to keep the complexity low at the encoder. The reference frame is available at both encoder and decoder. At the decoder, the WZ frame is reconstructed by using the decoded residual frame and the reference frame. The performance of PDWZ is much increased and it can even reach levels similar to that of TDWZ with the help of a hash code generated at the encoder. In [21], the case of WZ residual coding in the transform domain is considered. Superior performance is reported with respect to the normal TDWZ. Both works assume that the reference frame used to produce the residual frame is available at both the encoder and decoder. The principle on which the residual frame technique relies is similar to the one of DPMC. Due to the high correlation in the temporal direction, the residual value tends to be zero or a very small value compared to the original pixels. When the same number of quantization levels is applied, a smaller distortion is obtained after the reconstruction.

In order to keep the encoder complexity low while taking advantage of the temporal correlation among subsequent frames, this work proposes the use of a cubewise 3D-DCT transform. 3D-DCT is known for its ability to exploit correlation in both the spatial and the temporal domain at the same time.

Motivated by the possibility to further improve the DVC performance in a multiview scenario by means of the 3D-DCT transform without significantly increasing the complexity, this paper presents a new DMVC framework which combines the advantages of both the transform domain WZ video coding (TDWZ) and the 3D-DCT, from now on called 3TD-DMVC, as well as a variant of the 3TD-DMVC which performs coding of residual frames called 3RTD-DMVC. Moreover, in order to be able to apply the 3D-DCT transform, this paper also presents a method to separate video sequences into groups at encoder and decoder suitable for the application of cubewise 3D-DCT in the WZ video coding scenario. The simulation results confirm the validity of the proposed framework in terms of video quality improvement.

The rest of paper is organized as follows. Section 2 presents the camera configuration and the joint side information generation which is one of the most important components of the proposed DMVC framework. Section 3 and 4 present the architecture of 3TD-DMVC and 3RTD-DMVC, respectively. Section 5 details the processes and operations including 3D-DCT processing and 3D quantization volume design for the proposed 3TD-DMVC and 3RTD-DMVC architectures. Section 6 discusses the simulation outcomes and the performance of the proposed DMVC architectures in comparison with other DMVC architectures. Conclusions are drawn in Section 7.

2. Configuration in DMVC

The camera configuration and joint side information generation for the proposed DMVC framework are discussed in this section. The camera configuration determines the property of the cameras involved and it also has an impact on the coding mode of the captured frames. The joint side information generation (JSG) process, explained later, is used to generate the corresponding side information for the frame coded by WZ coding.

2.1 Camera Configuration

The camera configuration is the same as the one proposed in the DMVC in [15], in which cameras are divided into two types: the key camera and the WZ camera as shown in Fig. 5. All video frames in the key camera are coded by intra-frame coding while in the WZ camera frames are organized into Group of Pictures (GOP) for coding. The first frame of each GOP is coded by intra-frame coding and the rest are WZ frames which are coded by WZ coding. The decoding is performed in the so called Joint Decoding Unit (JDU) which will be discussed in further detail in Section 2.2. In the JDU, first all intra frames (I frames) are reconstructed by the decoder followed by WZ frames (W frames). For the W frames coded using the WZ technique, the corresponding side information which is considered as the estimation of each WZ frame will be generated through the joint side information generation process, with which the WZ frames can be decoded by exploiting the correlation among cameras.

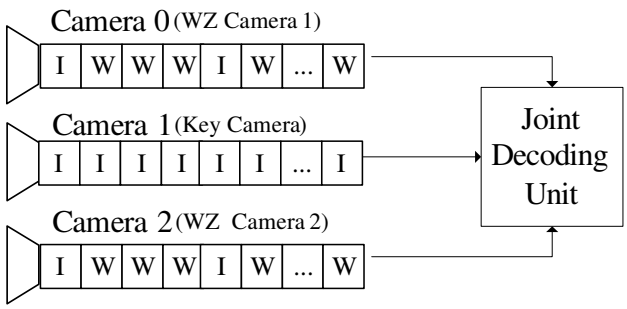

Fig. 5. Frame composition of key camera and WZ camera in [15]

2.2 Joint Side Information Generation

Based on the camera configuration type chosen in the previous section, this part describes the corresponding side information generation for the WZ frame in the WZ camera by jointly utilizing the received I frames from the same camera and neighboring cameras. The side information can be obtained by performing motion compensated interpolation (MCI) between two adjacent I frames from the same WZ camera. This is usually referred to intra-camera interpolation and widely used in WZ video coding for the monoview case. In DMVC, since high correlation exists among neighboring cameras, frames from neighboring cameras can actually be used to assist in generating more accurate side information. The side information generation using only frames from neighboring camera is called inter-camera interpolation (or inter-view interpolation). It has been shown in [14-17, 22] that the combination of intra-camera interpolation and inter-camera interpolation can provide better side information than using only a single interpolation scheme. Intra-camera interpolation is not good at estimating high motion areas while inter-camera interpolation has problems with scenes including occlusions, reflections, etc.

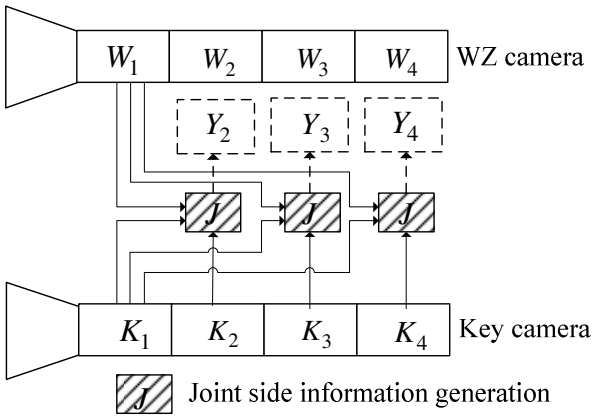

Fig. 6. Joint side information generation (JSG) structure

In this paper, we propose a joint side information generation (JSG) for the chosen camera configuration type. The JSG is the heart of joint decoding unit (JDU) used in the 3TD-DMVC

and 3RT-DMVC architectures (see Sections 3 and 4). The general structure of the joint side information is shown in Fig. 6. $W_1W_2W_3W_4$ are frames from the WZ camera, in which W_1 is intra-fame coded and the rest are WZ frames coded by means of the WZ coding. $K_1K_2K_3K_4$ are key frames from the key camera which are all coded using intra-frame coding. Y_2 is the corresponding side information of W_2 which is generated by jointly using $W_1K_1K_2$. Y_3 is generated by using $W_1K_1K_3$ and Y_4 is to be obtained by using $W_1K_1K_4$.

Fig. 7 shows the exact procedure of generating side information Y_2 by joint interpolation and the details are given as follows. Firstly, the temporal motion vectors indicated by MV_temporal together with a cost vectors indicated by CV_temporal is calculated between the frames K_1 and K_2 from the key camera based on block matching. Here the cost list contains the difference strengths of each pair blocks associated with the motion vectors and the Mean Square Difference (MSD) value is used to describe the cost of each pair of blocks. Meanwhile, another group of motion vectors indicated by MV_spatial together with a cost vector indicated by CV_spatial are subsequently calculated between the frame K_1 from the key camera and the intra frame W_1 from the WZ camera. Similarly, the MSD is used in estimating the coding cost.

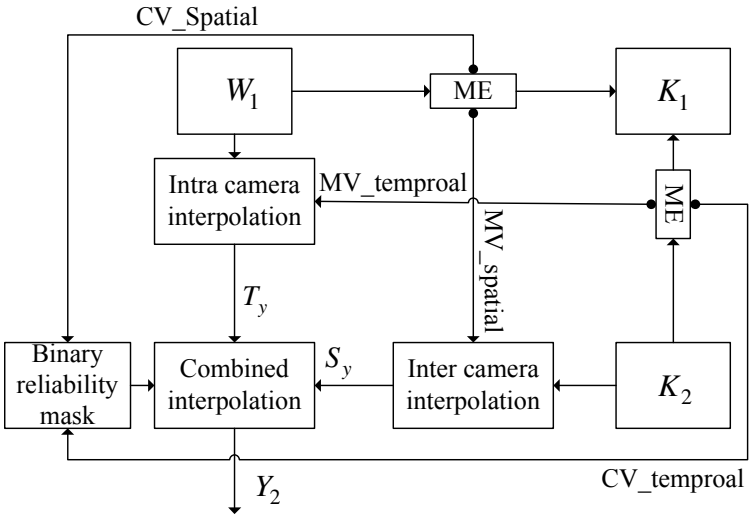

Fig. 7. Side information generation in JSG

Based on these two cost vectors, a binary reliability mask is computed. For each position of the mask, the value determines whether the corresponding block of final side information should be compensated by inter-camera interpolation or intra-camera interpolation. If the value of CV_temporal is higher than CV_spatial then we set 1 in the mask otherwise we set 0. Next, the first estimated version of side information T_y is obtained by performing motion compensation using MV_temporal and frame W_1. We also get the second estimation of side information S_y by similarly utilizing MV_spatial and K_2. Finally, according to the mask vectors, Y_2 is generated by choosing values from T_y or S_y based on the binary reliability mask. Value '1' means the block value is taken from S_y and value '0' means the block value is taken from T_y.

Fig. 8 illustrates the side information comparison generated via three interpolation algorithms for the "Vassar" sequence. Note that the difference in the area marked by a white circle. Neither single intra-camera interpolation nor single inter-camera interpolation can give better output than the joint interpolation.

(a) Original frame (b) Intra-camera interpolation

(c) Inter-camera interpolation (d) Joint interpolation

Fig. 8. Side information comparison for different interpolation in multiview

3. 3D-DCT Transform Domain DMVC Framework (3TD-DMVC)

In Fig. 9, the architecture of the proposed 3TD-DMVC is illustrated. We take one key camera and one WZ camera as an example to illustrate the whole coding process.

In the key camera, all frames are coded via conventional intra-frame coding. In the WZ camera, frames are organized into GOPs with size equal to n. We denote the frames from the WZ camera as $W_{g,t}$, where g represents the index of GOP and t ($1 \le t \le n$) represents the temporal index of the frames inside the GOP. $K_{g,t}$ denotes the corresponding key frame in key camera with the same temporal instant as $W_{g,t}$.

In each WZ camera, $W_{g,1}$, the first frame of each GOP, is intra-frame coded and the rest of the frames $W_{g,2} \ldots W_{g,n}$ are WZ frames and they are coded using transform domain WZ coding with 3D-DCT. These WZ frames $W_{g,2} \ldots W_{g,n}$ are grouped together. Then a (m,m,n-1) cubewise 3D-DCT is applied to $W_{g,2} \ldots W_{g,n}$ (see Section 5.1). After the transformation, the coefficients at the same band decided by the position in every DCT cube will be grouped together to compose the coefficient bands, denoted by $C_{w,k}$ (w indicates the coefficient band for WZ frame and k is coefficient band index with $1 \le k \le m \times m \times (n\text{-}1)$). The process of coefficient band grouping is similar to the one of TDWZ with 2D-DCT [9, 10].

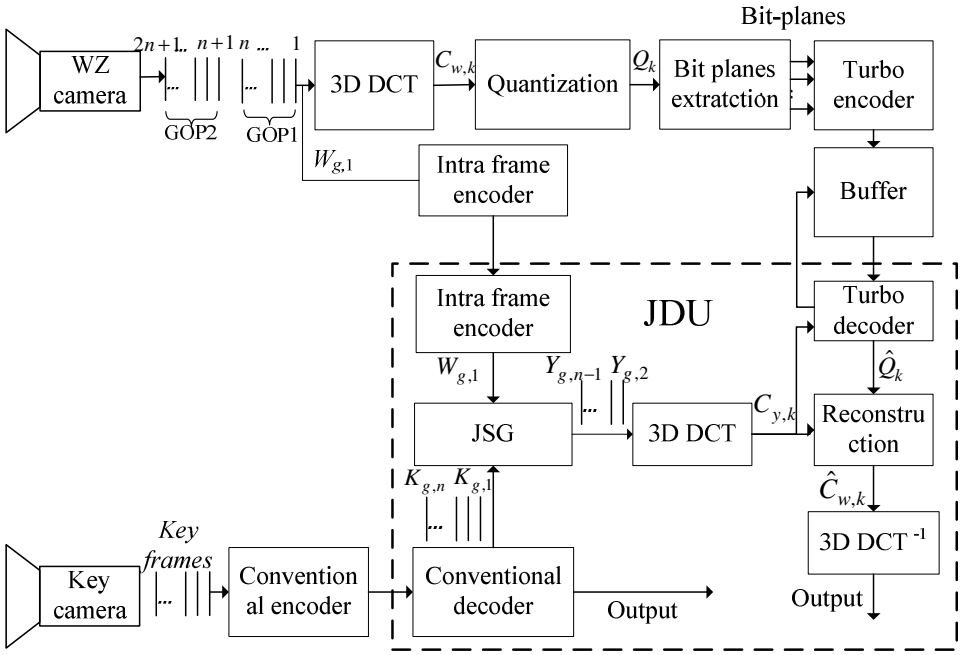

Fig. 9. 3D-DCT transform domain DMVC (3TD-DMVC) architecture

Each coefficient band $C_{w,k}$ is quantized by using a uniform scalar quantizer with 2^{M_k} levels. The quantization level for each coefficient band is determined by the predefined 3D quantization volume (see Section 5.2). After quantization, the quantized symbols Q_k are converted into fixed-length binary code words and the corresponding bit-planes extraction is performed. The bits with same significance in each quantized symbol are blocked together to form bit-planes.

Each bit-plane is then sent to the Slepian-Wolf encoder for encoding. The Slepian-Wolf codec is implemented by a rate-compatible punctured turbo code (RCPT) in combination with a feedback channel. After encoding, the parity bits produced by the turbo encoder are stored in a buffer which transmits a subset of these parity bits to the decoder for decoding.

Inside the Joint Decoding Unit (JDU) at the decoder, $K_{g,1}...K_{g,n}$ from the key camera are firstly decoded. Meanwhile, $W_{g,1}$ is also intra-frame decoded. With $K_{g,1}$ and $K_{g,2}$ and $W_{g,1}$, the corresponding side information $Y_{g,2}$, $Y_{g,3}$,…$Y_{g,n}$ are generated in the JSG. With this side information group $Y_{g,2}…Y_{g,n}$, $m×m×(n-1)$ cubewise 3D-DCT is applied (see Section 5.1).

Similarly as in the WZ camera, the coefficient band $C_{y,k}$ of the side information is grouped and used to help the turbo decoder to decode the bit-plane with the received parity bits. The correlation between $C_{y,k}$ and $C_{w,k}$ is modeled by a Laplacian distribution whose parameters are calculated and used to predict the soft input of the turbo decoder during the decoding. If the BER of current decoded bit-plane is higher than 10^{-3} then the decoding is considered not successful, and the request for more parity bits will be sent back to encoder via feedback channel. The encoder will send more parity bits and the whole decoding is repeated until the current bit-plane is successfully decoded. Note that the feedback channel may not be necessary if the rate control techniques proposed in [23-27] are used. However, in this paper, the 3TD-DMVC is based on the feedback channel in order to make it easily comparable to the previous works in literatures which are also based on the feedback channel.

After all bit-planes are decoded, the quantized symbol $\hat{Q}_k$ can be obtained, with which the reconstruction for coefficient band $\hat{C}_{w,k}$ is performed, following the same model as in the standard DVC by using $E(\hat{C}_{w,k} | Q_k, Y_w, k)$ [10, 11]. After all coefficient bands are reconstructed, the inverse 3D-DCT is performed and $W_{g,2} \ldots W_{g,n}$ are decoded, then the transmission of the next GOP can start.

4. 3D-DCT "Residual" Transform Domain DMVC (3RTD-DMVC)

In Fig. 10, the architecture of the proposed 3RTD-DMVC is illustrated. Similarly to the previous technique, we take one key camera and one WZ camera as an example to illustrate the whole coding process.

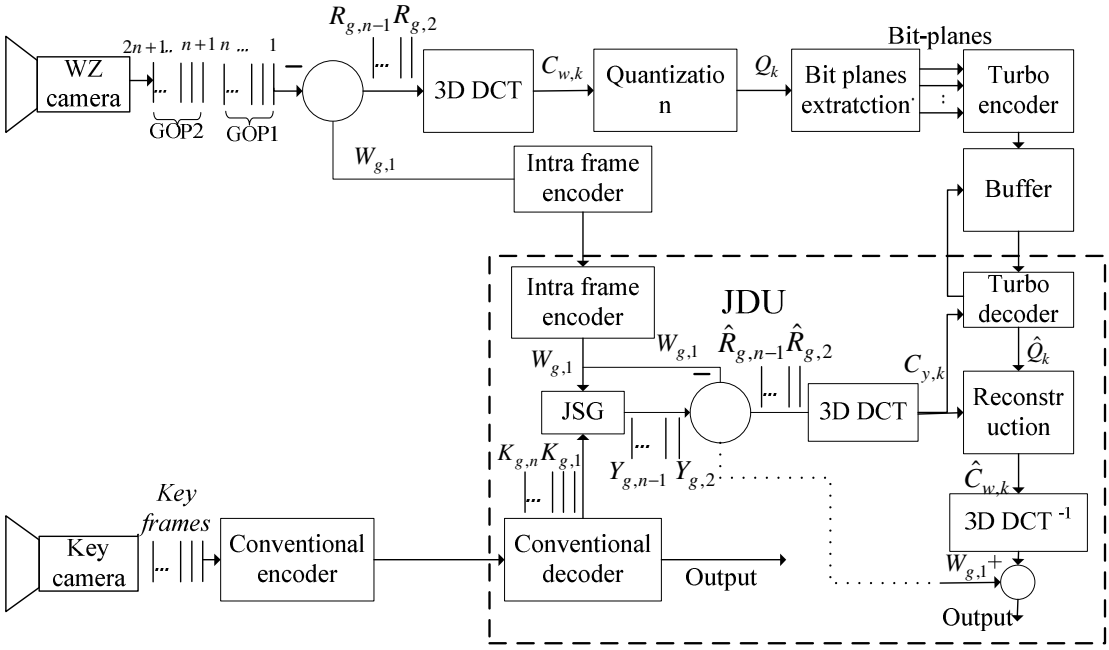

Fig. 10. 3D-DCT "residual" transform domain DMVC (3RTD-DMVC) architecture

In 3RTD-DMVC, the key camera and the WZ camera work similarly as in 3TD-DMVC. The main difference lies in the fact that in each WZ camera, $W_{g,2}\ldots W_{g,n}$ are not directly coded, but the residual frame is coded instead. Residual frames $R_{g,2}\ldots R_{g,n}$ are obtained by making subtraction between $W_{g,2}\ldots W_{g,n}$ and $W_{g,1}$ respectively. These residual frames $R_{g,2}\ldots R_{g,n}$ are grouped together, then an (m,m,n-1) cubewise 3D-DCT is applied. After the transformation, the coefficient bands are grouped, quantized, and sent to the turbo encoder for encoding using bit-planes as previously described.

Inside the Joint Decoding Unit (JDU) at the decoder, the corresponding side information $Y_{g,2}$, $Y_{g,3},\ldots Y_{g,n}$ is generated in the JSG. The residual side information frames $\hat{R}_{g,2}\ldots\hat{R}_{g,n}$ are obtained by making subtraction between $Y_{g,t}$ and $W_{g,1}$. Then, an m×m×(n-1) cubewise 3D-DCT is applied to $\hat{R}_{g,2}\ldots\hat{R}_{g,n}$ and similarly to the encoding process at the WZ camera, the coefficient band $C_{y,k}$ of the residual side information is grouped and used to help the turbo decoder to decode the bit-plane with the received parity bits.

If the BER of the currently decoded bit-plane is higher than 10^{-3} then more parity bits will be sent and the whole decoding process repeats until decoding is finished. After that, by means of the quantized symbol $\hat{Q}_k$ the coefficient band $\hat{C}_{w,k}$ can subsequently be reconstructed. After all coefficient bands have been reconstructed, the inverse 3D-DCT is applied, an addition with $W_{g,1}$ is performed and $W_{g,2}\ldots W_{g,n}$ are reconstructed, then the transmission of the next GOP can start.

5. 3D-DCT processing for WZ frame group

In both the 3TD-DMVC and the 3RTD-DMVC architectures, the WZ frames from the WZ camera are organized into groups for 3D-DCT processing.

5.1 3D-DCT

In current major video coding standards, the 2D-DCT transform is widely used to remove the spatial redundancy while the temporal redundancy is removed by a motion estimation (ME) and prediction process. Since ME has several disadvantages in non-translational motions (zooming, rotation), and it is one of the main complexity factors during encoding, 3D-DCT is proposed in this work as an alternative to ME to exploit both the spatial and the temporal correlation in the video while keeping the complexity low. Since a digital video sequence can be conceived as a three-dimensional signal, it is natural to extend the 2D-DCT algorithm to 3D-DCT by performing a third 1D-DCT in the temporal direction. The algorithm which applies 3D-DCT in video encoding typically involves the following steps:

- Stacking a group of frames;
- Dividing frames into pixel cubes;
- Applying 3D-DCT transformation to every pixel cube.

The forward 3D-DCT is given by

$$Y_{u,v,w} = \sqrt{\frac{8}{MNP}} E_u E_v E_w \times \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{p=0}^{P-1} y_{m,n,p} Cs_{2M}^{(2m+1)u} Cs_{2N}^{(2n+1)v} Cs_{2P}^{(2p+1)w}$$

$$E_u, E_v, E_w = \begin{cases} \frac{1}{\sqrt{2}} & m,n,p = 0 \\ 1 & otherwise \end{cases} \quad (1)$$

$$Cs_J^i = \cos\left(\frac{i\pi}{J}\right)$$

where $y_{m,n,p}$ is the 3D spatio-temporal data element of the *m*th row, *n*th column and *p*th frame, $Y_{u,v,w}$ is the 3D transform domain data element at position *u*,*v*,*w* in the 3D transform space, and *M,N,P* are the dimensions of the data cube.

Furthermore, $Y_{u,v,w}$ can be normalized as

$$\overline{Y}_{u,v,w} = \sqrt{\frac{MNP}{8}} \frac{1}{E_u E_v E_w} Y_{u,v,w} \quad (2)$$

Applying (2) into (1), we obtain

$$\overline{Y}_{u,v,w} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{p=0}^{P-1} y_{m,n,p} Cs_{2M}^{(2m+1)u} Cs_{2N}^{(2n+1)v} Cs_{2P}^{(2p+1)w} \quad (3)$$

The Cs_J^i values are the usual cosine coefficients found in the DCT transform used to compute the transformed coefficient. In the 3D case, the result will be a 3D matrix of values, denoted by $\overline{Y}_{u,v,w}$ in the previous equation. Analogously to the 2D-DCT transform, the inverse 3D-DCT (IDCT) that yields the pixel values in the spatio-temporal domain is given by

$$y_{m,n,p} = \sqrt{\frac{8}{MNP}} \times \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \sum_{w=0}^{P-1} E_u E_v E_w Y_{u,v,w} Cs_{2M}^{(2m+1)u} Cs_{2N}^{(2n+1)v} Cs_{2P}^{(2p+1)w} \quad (4)$$

Furthermore, $Y_{u,v,w}$ can be normalized as

$$\overline{Y}_{u,v,w} = \sqrt{\frac{8}{MNP}} E_u E_v E_w Y_{u,v,w} \tag{5}$$

which, using (4) and (5), yields

$$Y_{m,n,p} = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \sum_{w=0}^{P-1} \overline{Y}_{u,v,w} Cs_{2M}^{(2m+1)u} Cs_{2N}^{(2n+1)v} Cs_{2P}^{(2p+1)w} \tag{6}$$

The 3D-DCT transformation has the following advantages compared to ME:

- It avoids the problems introduced by the ME, mainly the high complexity;
- It is a much simpler algorithm. 3D-DCT can be considered as a series of 1D-DCT transformations in different direction;
- It allows building a low complexity encoder;
- It produces higher reconstruction quality frames.

However, applying 3D-DCT presents the following disadvantages:

- It requires larger memory to store data. Larger pixel cubes will increase the memory requirement correspondingly;
- Compared to the predictive coding with an IPPPI GOP structure, the system output is not frame-by-frame but group-by-group;
- The higher quality output comes at the cost of a moderate loss in compression ratio compared to predictive coding.

Although the application of 3D-DCT in video compression has several drawbacks, it is much more suitable than ME for the WZ video coding. It satisfies the requirement of low complexity encoder and it also exploits the temporal correlation which is not fully exploited in current TDWZ. With 3D-DCT, the TDWZ can achieve higher compression performance.

5.2 3D quantization volume

In order to investigate the RD performance of the proposed framework, we need to setup a quantization volume (3D quantization matrix) to determine the quantization levels for each coefficient band. In most 2D-DCT based coding techniques, such as JPEG and MPEG, a quantization matrix is generally used. However, it is not directly applicable for the situation with 3D-DCT based coding since it cannot cope with variables in temporal axis. Many works addressed how to construct quantization matrix for 2D-DCT [28-29], but so far few works have addressed the construction of an effective 3D quantization volume for 3D-DCT. In the 3D-DCT transformation, it is observed that the dominant coefficients (significant coefficients including DC and part of AC coefficients) are spread along the major axes (x,y,z) of the coefficients cube, as shown in the shadow area in Fig. 11. The dominant coefficients typically contain a high percentage of the total energy.

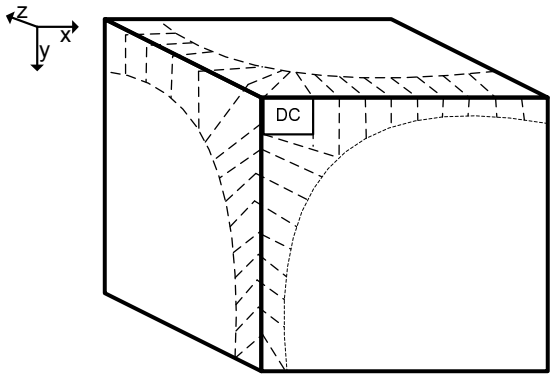

Fig. 11. Distribution of dominant coefficients in DCT coefficient cube

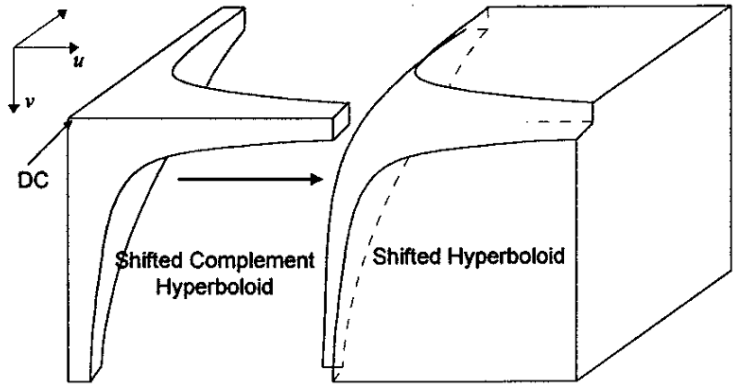

Fig. 12. Shape of the distribution of the dominant coefficients [30]

Let $C(u,v,w)$ denote the coefficient located on (u,v,w) in the cube where u, v and w are the coordinates on three major axes (x,y,z). It can be found that coefficients located farther from the DC (C (0,0,0)) tend to be less significant and should be more coarsely quantized. Based on these observations, a technique to build up quantization volume for 3D-DCT is proposed in [30]. First, the distribution of dominant coefficients is shaped as a complement shifted hyperboloid, as shown in Fig. 12, which can be described by a function $f(u,v,w) \leq C$. Here, C is an arbitrary constant to determine the size of shape and $f(u,v,w)=u*v*w$.

Then, the quantization volume containing quantization values for each coefficient band is build up as follows:

$$q(u,v,w) = \begin{cases} A_i \left(1 - \dfrac{e^{-\beta_i (u+1)(v+1)(w+1))}}{e^{-\beta_i}} \right) + 1; & for \quad f(u,v,w) \leq C \\ A_0 \left(1 - e^{-\beta_0 (u+1)(v+1)(w+1))} \right); & for \quad f(u,v,w) > C \end{cases} \tag{7}$$

where $q(u,v,w)$ is the quantization step for coefficient $C(u,v,w)$, A_i and A_0 are the initial amplitude, respectively, which are set to the maximum quantization value. β_i and β_o denote the decay speed with respect to the A_i and A_0, respectively, which make quantization values smaller inside the selected shape and bigger outside the shape. This function satisfies the requirement of quantizing those dominant coefficients located inside the shape with little distortion, thus

providing a better quality for the reconstructed frames, and quantizing high frequency coefficients outside the shape to zero. Parameters A_i , A_o, C, β_i and β_o can be determined experimentally and the quantization volume can be built up afterwards.

In our work, a maximum of 8 bits (256 levels) is used to quantize the DCT coefficients. In order to do that, the coefficients are required to be scaled down as follows

$$S_q = \left\lceil \frac{\max(C(u,v,w))}{255} \right\rceil \tag{8}$$

where the numerator represents the maximum value of DCT coefficients, $\lceil x \rceil$ denotes the smallest integer bigger than or equal to x and S_q denotes the scaling factor. The next step is to build up the quantization volume. A_i and A_o are set to 255 since the maximum value after scaling down is 255.

To determine the optimal range of C, β_i and β_o, we investigate several sequences first and test the parameter influence by fixing two parameters and varying another one. By means of several experiments, it has been found that the C parameter has the most direct influence on the coarseness of the quantization volume. With this results, we set β_i = 0.001 and β_o = 0.03 and vary the value of C to generate different quantization volumes, which gives different rate distortion points in the simulation. Smaller C values denote coarser quantization and vice versa. With one quantization volume, we can quantize the coefficient $C(u,v,w)$ as:

$$Q(u,v,w) = round\left(\frac{|C(u,v,w)|}{S_q * q(u,v,w)} \right) \tag{9}$$

$$Q_l = \left\lceil \frac{255}{q(u,v,w)} \right\rceil \tag{10}$$

where *q(u,v,w)* corresponds to the quantization step defined in quantization volume, *Q*(*u,v,w*) is the quantized symbol of coefficient *C(u,v,w)* and Q_l is the quantization level needed. $\lceil x \rceil$ denotes the smallest integer bigger than or equal to *x*. The *Q(u,v,w)* will be converted into a binary symbol with $\lceil \log_2 Q_l \rceil$ bits. For the coefficients with Q_l<4, we do not quantize them but use side information to replace it. Table 1 shows an example of an 8×8×8 3D-DCT quantization volume with *C*=15, β_i=0.001 and β_o=0.03 in which only the quantization step for each coefficient band and the corresponding quantization levels are computed according to (10). The table shows that increasingly coarser quantization levels are assigned to the places increasingly far from the dominant coefficients, i.e., the ones in the lower right part of the matrices, especially in tables with a high *w* value. As expected, the table corresponding to the highest *w* includes the largest amount of coarse quantization levels.

Table 1. Example of 3D-DCT quantization volume (C=15, β_i=0.001, β_0=0.03) used in the 3TD-DMVC and 3RTD-DMVC

W=0

U\V	0	1	2	3	4	5	6	7
0	1	2	2	2	2	2	2	2
1	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	120	131
3	2	2	2	2	2	131	145	158
4	2	2	2	2	135	152	166	179
5	2	2	2	131	152	169	183	195
6	2	2	120	145	166	183	197	208
7	2	2	131	158	179	195	208	218

W=1

U\V	0	1	2	3	4	5	6	7
0	2	2	2	2	2	2	2	2
1	2	2	2	2	2	131	145	158
2	2	2	2	131	152	169	183	195
3	2	2	131	158	179	195	208	218
4	2	2	152	179	199	213	224	232
5	2	131	169	195	213	226	235	241
6	2	145	183	208	224	235	242	247
7	2	158	195	218	232	241	247	250

W=2

U\V	0	1	2	3	4	5	6	7
0	2	2	2	2	2	2	120	131
1	2	2	2	131	152	169	183	195
2	2	2	142	169	189	205	217	226
3	2	131	169	195	213	226	235	241
4	2	152	189	213	229	238	245	249
5	2	169	205	226	238	246	250	252
6	120	183	217	235	245	250	252	254
7	131	195	226	241	249	252	254	255

W=3

U\V	0	1	2	3	4	5	6	7
0	2	2	2	2	2	131	145	158
1	2	2	131	158	179	195	208	218
2	2	131	169	195	213	226	235	241
3	2	158	195	218	232	241	247	250
4	2	179	213	232	243	249	252	253
5	131	195	226	241	249	252	254	255
6	145	208	235	247	252	254	255	255
7	158	218	241	250	253	255	255	255

W=4

U\V	0	1	2	3	4	5	6	7
0	2	2	2	2	135	152	166	179
1	2	2	152	179	199	213	224	232
2	2	152	189	213	229	238	245	249
3	2	179	213	232	243	249	252	253
4	135	199	229	243	250	253	254	255
5	152	213	238	249	253	254	255	255
6	166	224	245	252	254	255	255	255
7	179	232	249	253	255	255	255	255

W=5

U\V	0	1	2	3	4	5	6	7
0	2	2	2	131	152	169	183	195
1	2	131	169	195	213	226	235	241
2	2	169	205	226	238	246	250	252
3	131	195	226	241	249	252	254	255
4	152	213	238	249	253	254	255	255
5	169	226	246	252	254	255	255	255
6	183	235	250	254	255	255	255	255
7	195	241	252	255	255	255	255	255

W=6

U\V	0	1	2	3	4	5	6	7
0	2	2	120	145	166	183	197	208
1	2	145	183	208	224	235	242	247
2	120	183	217	235	245	250	252	254
3	145	208	235	247	252	254	255	255
4	166	224	245	252	254	255	255	255
5	183	235	250	254	255	255	255	255
6	197	242	252	255	255	255	255	255
7	208	247	254	255	255	255	255	255

W=7

U\V	0	1	2	3	4	5	6	7
0	2	2	131	158	179	195	208	218
1	2	158	195	218	232	241	247	250
2	131	195	226	241	249	252	254	255
3	158	218	241	250	253	255	255	255
4	179	232	249	253	255	255	255	255
5	195	241	252	255	255	255	255	255
6	208	247	254	255	255	255	255	255
7	218	250	255	255	255	255	255	255

6. Results and discussion

In the simulation, the performance of PD-DMVC, 2TD-DMVC, and the proposed 3TD-DMVC and 3RTD-DMVC are presented together for comparison purposes. In addition, H.263+ intra-frame coding which is commonly used in two-way conversational video transmission is added as reference. The two multiview video sequences "Ballroom" and "Vassar" from Mitsubishi Electric Research Laboratories (MERL) with resolution level of 320x240 and frame rate of 30 fps are used.

For both sequences, camera 1 is used as the key camera and camera 0 is used as the WZ camera. The key frames from the key camera and the intra frames from the WZ camera are assumed to be reconstructed perfectly in the JDU. For camera 0, 48 frames are considered. The GOP size is set to 4 thus there are 36 frames to be compressed as WZ. Simulation results show the average PSNR performance of the Y component of these 36 WZ frames.

The block size is 4 for 2D-DCT and the cube size is 4×4×3 for 3D-DCT. The quantization volume for 3D-DCT is designed with parameters β_i=0.001, β_o=0.3 and C varies in the set [1 2 3 4 5 6 7] in order to obtain seven different RD points [30]. For 2TD-DMVC, seven quantization matrices from [9] are used. For PD-DMVC, 2^M=[2 4 8 16] quantization levels are used.

For all DMVC architectures, the turbo codec is composed by two identical constituent convolutional encoders of rate 1/2 with constraint length of 4 and polynomial generator g = (13, 11). The puncturing period is set to 8 in order to provide the coding rate of {0,8/9,8/10,8/11…1/3}.

For the multiview applications with limited computation complexity at the encoder, only multiview video coding Case 1 (separate encoding and separate decoding) and Case 3 (DMVC)

are considered as suitable solutions. For Case 1, all cameras have to perform intra-frame coding, e.g., H.263+ intra-frame coding, for the purpose of low complexity.

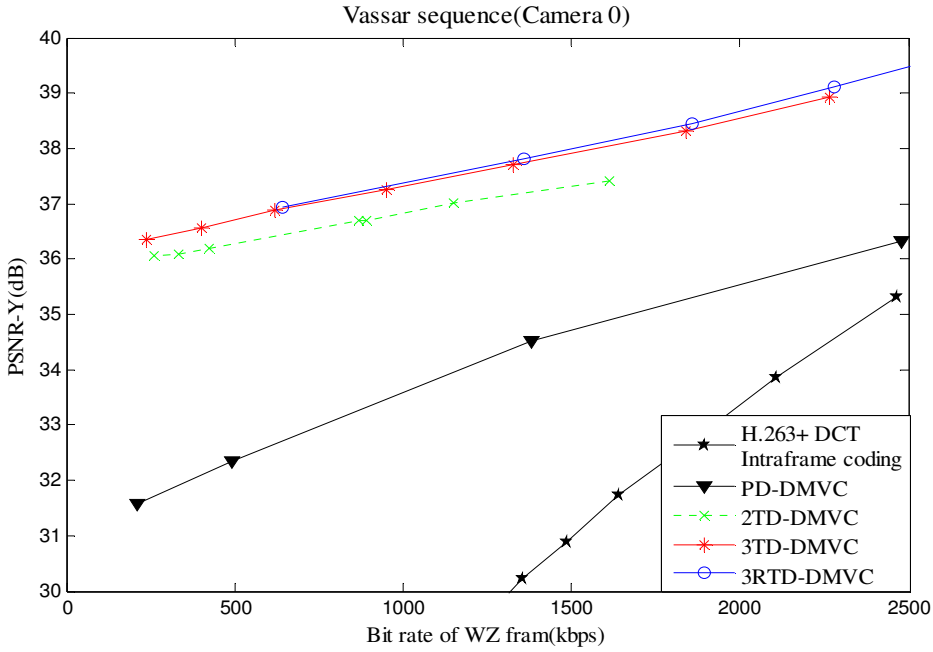

Fig. 13. R-D performance of proposed transform domain DMVC for "Vassar" sequence

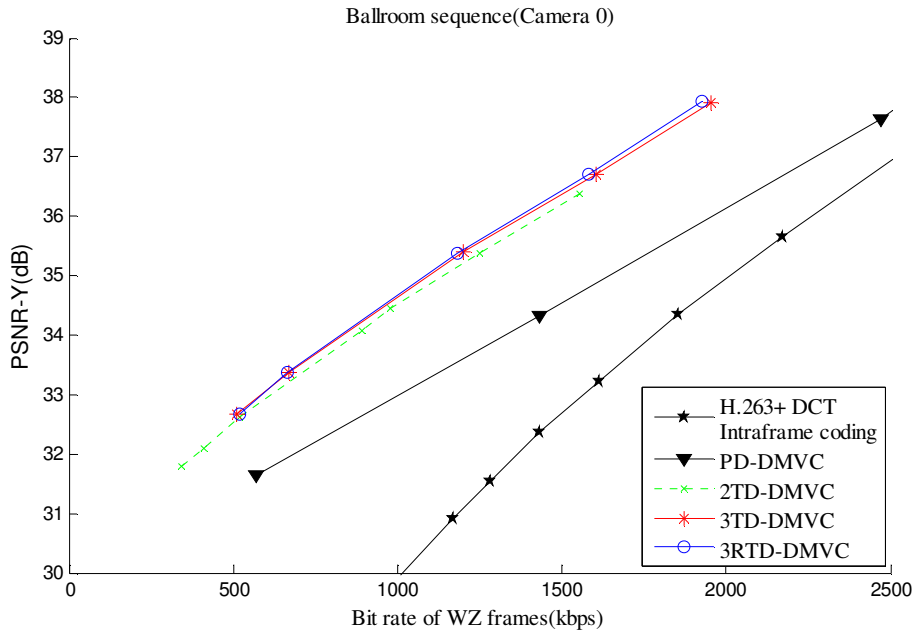

Fig. 14. R-D performance of proposed transform domain DMVC for "Ballroom" sequence

Fig. 13 and 14 show the advantage of DMVC over Case 1. It is clear that all the introduced DMVC variants significantly outperforms the H.263+ DCT based intra-frame coding whilst

DMVC keeps the complexity similar to the one of Case 1. Moreover, it is clear that the transform domain DMVC further improves the system performance significantly compared to PD-DMVC. A clear gap can be observed between PD-DMVC and 2TD-DMVC, up to 1.5 dB PSNR for the “Ballroom” sequence and 4.4 dB for the “Vassar” sequence.

Within the transform domain DMVC solutions, 3TD-DMVC outperforms the 2TD-DMVC due to the exploitation of the temporal correlation for the WZ frames. For the “Vassar” sequence, 3TD-DMVC provides PSNR gain up to 0.6 dB. In the case of “Ballroom”, the PSNR is improved up to about 0.3 dB. The “Ballroom” sequence has a lower correlation in temporal direction than the “Vassar” sequence due to the presence of fast motion and thus 3D-DCT based algorithms improve the PSNR of the “Ballroom” sequence less significantly than that of the “Vassar” sequence.

It can also be observed that the proposed 3RTD-DMVC can further enhance the system performance compared to the 3TD-DMVC. The improvement is about 0.2 dB for the “Vassar” sequence and 0.02 dB for the “Ballroom” sequence. For the sequence with fast motion (low temporal correlation), the 3RTD-DMVC performs similarly as 3TD-DMVC. For this type of sequence, the residual value between pixels tends to be very large thus the residual coding technique cannot show great advantages in this case.

The complexity of the encoder is gradually increased with the described DMVC solutions. The simplest case is the PD-DMVC, followed by the 2TD-DMVC. The encoder complexity of the 3TD-DMVC is slightly higher than that of previous two solutions and is similar to that of the conventional intra-frame coding algorithm. The 3RTD-DMVC slightly increases the complexity of the encoder compared to the 3TD-DMVC due to the subtraction needed to obtain the residual. However, its encoder is still a low complexity one.

7. Conclusion

The DMVC based on the WZ video coding principle is suitable for the multiview video applications with encoders limited in computational complexity. The majority of the previously proposed DMVC techniques are based on PDWZ or TDWZ with 2D-DWT or 2D-DCT. At the expense of a small complexity increase at the encoder, mainly due to the 3D-DCT computation, this paper showed that the proposed 3D-DCT transform domain WZ video coding architectures, named 3TD-DMVC and 3RTD-DMVC, can provide significant improvements of the compression performance. Simulation results have shown the improvement of the proposed DMVC architectures over the PD-DMVC and 2TD-DMVC, which is significant especially in case of video sequences which present slow motion characteristics. The complexities of the encoders of the proposed DMVC architectures are low and comparable to those of conventional DCT intra-frame coding. For multiview video applications that require a low complexity encoder, the proposed DMVC architectures seem to achieve one of the best tradeoffs available in literature so far. Future work will be devoted to analyze the performance of the proposed techniques in a larger variety of settings and to investigate the impact of each parameter on the overall performance.

References

[1] R. S. Wang and Y. Wang, "Multiview video sequence analysis, compression, and virtual viewpoint synthesis", *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 10, pp. 397-410, 2000.

[2] K. Muller, A. Smolic, K. Dix, P. Merkle, P. Kauff, and T. Wiegand, "View synthesis for advanced 3D video systems", *Eurasip Journal on Image and Video Processing,* article ID 438148, vol. 2008, 2008.

[3] S. Yea and A. Vetro, "View synthesis prediction for multiview video coding", *Signal Processing: Image Communication,* vol. 24, pp. 89-100, 2009.

[4] X. Li, D. Zhao, S. Ma, and W. Gao, "Fast disparity and motion estimation based on correlations for multiview video coding", *IEEE Transactions on Consumer Electronics,* vol. 54, pp. 2037-2044, 2008.

[5] L. F. Ding, P. K. Tsung, S. Y. Chien, W. Y. Chen, and L. G. Chen, "Content-aware prediction algorithm with inter-view mode decision for multiview video coding", *IEEE Transactions on Multimedia,* vol. 10, pp. 1553-1564, 2008.

[6] P. Merkle and K. Muller, "Efficient prediction structures for multiview video coding", *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 17, pp. 1461-1473, 2007.

[7] X. Cao, Y. Liu, and Q. Dai, "A flexible client-driven 3DTV system for real-time acquisition, transmission, and display of dynamic scenes", *Eurasip Journal on Advances in Signal Processing,* Article ID 351452, vol. 2009, 2009.

[8] A. Smolic, K. Mueller, N. Stefanoski, J. Osteraiann, A. Gotchev, G. B. Akar, G. Triantafyllidis, and A. Koz, "Coding algorithms for 3DTV - a survey", *IEEE Trans. Circuits and Systems for Video Technology,* vol. 17, pp. 1606-1621, 2007.

[9] A. Aaron, S. Rane, E. Setton, and B. Girod, "Transform-domain Wyner-Ziv codec for video", in *Proc. of SPIE*, pp. 520-528, 2004.

[10] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding", in *Proceedings of the IEEE (Special Issue on Video Coding and Delivery)*, pp. 71-83, 2005.

[11] A. Aaron, R. Zhang, and B. Girod, "Wyner-Ziv coding of motion video", in *Asilomar Conf. on Signals, Systems and Computers*, pp. 240-244, 2002.

[12] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: architecture, techniques and evaluation", in *Int. Picture Coding Symposium,* Lisboa, Portugal, 2007.

[13] F. Pereira, C. Brites, and J. Ascenso, "Distributed video coding: basics, codecs, and performance", in *Distributed Source Coding,* Academic Press, Boston, pp. 189-245, 2009.

[14] X. Artigas, E. Angeli, and L. Torres, "Side information generation for multiview distributed video coding using a fusion approach", in *The 7th Nordic Signal Processing Sym.*, pp. 250-253, 2007.

[15] M. Morbee, L. Tessens, J. Prades-Nebot, A. Pizurica, and W. Philips, "A distributed coding-based extension of a mono-view to a multi-view video system", in *Proc. 3DTV Conference*, 2007.

[16] M. Morbee, L. Tessens, H. Quang Luong, and J. Prades-Nebot, A. Pizurical, and W. Philips, "A distributed coding-based content-aware multi-view video system", in *ACM/IEEE Int. Conf. Distributed Smart Cameras*, pp. 355-362, 2007.

[17] X. Guo, Y. Lu, F. Wu, W. Gao, and S. Li, "Distributed multi-view video coding", *Proceedings of SPIE - The International Society for Optical Engineering,* 2006.

[18] X. Guo, Y. Lu, F. Wu, D. Zhao, and W. Gao, "Wyner-Ziv-based multiview video coding", *IEEE Trans. Circuits and Systems for Video Technology,* vol. 18, pp. 713-724, 2008.

[19] C. Guillemot, F. Pereira, L. Torres, T. Ebrahimi, R. Leonardi, and J. Ostermann, "Distributed monoview and multiview video coding", *IEEE Signal Processing Magazine,* vol. 24, pp. 67-76, 2007.

[20] A. Aaron, D. Varodayan, and B. Girod, “Wyner-Ziv residual coding of video”, in *The 25th Proc. Picture Coding Symposium*, 2006.

[21] M. B. Badem, H. Kodikara Arachchi, S. T. Worrall, and A. M. Kondoz, “Transform Domain Residual Coding Technique for Distributed Video Coding”, in *Proc. of Intl. Picture Coding Symposium*, 2007.

[22] Y. Li, X. Ji, D. Zhao, and W. Gao, “Region-based fusion strategy for side information generation in DMVC”, in *Proceedings of SPIE - The International Society for Optical Engineering*, 2008.

[23] C. Brites and F. Pereira, “Encoder rate control for transform domain Wyner-Ziv video coding”, in *IEEE Int.Conf. Image Processing*, vol. 2, pp. 5 -8, 2007.

[24] D. Kubasov, K. Lajnef, and C. Guillemot, “A hybrid encoder/decoder rate control for Wyner-Ziv video coding with a feedback channel”, in *IEEE 9th Workshop on Multimedia Signal Processing*, pp. 251-254, 2007.

[25] A. Roca, M. Morbee, J. Prades-Nebot, and E. J. Delp, “Rate control algorithm for pixel-domain Wyner-Ziv video coding”, in *Proceedings of SPIE - The International Society for Optical Engineering,* 2008.

[26] M. Morbee, J. Prades-Nebot, A. Pizurica, and W. Philips, “Rate allocation algorithm for pixel-domain distributed video coding without feedback channel”, in *IEEE Int. Conf. Acoustics, Speech and Signal Processing,* Honolulu, HI, vol. 1, pp. 521-524, 2007.

[27] M. Morbee, J. Prades-Nebot, A. Roca, A. Pizurica, and W. Philips, “Improved pixel-based rate allocation for pixel-domain distributed video coders without feedback channel”, in *Lecture Notes in Computer Science*. vol. 4678, pp. 663-674, 2007.

[28] A.B. Watson, “DCT quantization matrices visually optimized for individual images”, in *Proceedings of SPIE - The International Society for Optical Engineering,* pp. 1913-14, 1993.

[29] A.J. Ahumada, H. Peterson, “Luminance-model-based DCT quantization for color image compression”, in *Proc. of Human Vision, Visual Processing, and Digital Display III*, pp. 365-374, 1992.

[30] M. C. Lee, R. K. W. Chan, and D. A. Adjeroh, “Quantization of 3D-DCT coefficients and scan order for video compression”, *Elsevier Journal of Visual Communication and Image Representation,* vol. 8, pp. 405-422, 1997.