3D VIDEO CODING USING MOTION INFORMATION AND DEPTH MAP

Fei CHENG, Jimin XIAO, Tammam TILLO

Xi'an Jiaotong-Liverpool University (XJTLU)
Department of Electrical and ElectronicEngineering
Suzhou, China
{firstname.lastname}@xjtlu.edu.cn

ABSTRACT

In this paper, a motion-information-based 3D video coding method is proposed for the texture plus depth 3D video format. The synchronized global motion information of camcorder is sampled to assist the encoder to improve its rate-distortion performance. This approach works by projecting temporal previous frames into the position of the current frame using the depth and motion information. These projected frames are added in the reference buffer as virtual reference frames. As these virtual reference frames are more similar to the current frame than the conventional reference frames, the required residual information is reduced. The experimental results demonstrate that the proposed scheme enhances the coding performance in various motion conditions including rotational and translational motions.

Index Terms— 3D video coding, global motion information, depth map, reference buffer

1. INTRODUCTION

To reduce the redundancy between different viewpoints of the 3D video, besides the commonly used conventional temporal prediction, inter-view prediction [1] is also introduced in the Multi-view Video Coding (MVC) [2] extension of the Advanced Video Coding standard [3]. Though MVC has enormously improved the compression performance of multi-view video, it still requires a bit rate that is proportional to the number of views [2]. Furthermore, the configuration and arrangement of conventional multi-view camera system are required to be fixed, which puts constraints on the post-processing stage and consequently limits potential applications.

A depth map, which represents the distance from the objects in the scene to the capturing camcorder, and its aligned texture, have been exploited to describe 3D scenes. Multi-view Video plus Depth (MVD) format is a promising way to represent 3D video content, and recent extensions supporting the MVD format have been introduced [4, 5]. With the MVD format, only a small number of texture views associated with their depth views are required, and the rates for the texture and depth views should be properly allocated [6]. At the decoder or display side, Depth-Image-Based Rendering (DIBR) [7, 8] is used to synthesize additional viewpoint video.

In the texture plus depth map 3D video coding scheme, depth map is represented as a gray-scale image, which is encoded independently, as shown in Fig.1. However, a texture and its corresponding depth map describe the features of the same scene in terms of content and distance respectively, thus the correlation between them could be exploited by an encoder to reduce the redundancy.

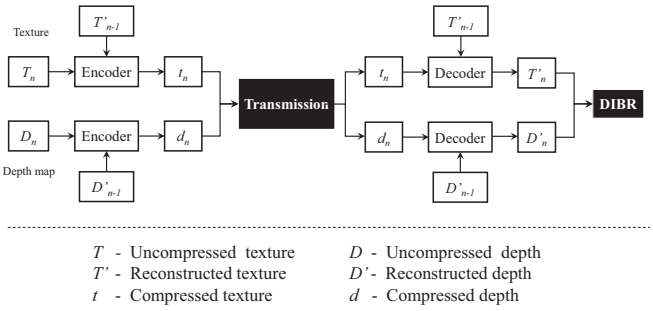

Fig. 1. The texture plus depth map 3D video CODEC scheme

In [9], the coding performance of depth maps is improved by taking into account the Motion Vector (MV) of texture. This can reduce the time of the Motion Estimation (ME) for depth map encoding due to the reduced coding complexity. Furthermore, it is proposed to add the 3D search to expand the ME of depth map. However, only the information from texture was used to assist the depth map encoding, and the coding performance of texture is not improved.

In many video capturing scenarios, the camcorder is not static, and the global motion, generally, causes more ME time, or even leads to the failure of the ME process. This results in the use of Intra-block coding. A great deal of research has been devoted to reduce global motion, but these works only

This work was supported by National Natural Science Foundation of China (No.61210006, No.60972085).

relies on texture information [10, 11]. It is worth motioning that a considerable amount of time is needed in order to determine the global motion to compensate it. Furthermore, this process is prone to errors.

With the advances of the sensor technology, the physical motion of a camcorder can be measured directly from some motion sensors. The accelerometer and the gyroscope are widely integrated in different kinds of smart mobile devices, such as smart phones, tablet PCs and some camcorders.

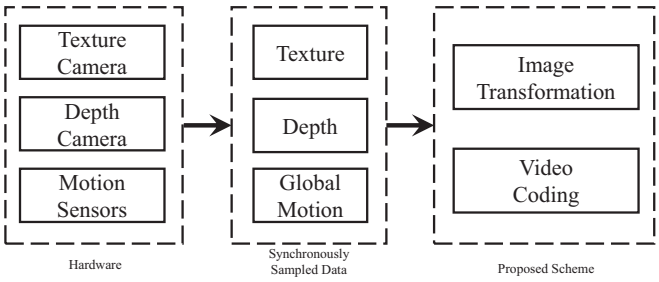

Fig. 2. The diagram the proposed 3D video coding scheme using motion and depth information

In this paper, we propose a novel texture plus depth map 3D video coding scheme. As shown in Fig.2, the synchronously sampled global motion information of a camcorder and its depth map have been exploited to improve the performance of texture coding. The contribution of this paper is many-fold. Firstly, the camcorder global motion is measured directly from motion sensors by using a data fusion method. Therefore, it is less time consuming and more reliable than conventional global motion estimation method.

Secondly, the depth map is used in a novel way to improve texture coding performance. Conventionally, the depth map is only used by DIBR. In this paper, depth map is exploited to project a previous texture frame to the position of the current frame. Therefore, the virtual image projection is employed in temporal domain instead of the spatial domain, which is a key contribution of this paper. The view transformation between a previous frame and the current frame can be obtained from the camcorder global motion and parameters of camcorder (such as view angle, zoom-level and resolution).

Thirdly, we add virtual frames in the reference buffer list before encoding the current frame as shown in Fig.3. Consequently, the new reference buffer list improves the coding performance. Since the newly added reference frame is more similar with the current coding frame, the residual information is reduced. Hence, the coding performance is improved. It is worth mentioning that the new reference frame is added into reference buffer list instead of replacing other existing reference frame. Therefore, even if the depth and motion information are not accurate, the proposed scheme could still guarantees coding performance gain.

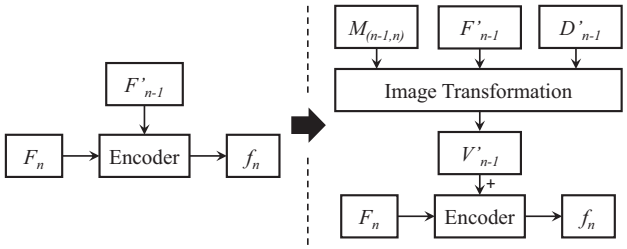

Fig. 3. The virtual reference is exploited for encoding the current frame

We developed hardware prototype to simulate two types of camcorder global motion and to produce our own sequences. In order to simplify the experiment, we test the proposed method using the H.264/AVC standard. The experimental results demonstrate that the proposed scheme can improve the coding performance compared to H.264/AVC, with an average PSNR gain of 0.49 dB. The proposed method is not limited to H.264/AVC platform and it could be applied in H.264/AVC based ATM [12] and HEVC based HTM [13]. Since depth data has been widely used in many areas such as object recognition, the texture video coding can benefit from the transmitted depth information. Therefore, in some 2D video applications with depth information available, the proposed scheme can also be adopted.

The rest of this paper is organized as follows. In Section 2, the details of the proposed scheme are presented. After this, the experimental methods of the proposed scheme and the results are presented in Section 3. Finally, Section 4 concludes this work.

2. PROPOSED SCHEME

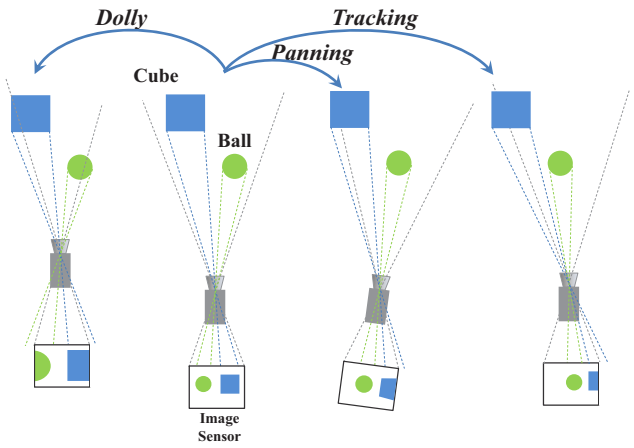

Fig. 4. The impact of camcorder global motion on imaging

In many video capturing scenarios, the camcorder is not

static and the content of the image changes with the camcorder global motion. According to the imaging principle, the impact of camcorder motion on images can be pictorially presented in Fig. 4. In this example, a cube and a ball as examples are captured by a camcorder, then, the camcorder captures a new image after dolly, panning and tracking. With the dolly motion, the camcorder moves forwards, then two objects are enlarged with different scales. The scale of the ball is bigger as it gets closer to the camcorder. When the camcorder is panning, the position of each object is shifted, at the same time the shape of the cube is distorted from rectangle into a trapezoid. For camcorder tracking, the position of each object is shifted, meanwhile the relative distance between them is also changed. If the motion information is known, the previous image can be projected to the new position of the camcorder, and the projected image will be similar to the new captured image.

Using the same principle, textures and depth maps of previous frames in a video sequence are exploited to project the previous frames to the current position. This can reduce the difference between each virtual frames and the current frame. Subsequently, the virtual frames are added in the reference buffer list as virtual reference frames. As a consequence, the content change of the image due to the camcorder motion is reduced. Therefore, ideally, the differences between virtual frame and current frame are only due to the moving objects and newly appeared objects and some small changes of light condition.

In summary, two key procedures have to be implemented in order to achieve the proposed scheme. Firstly, virtual reference frames need to be projected from previous frames. Secondly, these reference frames need to be inserted into the reference buffer list of the video encoder. Similarly, the decoder will process the buffer list in the same way. In addition, the additional data related to the proposed scheme, including global motion and camcorder parameters, will be transmitted to the video decoder side. The data size is much smaller than the video sequences, but this data can benefit many applications, such as image deblurring and background extraction.

The details of each procedure are introduced as follows.

2.1. Virtual Reference Frames Projection

Each pixel in the image needs to be converted from a 2D point into 3D coordinate space for 3D projection. First, the 3D coordinate system needs to be defined as shown in Fig.5. The x-axis and the y-axis are two dimensions parallel to the image, while the z-axis represents the depth. Let $\mathbf{P} = [w, h]$, where w and h represent the horizontal and vertical coordinate of a pixel in the image, while d is the corresponding depth. The 3D homogeneous coordinate of a pixel can be projected by:

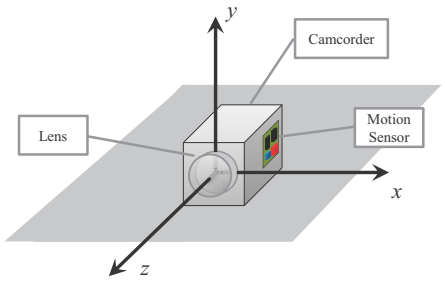

Fig. 5. The 3D coordinate system definition for the camcorder

$$\mathbf{C} = [x, y, z, 1] = \left[K(\frac{W}{2} - w) \cdot d, K(\frac{H}{2} - h) \cdot d, d, 1\right], \tag{1}$$

where W and H are horizontal and vertical resolution of the image respectively. K is the intrinsic parameters of the camcorder, which is represented as:

$$K = \frac{\tan(\phi_w)}{W} = \frac{\tan(\phi_h)}{H}, \tag{2}$$

where ϕ_w and ϕ_h are the horizontal and vertical angles of the view respectively. The 4×4 projective transformation matrix is represented by $\mathbf{T}$, which describes the translation and rotation from the previous view to the current view. The new coordinate of a pixel on the virtual frame can be obtained by using:

$$\mathbf{C}_v = \mathbf{C} \times \mathbf{T} = [x_v, y_v, z_v, 1]. \tag{3}$$

The 3D coordinate of each pixel in the virtual view has to be inversely converted to 2D coordinate in the image:

$$\mathbf{P}_v = [w_v, h_v] = \left[\frac{W}{2} - \frac{x_v}{Kz_v}, \frac{H}{2} - \frac{y_v}{Kz_v}\right]. \tag{4}$$

Finally, each pixel on the previous frame can be transformed to a new 2D coordinate in the image. However, some of them might be located outside of the image and some of them might not be integers, which leads to some holes. Therefore, an interpolation algorithm is utilized to fill holes and smooth the virtual image, which is represented as $\mathbf{V}$.

2.2. Reallocation of Reference Buffer List

The reference buffer list plays an important role for video coding. The reallocation of the reference buffer list is flexible enough to adapt to different configurations of applications. Fig. 6 describes one example of the reallocation. The number of real reference frames is one, which is F'_{n-1}, meanwhile, a virtual frame V'_{n-1} that projected from F'_{n-1} is inserted to the reference buffer list.

It is worth mentioning that some popular codecs (such as HEVC and H.264/AVC) use quarter-pixel motion estimation

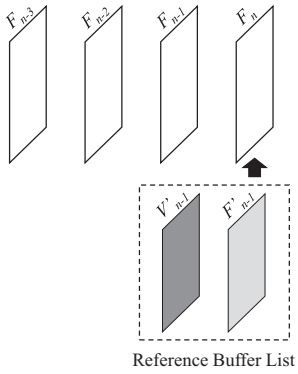

Fig. 6. An example of the reallocation of reference buffer list

and motion prediction. Their reference buffer list stores one or more up-sampled reconstructed frames. Therefore, before the virtual reference frames are inserted into the reference buffer list, they have to be up-sampled using the same algorithm in the encoder.

3. EXPERIMENTAL METHOD AND RESULTS

To the best of our knowledge, there are no standard sequences where the proper (not estimated) global motion information are available. Consequently, we decided to produce some sequences with synchronously sampled global motion information using our platforms. We tested the proposed scheme using the sequences we produced. These sequences are available for download at http://www.mmtlab.com/3dvcmb.

3.1. Data Acquisition and Prototypes

As any motion of a rigid body can be composed of rotation and translation, we developed two prototypes to examine the proposed scheme under rotational and translational motion cases respectively.

Ideally, the motion information should be obtained from motion sensors. We exploited a sensor integrating a 3-axis accelerometer and a 3-axis gyroscope.

As the sampling rate of the sensor data is much faster than video recording, several sensor data slices can be obtained between two video frames. After this, the Kalman filter [14] was used to fuse the data such as angular velocity and acceleration in order to obtain the rotation angle and translational distance. Fig.7 shows the timing of data sampling.

Fig.8 presents the prototype for rotational motion test. The motion sensor is InvenSense MPU6050. A Microsoft HD WebCam is employed as the camcorder. Depth camera is not used in this test because the projection of pure rotational motion is not affected by the depth information.

Fig.9 shows the prototype for translational motion test. In order to obtain smooth and accurate translational motion

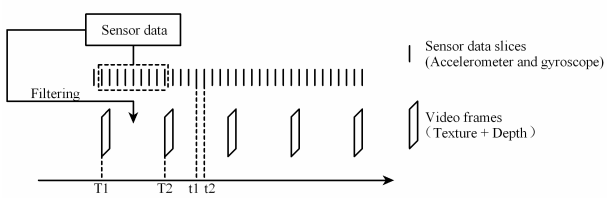

Fig. 7. Timing of sensor data and video sampling

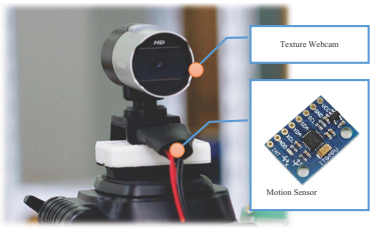

Fig. 8. The customized prototype for the rotational motion

information, we developed a programmable track slider as shown in Fig.9. A shaft encoder is employed as the motion sensor to get the accurate translational distance. A Balser acA640-90gc camera is used to capture texture video, while the depth camera is Mesa Imaging SwissRanger SR4000.

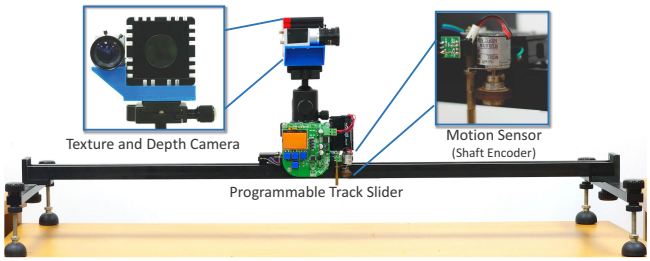

Fig. 9. The customized prototype for the translational motion

3.2. Experiments and Results

In the rotation experiment, the image resolution is VGA (640×480) @ 25 fps. The video sequence is captured when the camcorder was rotating along the y-axis clockwise, which is panning motion. The average angle of rotation per frame is around 1.3 degree, while each angle is recorded respectively. The total number of frames is 15. Fig.10 shows an example of this experiment. To evaluate the proposed scheme, the experimental software was modified from H.264/AVC JM reference code, which made it possible to add virtual reference frames to the reference buffer list according to our requirement. The virtual frame V'_{n-1} was projected from the previous frame F'_{n-1}. V'_{n-1} was added in the reference buffer list as a virtual reference frame. To span a reasonable range of bit rate, the QP was set from 20 to 34.

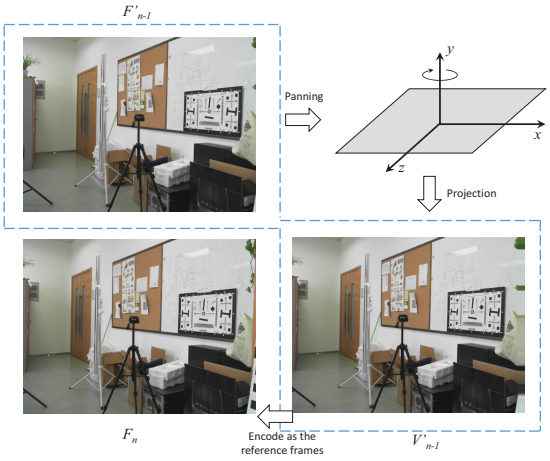

Fig. 10. An example of encoding process of the panning motion experiment

Fig.11 presents the PSNR comparison between the proposed scheme and standard H.264/AVC. The BD-Rate was -5.48%, while the BD-PSNR was 0.22 dB.

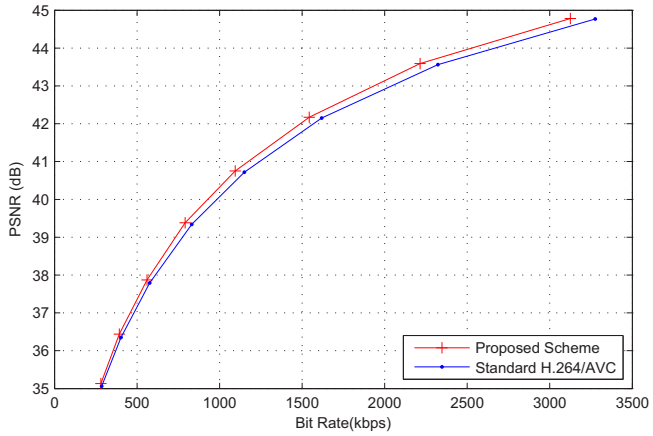

Fig. 11. PSNR versus bit rate for the proposed scheme and standard H.264/AVC in panning motion; the panning angle is around 1.33 degree per frame

In the translation motion experiment, we used forward movement (dolly) as examples. The resolution of texture images is VGA (640×480) @ 25 fps, while the resolution of depth images is QCIF (176×144). Moreover, there is a 5 cm distance between texture and depth cameras. Therefore, the depth images were projected, up-sampled and cropped in order to align the texture. In Fig.12, an example of dolly test is described. The virtual frame V'_{n-1} was projected from the previous frame F'_{n-1} using the depth information and the global motion information. The encoder settings were the same as the panning test.

In order to evaluate the impact of different translational distances, we set the distances as 2 cm and 5 cm per frame.

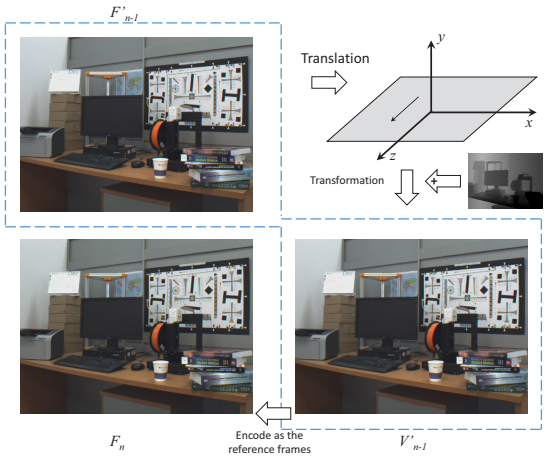

Fig. 12. An example of encoding process of the dolly motion experiment

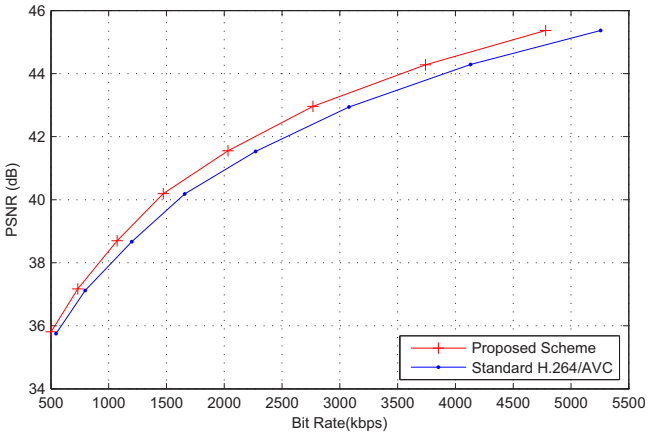

Fig. 13. PSNR versus bit rate for the proposed scheme and standard H.264/AVC in dolly forward motion; the distance is 2 cm per frame

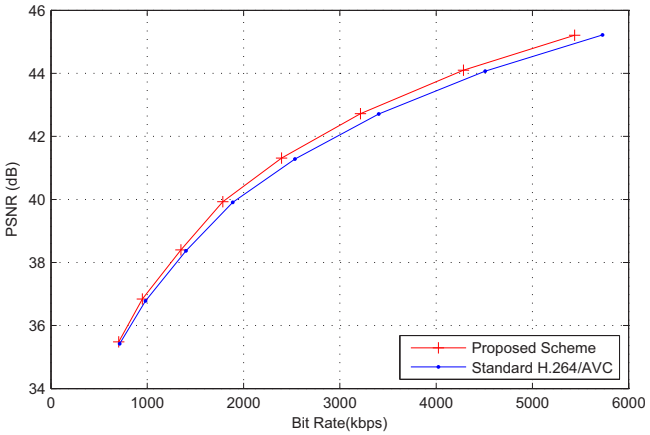

Fig. 14. PSNR versus bit rate for the proposed scheme and standard H.264/AVC in dolly forward motion; the distance is 5 cm per frame

The experimental results are showed in Fig.13 and Fig.14. In Fig.13, the BD-Rate is -11.87%, while the BD-PSNR is 0.49 dB. Although the 5 cm distance test presents lower coding gain than the 2 cm distance test, it is still better than standard H.264/AVC, with the BD-Rate being -5.7% and the BD-PSNR being 0.26 dB. From the results, we could conclude that when the translational distance is larger, the coding gain is smaller. The reason is that large motion means more pixels need to be interpolated. These pixels are not accurate enough. Therefore, the gain decreases with the increase of moved distance.

4. CONCLUSIONS

This paper has introduced a novel 3D video coding scheme using the motion information and depth information of the camcorder. Compared with the existing H.264/AVC standard, the proposed scheme is able to improve the coding performance up to 0.49 dB. It is noticed that the accuracy of the depth and motion information affects the performance of the proposed method, our future work is to improve the data precision. Meanwhile, we are going to investigate more advanced depth up-sampling methods.

5. REFERENCES

[1] Philipp Merkle, Aljoscha Smolic, Karsten Muller, and Thomas Wiegand, "Efficient prediction structures for multiview video coding," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, no. 11, pp. 1461–1473, 2007.

[2] Anthony Vetro, Thomas Wiegand, and Gary J Sullivan, "Overview of the stereo and multiview video coding extensions of the h. 264/mpeg-4 avc standard," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 626–642, 2011.

[3] H ITU-T RECOMMENDATION, "264 advanced video coding for generic audiovisual services," *ISO/IEC*, vol. 14496, 2003.

[4] M Hannuksela, Y Chen, T annd J.-R. Ohm Suzuki, and G. Sullivan (ed.), "Avc draft text 8," *JCT-3V document JCT3V-F1002*, vol. 16, 2013.

[5] Ying Chen, Miska M Hannuksela, Teruhiko Suzuki, and Shinobu Hattori, "Overview of the mvc+ d 3d video coding standard," *Journal of Visual Communication and Image Representation*, vol. 25, no. 4, pp. 679–688, 2014.

[6] Jimin Xiao, Tammam Tillo, Hui Yuan, and Yao Zhao, "Macroblock level bits allocation for depth maps in 3-d video coding," *Journal of Signal Processing Systems*, vol. 74, no. 1, pp. 127–135, 2014.

[7] Philipp Merkle, Aljoscha Smolic, Karsten Muller, and Thomas Wiegand, "Multi-view video plus depth representation and coding," in *Image Processing, 2007. ICIP 2007. IEEE International Conference on*. IEEE, 2007, vol. 1, pp. I–201.

[8] Christoph Fehn, "Depth-image-based rendering (dibr), compression, and transmission for a new approach on 3d-tv," in *Electronic Imaging 2004*. International Society for Optics and Photonics, 2004, pp. 93–104.

[9] Pei-Jun Lee and Xu-Xian Huang, "3d motion estimation algorithm in 3d video coding," in *System Science and Engineering (ICSSE), 2011 International Conference on*, June 2011, pp. 338–341.

[10] Kung-Yen Hsu and Shao-Yi Chien, "Hardware architecture design of frame rate up-conversion for high definition videos with global motion estimation and compensation," in *Signal Processing Systems (SiPS), 2011 IEEE Workshop on*, Oct 2011, pp. 90–95.

[11] Abdalbassir Abou-Elailah, Frédéric Dufaux, Joumana Farah, Marco Cagnazzo, and Béatrice Pesquet-Popescu, "Fusion of global and local motion estimation for distributed video coding," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 23, no. 1, pp. 158–172, 2013.

[12] Karsten Muller, Heiko Schwarz, Detlev Marpe, Christian Bartnik, Sebastian Bosse, Heribert Brust, Tobias Hinz, Haricharan Lakshman, Philipp Merkle, Franz Hunn Rhee, et al., "3d high-efficiency video coding for multi-view video and depth data," *Image Processing, IEEE Transactions on*, vol. 22, no. 9, pp. 3366–3378, 2013.

[13] Miska M Hannuksela, Dmytro Rusanovskyy, Wenyi Su, Lulu Chen, Ri Li, Payman Aflaki, Deyan Lan, Michal Joachimiak, Houqiang Li, and Moncef Gabbouj, "Multiview-video-plus-depth coding based on the advanced video coding standard," *Image Processing, IEEE Transactions on*, vol. 22, no. 9, pp. 3449–3458, 2013.

[14] Xiaoping Yun and Eric R Bachmann, "Design, implementation, and experimental results of a quaternion-based kalman filter for human body motion tracking," *Robotics, IEEE Transactions on*, vol. 22, no. 6, pp. 1216–1227, 2006.