IMPROVED DEPTH MAPS CODING EFFICIENCY OF 3D VIDEOS

Brian W. Micallef[1], Carl J. Debono[2], and Reuben A. Farrugia[3]

Department of Communications and Computer Engineering,
University of Malta,
Msida, Malta.
{[1] brian.micallef, [2] c.debono}@ieee.org, [3] reuben.farrugia@um.edu.mt.

ABSTRACT

Immersive 3D video services demand the transmission of the viewpoints' depth map together with the texture multi-view video to allow arbitrary reconstruction of intermediate viewpoints required for free-view navigation and 3D depth perception. The Multi-view Video Coding (MVC) standard is generally used to encode these auxiliary depth maps and since their estimation process is highly computational intensive, the coding time increases. This paper proposes a technique that exploits the multi-view geometry together with the depth map itself to calculate more accurate initial compensation vectors for the Macro-blocks' estimation. Starting from a more accurate position allows for a smaller search area, reducing the computations required during depth map MVC. Furthermore, the SKIP mode is extended to predict also the disparity vectors from the neighborhood encoded vectors, to omit some of them from transmission. Results demonstrate that these modifications provide an average computational reduction of up-to 87% with a bit-rate saving of about 8.3% while encoding an inter-view predicted viewpoint from a depth map multi-view video.

Index Terms—Depth multi-view video coding, Multi-view Video Coding, Multi-view video plus depth, efficient SKIP mode

1. INTRODUCTION

Recent work by the research and the standardization communities has shown that immersive Three-Dimensional Videos (3DVs) are more efficiently transmitted as texture Multi-View Videos (MVVs) together with their per-pixel depth maps, to form the Multi-view Video plus Depth (MVD) sequences [1]. These are essential for Depth Image Based Rendering [2], to allow the generation of arbitrary virtual viewpoints in-between fixed camera videos. This is used to create the second viewpoint required for depth perception in Three-Dimensional Television (3DTV) [3], and/or an accurately estimated intermediate viewpoint required for smooth navigation in Free-viewpoint Television (FTV) [4], thus it allows also a joint 3DTV/FTV service.

The additional transmission of the depth maps implies that further encoding and its associated computational overheads are required. The Multi-view Video Coding (MVC) standard was developed to efficiently transmit the texture MVVs [5] and later adopted to efficiently encode the new auxiliary depth map MVVs [6]. This standard exploits the fact that these videos contain also a high correlation between their viewpoints and that this redundancy can be removed by estimating a Macro-block (MB) from an already encoded viewpoint frame as well, to form disparity compensated MBs [5]. Thus, Disparity Estimation (DE) is performed together with classical Motion Estimation (ME), to estimate a MB from either a viewpoint or a temporal reference frame, respectively. The search for these efficient vectors is the most computational intensive part of the MVV encoder [7]. However, the depth map MVV consists of rich geometrical data that can provide useful information for fast estimation from each reference frame. Moreover, the H.264/MVC standard is based on the H.264/AVC and its SKIP mode still estimates and skips only the motion vectors. Thus, to increase the MVC efficiency, this can be extended to estimate also some of the disparity vectors.

This paper proposes a depth map MVC technique that exploits the latter two ideas. It proposes a technique that makes the motion and disparity estimation process faster by exploiting the multi-view geometry and the same depth map pixel values to be encoded. This information is used to calculate more accurate initial predictors for sub-MB motion or disparity estimation, such that they are more accurate and require a smaller search area. Moreover, the SKIP mode has been modified to use the neighborhood vectors to estimate also some disparity vectors. These result in a faster and more efficient depth map MVC such that an average speed-up gain of up-to 7.8 times and a bit-rate reduction of about 8.3% can be obtained while encoding an inter-view predicted depth map video. Such performance gain is highly desirable for MVD coding [7].

The rest of the paper is structured as follows; Section II describes the current MVC standard and how it can be used to encode the depth maps. Section III explains how the depth map data can be exploited during the macro-block estimation to provide a faster and a more efficient depth map MVC, while section IV presents the performance results obtained. Finally, Section V concludes this work.

2. DEPTH MAP MULTI-VIEW VIDEO CODING

The depth map video can be considered as a luminance video signal and can be efficiently estimated from another encoded depth map video [6], similar to the texture one [5]. Thus, likewise to the texture MVC, efficient depth map coding requires both motion and disparity estimation. Hence, the depth map reference frames are encoded using the H.264/AVC codec which is also extended to include the depth viewpoint reference frames to perform disparity estimation [5] for inter-view compensation. The estimation techniques perform a block-based matching method on these to identify an optimal compensation vector between two corresponding sub-MBs; one in the current frame to be encoded and one in each reference frame. This search is performed within all the search points enclosed in a defined search area. The Lagrangian Rate-Distortion (R-D) matching cost function is calculated for each search point and the compensation vector which minimizes this function is identified as the optimal vector. This vector minimizes the transmission bits with the least possible matching distortion error. When this optimal vector results from a temporal reference frame, it is called a Motion Vector (MV), while when it occurs from a viewpoint one, it is referred to as a Disparity Vector (DV). This exhaustive Full Search Estimation (FSE) makes this process the most computational intensive component of the MVC encoder. However, Fast Search Estimations (FASE), such as the Diamond Search [8], can be used in H.264/AVC to drastically reduce the search points and maintain near-optimal R-D performance. The initial and the central vector of the search area is the median of the neighborhood vectors as it contains high correlation with the optimal vector of the current MB. Following this, the resulting optimal vector from the estimation is transmitted as a residual vector from it. A viewpoint that is encoded without inter-view prediction and is used for initial reference is called the Base view. View 0 is generally encoded as the Base view as no other views are available for reference.

The median vector may be efficient for motion estimation, as the MBs are generally static or with homogeneous motion. However, it may not be accurate for disparity estimation [9-10], and may also result inefficient for dynamic regions during motion estimation [11-12]. Therefore, a large search area is required. Nevertheless, more accurate initial disparity [9-10] and motion [11-12] vector prediction can be respectively calculated for disparity and motion estimation in MVC by using the multi-view geometry, the depth maps, and the motion vectors which are already encoded for the Base view. This geometry can be used to predict the compensation vectors for the low-latency [11] and for the Hierarchical Bi-Prediction (HBP) [12] MVC prediction structures. Thus, more accurate initial vectors that indicate better substitutes and allow a reduction in the estimation's search area are obtained for fast depth map MVC, as in [9-12].

Generally, the median motion vector results as a good predictor to motion estimate the current MB and a large number of MBs are encoded as 16×16 modes with a zero residual motion vector. Thus, for efficiency, they are transmitted as a SKIP mode where no information is encoded for them and are compensated with the median motion vector. However, in MVC, the disparity vectors are also efficient to compensate the current MB, so the SKIP mode can be extended to estimate and encode also some disparity vectors, for more efficient depth map coding.

3. PROPOSED DEPTH MAP CODING

Usually, the depth map MVC, as its texture counterpart, has a good probability that it has its disparity vectors pointing to the corresponding multi-view regions in the viewpoint reference frame. These MBs can give the least distortion errors for compensation, as they are representing the same objects from another encoded viewpoint frame [9-10]. These corresponding macro-blocks can be identified using the multi-view geometry equation:

$$\zeta \boldsymbol{m} = \mathbf{P}\boldsymbol{M} \quad (1)$$

where $\boldsymbol{m}=(u, v, 1)^T$ are the homogeneous image coordinates of the current macro-block, $\boldsymbol{M}=(x, y, z, 1)^T$ are the homogeneous coordinates of the projected 3D point in space, $\mathbf{P}$ is the projection matrix, and ζ is the object's perpendicular depth [13]. This depth is the average value of the actual depth map pixel elements (pels) of the current sub-MB being encoded, since these re-project the sub-MB's pixels in 3D space in this camera coordinate system. A translation vector from the zero disparity vector to this identified warped position is then formed to obtain an initial vector which points to the potential matching inter-view sub-MB. This indicates an accurate position from where to start the search for the potential inter-view matching blocks for depth map compensation. Furthermore, using the appropriate viewpoint's fundamental matrix, a search area can be identified for the forward and backward viewpoint reference depth map, for bi-prediction disparity estimation.

Moreover, the initial motion vectors for temporal depth map estimation in MVC can also be calculated as the texture ones, since the depth correlation between the temporal frames is also dependent on the objects' motion. Thus, the optimal motion vectors already encoded for the Base view can be warped in the current viewpoint to obtain an initial vector for the currently being encoded sub-MB in both the low-latency [11] and the HBP [12] MVC schemes using equations (2) and (3). In this way, more accurate initial motion vectors are used to initiate the motion estimation from each temporal reference frame and this allows a substantial reduction in their search areas. Thus, the initial MV predictors in Figure 1 are obtained by:

$$V_2MV_F = DV_8 + V_0MV_{avg\,F} + DV_{16} \quad (2)$$

$$V_2MV_B = DV_8 + V_0MV_{avg\,B} + DV_0 \quad (3)$$

where DV_0, DV_8 and DV_{16} are translational vectors calculated using the multi-view geometry together with the

appropriate depth map data, and $MV_{avg\,F}$ and $MV_{avg\,B}$ are the averaged motion vectors for the forward and backward temporal reference frames in the Base view. The resulting optimal vectors should still be transmitted with respect to their original median ones since the depth map is not available during decoding, to re-calculate these geometric predictors.

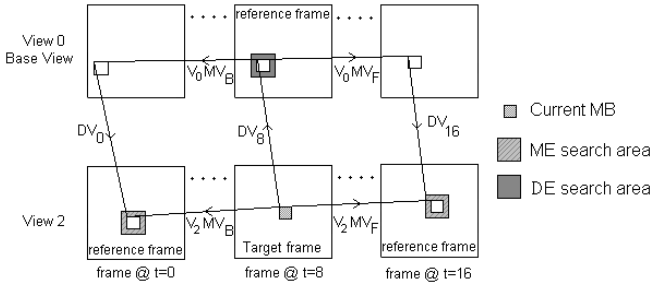

Fig. 1. Initial MVs for ME in both directions of the HBP MVC structure.

Since the H.264/MVC technique is an extension of the H.264/AVC, the SKIP mode estimates and omits only motion vectors. However, this can be extended to efficiently estimate also disparity vectors by taking the decision on which vector type to estimate from the neighborhood encoded vectors. This is because they still contain a high correlation as they are usually compensating the same objects. Thus, if the majority of these vectors are disparity compensated, the skipped MB's reference frame should change to a viewpoint one and a disparity vector should be estimated. Otherwise, if the majority is motion compensated, the skipped MB should be motion compensated. The estimated vector value of the SKIP mode can then be determined from the availability and the majority of the neighborhood vectors. If these are predicted from the same reference frame, then all the vectors exist and their median can be used. However, if only two of them are available, then only the nearest one should be selected as demonstrated in Figure 2, since a median cannot be determined. Finally, if only one vector is available, then this is used for compensation. Using this technique, some of the disparity vectors in the dynamic regions can be estimated from the neighborhood vectors, thus their transmission can be omitted. When hierarchal bi-prediction is used to encode the depth map video, the majority of the vector types from each compensation direction are used to determine the reference frames for bi-estimation of the SKIP mode.

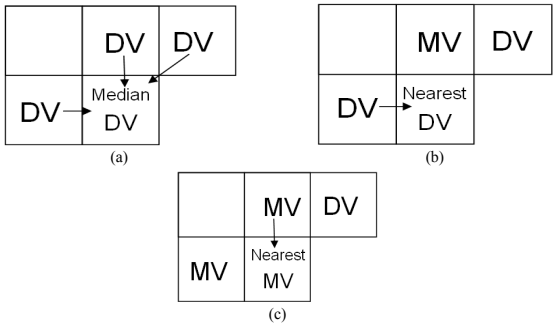

Fig. 2. The selection of the SKIP mode's compensation vector.

4. SIMULATION OVERVIEW

The proposed extensions for a faster motion and disparity estimation with the proposed SKIP mode, were implemented within the Joint Multi-view Video Coding model (JMVC ver. 6.0) [14], to determine their efficiency. This model was used to encode the depth map MVV of two standard MVD test sequences, which are known as the *Breakdancers* and the *Ballet* [15] sequences. Their texture MVV was captured using eight fixed cameras (1024×768, 15Hz) and their per-pixel depth map MVV was estimated using the techniques reviewed in [3]. The first three views from both MVD sequences were encoded. These were encoded for a low-delay scenario, where the low-latency MVC structure is required (Table I) and for a broadcast scenario, where the efficient Hierarchical Bi-Prediction MVC structure is required (Table II). View 0 was encoded as the Base view, view 2 was inter-view predicted from view 0, while view 1 was inter-view bi-predicted from views 0 and 2. Both exhaustive FSE and the diamond search [8] as FASE were used to determine the optimal compensation vectors. The simulations were carried out on a PC with an Intel® Core™ i7 CPU @ 3.2GHz, with 6GB of RAM and Microsoft Windows® 7 Ultimate x64.

The reduction in the required encoding computational cost is measured as a speed-up gain obtained while encoding the inter-view predicted depth map viewpoints when using the MVC encoder with the proposed extensions, compared to the original one. Then, the coding efficiency of the proposed estimation techniques was determined as the bit-rate saving obtained. Finally, the MVC decoder was modified to re-determine the SKIP mode's compensation vector and it was used to decode the formed bit-streams.

TABLE I. THE LOW-LATENCY MVC SIMULATION PARAMETERS.

Multi-view HIGH Profile
Temporal low-latency coding structure (I–P–P–P)
I – B – P inter-view coding structure
CAVLC as main the entropy encoder
Original search area of ± 32 pels
Proposed search area of ± 10 pels
Fixed Quantization Parameters (QPs) of 28, 32, 36, and 40

TABLE II. THE HBP MVC SIMULATION PARAMETERS.

Temporal HBP coding structure with a GOP of 16		
I – B – P inter-view prediction structure		
CABAC as the main entropy encoder		
Original search area of ± 32 pels		
Proposed search area of ± 10 pels		
Base Quantization Parameters (QPs) of 28, 32, 36, and 40 [16]		
Delta QP Values:	TemporalDeltaLayer0Quant	0
	TemporalDeltaLayer1Quant	3
	TemporalDeltaLayer2Quant	4
	TemporalDeltaLayer3Quant	5
	TemporalDeltaLayer4Quant	6
	TemporalDeltaLayer5Quant	7

5. RESULTS AND ANALYSIS

Tables III to VI present the simulation results obtained by the modified JMVC model with respect to the original one, when the low-latency and the Hierarchal Bi-Prediction

TABLE III. R-D RESULTS FOR THE LOW-LATENCY MVC ENCODERS WITH FSE ON THE *BREAKDANCERS'* DEPTH MAP VIEWPOINTS.

		Viewpoint 2			Viewpoint 1		
	QP	PSNR(dB)	Time(hr)	Bit-rate	PSNR(dB)	Time(hr)	Bit-rate
Original	28	46.52	11.08	709.224	46.68	20.80	632.315
	32	43.38	11.07	441.615	43.68	20.81	400.484
	36	40.37	11.04	260.157	40.85	20.73	250.394
	40	37.40	11.02	157.909	38.10	20.66	156.429
Proposed	28	46.32	1.36	665.787	46.41	2.55	585.074
	32	43.25	1.36	415.921	43.45	2.54	370.414
	36	40.30	1.35	240.737	40.67	2.53	228.110
	40	37.46	1.35	139.587	37.97	2.51	139.609

TABLE IV. R-D RESULTS FOR THE HBP MVC ENCODERS WITH FSE ON THE *BREAKDANCERS'* DEPTH MAP VIEWPOINTS.

		Viewpoint 2			Viewpoint 1		
	QP	PSNR(dB)	Time(hr)	Bit-rate	PSNR(dB)	Time(hr)	Bit-rate
Original	28	43.02	25.70	306.848	43.18	27.94	284.026
	32	40.45	25.60	185.624	40.71	27.88	170.981
	36	38.06	25.31	112.849	38.37	27.69	104.115
	40	35.51	25.31	69.320	36.02	27.89	64.4856
Proposed	28	42.97	3.34	285.541	42.99	3.56	260.906
	32	40.42	3.33	175.026	40.57	3.63	153.245
	36	37.92	3.31	104.144	38.15	3.71	96.079
	40	35.49	3.35	61.453	35.82	3.60	57.871

TABLE V. R-D RESULTS FOR THE LOW-LATENCY MVC ENCODERS WITH FASE ON THE *BREAKDANCERS'* DEPTH MAP VIEWPOINTS.

		Viewpoint 2			Viewpoint 1		
	QP	PSNR(dB)	Time(hr)	Bit-rate	PSNR(dB)	Time(hr)	Bit-rate
Original	28	46.45	0.82	723.622	46.57	1.34	644.863
	32	43.33	0.73	444.779	43.64	1.24	408.461
	36	40.35	0.65	261.734	40.81	1.13	251.569
	40	37.36	0.58	157.512	38.09	1.01	157.897
Proposed	28	46.409	0.30	682.064	46.492	0.52	600.874
	32	43.266	0.30	417.743	43.496	0.50	372.396
	36	40.373	0.29	244.831	40.788	0.49	227.393
	40	37.536	0.27	140.167	38.123	0.48	139.185

TABLE VI. R-D RESULTS FOR THE HBP MVC ENCODERS WITH FASE ON THE *BREAKDANCERS'* DEPTH MAP VIEWPOINTS.

		Viewpoint 2			Viewpoint 1		
	QP	PSNR(dB)	Time(hr)	Bit-rate	PSNR(dB)	Time(hr)	Bit-rate
Original	28	42.94	1.72	309.474	43.09	1.77	284.592
	32	40.47	1.51	187.400	40.65	1.59	170.437
	36	38.03	1.30	112.104	38.33	1.40	102.693
	40	35.49	1.15	68.0424	36.03	1.24	63.597
Proposed	28	42.97	0.59	291.199	43.06	0.61	260.090
	32	40.27	0.65	172.118	40.46	0.60	149.784
	36	37.84	0.67	101.738	38.08	0.65	91.8716
	40	35.39	0.64	59.618	35.71	0.69	56.4432

MVC structures are used to encode the first three views of the *Breakdancers*' depth map MVV. The comparisons are in terms of the Peak-Signal-to-Noise Ratio (PSNR) in dB, the required bit-rate in kbps, and the overall encoding duration in hours, obtained after encoding the inter-view predicted viewpoints. The Base view encoding results are not included in the tables since the original MVC encoder was used for both the original and proposed method, thus there is no gain or loss in the tested parameters. Furthermore, Figure 3 illustrates the R-D performance obtained by the different MVC encoders on the depth map MVV of the *Ballet* sequence. Both the exhaustive full search estimation and the diamond search were used to determine the optimal compensation vectors. Finally, the R-D gain obtained is also presented as a potential gain in video quality in terms of the Bjøntegaard Delta [17] (BD)-PSNR.

Tables III to VI demonstrate that with the proposed estimation extensions, faster and more efficient depth map MVC can be achieved. In fact, an average speed-up gain of 8.2 with a bit-rate reduction of 8.2% for full search and a speed-up gain of 2.3 with a bit-rate reduction of 8.3% for fast search estimation were achieved, with the low-latency MVC of the inter-view predicted viewpoints. It also provides an average speed-up gain of 7.7 and 2.3, while providing also a bit-rate reduction of 8.5% and of 8.8%, for full search and fast search estimations respectively, with the HBP MVC. These become equivalent to the average gains in video quality of 0.38dB as BD-PSNR for low-latency MVC and of 0.3dB as BD-PSNR for HBP MVC.

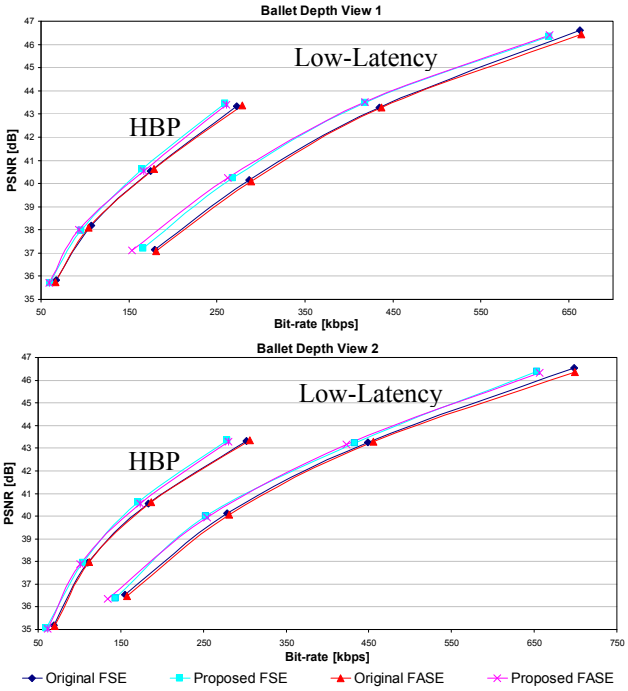

Fig. 3. R-D performance of the encoders on the *Ballet*'s depth map MVV.

Figure 3 illustrates that when the results are averaged over both the full and the fast search estimations, a reduction of 7.6% in the encoded bit-rate or an equivalent

gain in video quality of 0.46dB as BD-PSNR resulted for the low-latency inter-view predicted viewpoints. On the other hand, for the HBP MVC, a bit-rate reduction of 8.5% or a quality gain of 0.38dB as BD-PSNR was achieved. These also provide an average speed-up gain of 7.6 for the full search and of 2.2 for the fast search estimation during low-latency MVC, and a gain of 8.1 for the full search and of 2.2 for the fast search estimation during HBP MVC, similar to those presented in the Tables above.

Using the proposed geometric predictors on their own provides a slight decrease in the R-D performance of the depth map MVC, as in [9-12], since a reduction in the search area will lose the encoding of some optimal large compensation vectors in areas with high dynamics or large depth variations. However, the new SKIP mode is very efficient and it compensates for this loss while providing also a good bit-rate reduction, without any significant increase in the MVC codec's complexity. The efficiency of the proposed SKIP mode is obtained because it manages to predict some of the disparity vectors from the neighborhood, especially for the dynamic areas, thus, by excluding them from transmission it provides a decrease in the encoded bit-rates with a small loss in the encoded quality.

6. CONCLUSION

A method that decreases the computational cost of motion and disparity estimation was proposed for depth map MVC. It exploits the geometric information provided by the pixel elements to better encode the depth map MVV, by obtaining more accurate geometric compensation vectors to initialize and reduce the respective search areas. Then, the SKIP mode was modified to predict also some disparity vectors, to increase the multi-view video coding efficiency. Simulation results showed that these estimation extensions can provide an average video quality gain of about 0.4dB BD-PSNR while reducing the computations of the exhaustive full search estimation method by about 87% and those of the fast search estimation technique by about 55%.

7. ACKNOWLEDGMENTS

The research work disclosed in this publication is partially funded by the Strategic Educational Pathways Scholarship Scheme (Malta). The scholarship is part-financed by the European Union – European Social Fund. We would like to thank the Interactive Media Group of Microsoft Research for providing us with the *Breakdancers* and the *Ballet* multi-view video-plus-depth sequences.

8. REFERENCES

[1] ISO/IEC, and ITU-T, "*Multi-view Video plus Depth (MVD) format for advanced 3D video systems*," Doc. JVT-W100, April 2007.

[2] P. Kauff, N. Atzpadin, C. Fehn, M. Müller, O. Schreer, A. Smolic, and R. Tanger, "Depth map creation and image based rendering for advanced 3DTV services providing interoperability and scalability," *Signal Processing: Image Communication; Special issue on three-dimensional video and television*, vol. 22, no. 2, pp. 217-234, Feb. 2007.

[3] C. Zitnick, S. Kang, M. Uyttendaele, S. Winderm, and R. Szeliski, "High-quality video view interpolation using a layered representation," *ACM SIGGRAPH and ACM transaction on Graphics*, pp. 600-608, Aug. 2004.

[4] Y. Mori, N. Fukushima, T. Fujii, and M. Tanimoto, "View generation with 3D warping using depth information for FTV," in *Proc. of 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video*, pp. 229-232, May 2008.

[5] P. Merkle, K. Müller, A. Smolic, and T. Wiegand, "Efficient compression of multi-view video exploiting inter-view dependencies based on H.264/MPEG-AVC," in *Proc. of IEEE International Conference on Multimedia and Expo*, pp. 1717-1720, July 2006.

[6] P. Merkle, A. Smolic, K. Müller, and T. Wiegand, "Efficient compression of multi-view depth data based on MVC," in *Proc. of 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video*, May 2007.

[7] ISO/IEC, and ITU-T, "*Survey of algorithms used for Multi-view Video Coding (MVC)*," Doc. N6909, Jan. 2005.

[8] S. Zhu, and K. –K. Ma, "A new diamond search algorithm for fast block-matching motion estimation," *IEEE Transactions on Image Processing*, vol. 9, no. 2, pp. 287-290, Feb. 2000.

[9] B. W. Micallef, C. J. Debono, and R. A. Farrugia, "Exploiting depth information for fast multi-view video coding," in *Proc. of Picture Coding Symposium*, pp. 38-41, Dec. 2010.

[10] B. W. Micallef, C. J. Debono, and R. A. Farrugia, "Fast disparity estimation for multi-view video plus depth coding," in *Proc. of Visual Communications and Image Processing*, Nov. 2011.

[11] B. W. Micallef, C. J. Debono, and R. A. Farrugia, "Exploiting depth information for fast motion and disparity estimation in multi-view video coding," in *Proc. of 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video,* May 2011.

[12] B. W. Micallef, C. J. Debono, and R. A. Farrugia, "Fast multi-view video plus depth coding with Hierarchical Bi-Prediction," in *Proc. of International Symposium on Communications, Control and Signal Processing*, May 2012.

[13] R. Hartley, and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge Press, 2003, UK, pp. 279-309.

[14] ISO/IEC, and ITU-T, "*Joint Multi-view Video Coding model (JMVC 6.0)*," JVT-AE207, Sept. 2009.

[15] *MSR Multi-view Video-plus-Depth Sequences* [Online]. Available: http://www.research.microsoft.com/en-us/um/people/sbkang/3dvideodownload.

[16] ISO/IEC, and ITU-T, "*Common test conditions for Multiview Video Coding*," Doc. JVT-U211, Oct. 2006.

[17] G. Bjøntegaard, "*Calculation of average PSNR differences between RD-curves*," Doc. VCEG-M33, April 2001.