J Sign Process Syst (2014) 74:127–135
DOI 10.1007/s11265-013-0743-5

Macroblock Level Bits Allocation for Depth Maps in 3-D Video Coding

Jimin Xiao · Tammam Tillo · Hui Yuan · Yao Zhao

Received: 14 January 2013 / Revised: 21 March 2013 / Accepted: 25 March 2013 / Published online: 12 May 2013

Abstract For 3-D videos, one commonly used representation method is texture videos plus depth maps for several selected viewpoints, whereas the other viewpoints are synthesized based on the available texture videos and depth maps with the depth-image-based rendering (DIBR) technique. As both the quality of the texture videos and depth maps will affect the quality of the synthesized views, bits allocation for the depth maps become indispensable. The existing bits allocation approaches are either inaccurate or requiring pre-encoding and analyzing in temporal dimension, making them unsuitable for the real-time applications. Motivated by the fact that different regions of the depth maps have different impacts on the synthesized image quality, a real-time macroblock level bits allocation approach is proposed, where different macroblocks of the depth maps are encoded with different quantization parameters and coding modes. As the bits allocation granularity is fine, the R-D performance of the proposed approach outperforms other bits allocation approaches significantly, while no additional pre-encoding delay is caused. Specifically, it can save more than 10 % overall bit rate comparing with Morvan's full search approach, while maintaining the same synthesized view quality.

Keywords 3-D video coding · Bits allocation · Macroblock level · Real-time · R-D optimization

J. Xiao
Department of Electrical Engineering & Electronic,
University of Liverpool,
Liverpool, UK
e-mail: jimin.xiao@liverpool.ac.uk

J. Xiao · T. Tillo (✉)
Department of Electrical & Electronic Engineering,
Xi'an Jiaotong-Liverpool University,
Suzhou, China
e-mail: tammam.tillo@xjtlu.edu.cn

H. Yuan
School of Information Science and Engineering,
Shandong University,
Jinan, China

Y. Zhao
Institute of Information Science,
Beijing Jiaotong University,
Beijing, China

1 Introduction

3-D video is a set of motion pictures that provide the illusion of depth perception for the video content. The first step of introducing 3-D video could be regarded as the stereo video system, which is attracting lots of attention due to its success in the real-world applications, such as in 3-D cinema. Stereoscopic video allows the viewers to view the scene from one fixed viewpoint, where the next step of the 3-D video would be multiple viewpoint video, or even arbitrary viewpoint video. For the multiview system, one of the major challenges is how to represent the tremendous amount of information with limited bit rate resource. Simulcast coding scheme or more advanced multiview video coding (MVC) [1, 2] are possible solutions. In MVC, besides the temporal correlation, the interview correlation is also exploited to further enhance the compression performance. Both simulcast and MVC, however, require a bit rate that is proportional to the number of views [3], which means for the applications with large number of views, the generated bit rate is enormous. One commonly used technique to reduce the

3-D bit rate is depth-image-based rendering (DIBR) [4], such as 3-D warping, which uses texture videos plus depth maps to synthesize other viewpoint videos, it is a promising solution, because it can represent the large viewpoint scenes with much less data than that of only using the simulcast or MVC.

The texture-plus-depth format, the 3-D representation scheme used in this paper, has one texture video and depth map at each captured viewpoint, where the depth map is a 2-D grey image recording the distance of objects from capturing devices. The encoded texture and depth video sequences can enable the decoder to synthesize any intermediate virtual views via the DIBR technique. This format is currently the chosen format for 3-D scene representation in the free viewpoint video (FVV) working group in MPEG.

Due to the popularity of the texture-plus-depth format, many research works on how to compress the depth maps in an efficient way have been carried out. These depth coding methods include using depth-adaptive reconstruction filter [5] to replace erroneous pixels, introducing allowable depth distortion to lower the bit rate [6], and using linear modeling functions to represent the depth maps [7]. In [8], motivated by the fact that the depth maps are not perceived by the viewers but only supplement data for view synthesis, instead of using the distortion of the depth map itself, the authors proposed to evaluate and use the distortion of the synthesized view in the coding mode selection step, where rate-distortion optimization is used; later in [9], the synthesized view distortion was modeled at pixel-level and in a more accurate way, which eventually leaded to better overall rate-distortion performance than that of [8].

In the texture-plus-depth format, the depth maps are always used together with the associated texture videos, and both the texture videos and the depth maps quality will affect the synthesized view quality. So, prior to the compression technique of the depth map itself, one more fundamental problem needs to be addressed is how to allocate bit rate for the depth maps. A heuristic approach with fixed ratio (5:1) bits allocation between texture videos and depth maps was used in [10]. Later, Morvan [11] proposed a full search algorithm to find the optimal quantization parameter (QP) pair for texture videos and depth maps. However, this algorithm assumes that a real view exists at the synthesized viewpoint, and the distortion of the synthesized virtual view is evaluated using mean squared error (MSE) between the synthesized virtual view and its corresponding real view in order to do the optimization. In [12], Liu proposed a distortion model to estimate the distortion of the synthesized views without the need of comparing the synthesized view with its corresponding real view. A fast bits allocation algorithm was proposed in [13] to reduce the complexity, where the allocation performance is comparable with that of [12]. In recent work [14], a region-based view synthesis distortion estimation approach and a general R-D property estimation model is proposed, the reported results in [14] show that it can provide better R-D performance than [12] with lower computational cost. There are, however, some major issues in all the above mentioned bits allocation approaches, one is that [11–13] need to pre-encode and analyze a certain number of frames of the encoded video sequence, which makes them not suitable for the real-time 3-D streaming applications; another is that the granularity of bits allocation in [11–14] is frame level, which means that the same QP is used for the whole frame. In the 3-D scenes, the importance of different regions in the same depth map is usually different, and the same level of depth map distortion in different regions may lead to different level of rendering view distortion. To address these two issues, a Real-Time Macroblock (MB) level Bits Allocation (RT-MBA) approach for depth maps is proposed in this paper, where different amount of bit rate is allocated to different regions of the depth maps. The allocation is based on the texture video QP and other texture video characteristics and using the synthesized view R-D optimization. The preliminary version of this work was presented at [15].

The rest of the paper is organized as follows. The proposed approach is presented in Section 2. In Section 3 experimental results validating the proposed approach are given. Finally, conclusions are drawn in Section 4.

2 Proposed Real-time Macroblock level Bits Allocation Approach

2.1 Distortion Model for Synthesized View

In order to optimally allocate the bits for the depth maps, the distortion model for the synthesized view will be required. Given that the proposed RT-MBA approach aims to do bits allocation at MB level, then the distortion modeling unit should not be larger than one MB. The synthesized view distortion will be estimated without comparing the virtual view with its corresponding real view, as in [9, 12, 13], because in practical applications, the existence of the real view is not guaranteed. Based on the above requirements, we select the synthesized view distortion model presented in [9], where the distortion is modeled at pixel level, and it mimics the view synthesizing process with sub-pixel interpolation.

The distortion of the synthesized view will be the sum of squared distance (SSD) between two versions of the synthesized view; the first version, denoted by $V_{x',y'}$, is synthesized from the original texture videos and the depth maps; whereas the other is generated from the compressed

version of the decoded texture videos and their associated depth maps, denoted by $\tilde{V}_{x',y'}$. The SSD in this case is:

$$SSD_V = \sum_{(x',y')} \left| V_{x',y'} - \tilde{V}_{x',y'} \right|^2 = \sum_{(x,y)} \left| f_w(C, D_{x,y}) - f_w\left(\tilde{C}, \tilde{D}_{x,y}\right) \right|^2 \quad (1)$$

where C and D indicate the original color video and the depth map, respectively; whereas $\tilde{C}$ and $\tilde{D}$ denote the decoded color video and depth map, respectively; (x', y') is warped pixel position for the synthesized view V corresponding to (x, y) in C and D by the predefined warping function, f_w, and (x, y) is the pixel inside the current non-synthesized macroblock B. As in [9], the Eq. 1 can be further simplified as $SSD_V = E_t + E_d$, with $E_t = \sum_{(x,y)} \left| f_w(C, D_{x,y}) - f_w\left(\tilde{C}, D_{x,y}\right) \right|^2$, denoting the distortion caused by the compression of the texture videos, and $E_d = \sum_{(x,y)} \left| f_w\left(\tilde{C}, D_{x,y}\right) - f_w\left(\tilde{C}, \tilde{D}_{x,y}\right) \right|^2$, denoting the distortion caused by the compression of the depth maps.

In the 1-D parallel camera setting configuration, the 3-D configuration used in this paper, the synthesized view distortion that caused by the depth maps can be further approximated as [9]:

$$E_d \approx \sum_{(x,y)} \left| \tilde{C}_{x,y} - \tilde{C}_{x-\Delta p(x,y),y} \right|^2 \quad (2)$$

where Δp denotes the translational horizontal rendering position error. It is already proven that it is proportional to depth map error:

$$\Delta p(x, y) = \alpha \cdot \left(D_{x,y} - \tilde{D}_{x,y} \right) \quad (3)$$

where α is a proportional coefficient determined by the following equation:

$$\alpha = \frac{f \cdot L}{255} \left(\frac{1}{Z_{\text{near}}} - \frac{1}{Z_{\text{far}}} \right) \quad (4)$$

with f being the focal length, L being the baseline between the current and the rendered view, Z_{near} and Z_{far} being the values of the nearest and farthest depth of the scene, respectively. Finally, the value of E_d can be approximated as [9]:

$$E_d \approx \sum_{(x,y)} \frac{|\Delta p(x, y)|}{2} \times \left(\left| \tilde{C}_{x,y} - \tilde{C}_{x-1,y} \right| + \left| \tilde{C}_{x,y} - \tilde{C}_{x+1,y} \right| \right) \quad (5)$$

2.2 Optimal Bits Allocation for the Depth Maps

In the proposed RT-MBA approach, we optimally allocate bit rate for the depth maps at MB level. To find the optimal bits allocation, the following constrained minimization is formulated.

$$\begin{cases} \min SSD_V \\ \text{subject to } R_t + R_d = R_B \end{cases} \quad (6)$$

where R_t and R_d denote the amount of bits used to encode the MB's texture and depth map, respectively; R_B is the total number of bits dedicated for both texture and the depth map for the current MB B. For the term SSD_V, it is a convex function of bit rate, because E_t is a convex function, whereas for the term E_d, it is observed that its relationship with bit rate is also convex. Two examples of E_d versus bit rate are demonstrated in Fig. 1. In the figures, the achievable E_d versus bit rate performance is marked as red curve, and it is observed that this curve, which is formed by the convex hull of all the rate distortion points, is convex.

Therefore, this problem can be solved by means of the standard Lagrangian approach by minimizing the cost function:

$$J = SSD_V + \lambda(R_t + R_d) \quad (7)$$

where λ is the Lagrangian multiplier. Imposing $\Delta J = 0$ we get:

$$\frac{\partial J}{\partial R_t} = \frac{\partial E_t}{\partial R_t} + \lambda = 0 \quad (8)$$

$$\frac{\partial J}{\partial R_d} = \frac{\partial E_d}{\partial R_d} + \lambda = 0 \quad (9)$$

By combining (8) and (9) we can conclude that in order to minimize J the following condition must be satisfied:

$$\frac{\partial E_d}{\partial R_d} = \frac{\partial E_t}{\partial R_t} = -\lambda \quad (10)$$

This means that the slope of the distortion of the texture versus its bit rate should be equal to the slope of the depth distortion versus its bit rate. This slope is λ, the Lagrangian multiplier, and for H.264/AVC it is given by [16]

$$\lambda = 0.85 \cdot 2^{(QP_t - 12)/3} \quad (11)$$

with QP_t being the quantization parameter of the texture videos. Combining Formula (10) and (11), we can get

$$\frac{\partial E_d}{\partial R_d} = -\lambda = -0.85 \cdot 2^{(QP_t - 12)/3} \quad (12)$$

This means that for a fixed QP_t for the texture video, we need to find a combination of QP and coding mode for the depth map that satisfies the Eq. 12. Therefore, finding the

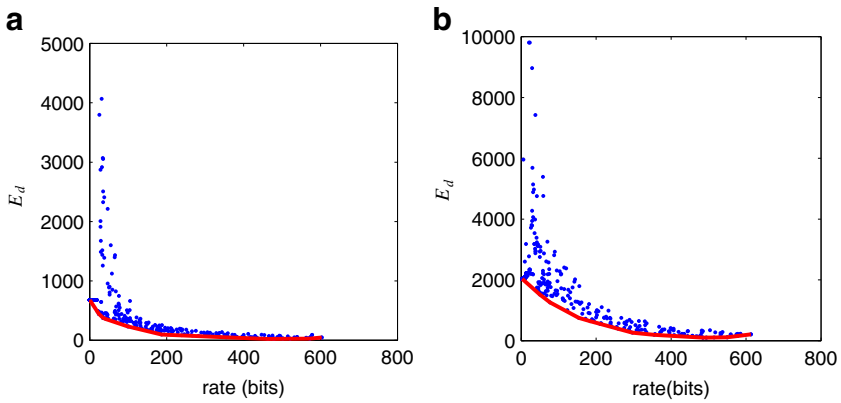


Figure 1 Two examples of E_d versus bit rate, different QPs and coding modes are used to generate different pairs of bit rate and E_d. The *red curves* represent the best achievable performance curve, formed by the convex hull of all the rate distortion points.

optimal allocation of the bit rate for the depth map requires the QP value and the coding mode that minimize the cost function J'

$$J' = E_d + 0.85 \cdot 2^{(QP_t - 12)/3} \cdot R_d \tag{13}$$

Based on this, the optimal encoding option for the depth maps, O^* could be denoted as follows:

$$O^* = \arg\min_{o \in \Gamma} J' \tag{14}$$

where the possible candidates of QP (QP_d) and encoding mode (m) pairs are $\Gamma = \{(QP_d, m) \mid QP_t - T \leq QP_d \leq QP_t + T, m \in MODE\}$, with T being the searching range of the QP for the depth map ($T = 14$ is used this article), and $MODE$ being all the available prediction coding modes for the current type of slice. Typically, decreasing the texture QP also requires decreasing the depth map QP, this is reasonable as high quality texture videos also require high quality depth maps to achieve high overall R-D performance, whereas for low quality texture videos, providing high quality depth maps may not synthesize to high quality virtual views. For this reason, QP_t is set to be at the middle point of the searching range. Thus, the full search algorithm becomes trying all the QP and coding mode pairs, and select the pair that leads to minimal J', and using this QP and coding mode could lead to optimal bits allocation for the depth map.

2.3 Fast Algorithm for QP and Coding Mode Selection

The number of possible QP and coding mode combinations is large, which means the computational complexity of the full search algorithm is high. Nevertheless, it is observed that the optimal coding modes for adjacent QPs are highly correlated. In other words, the selected coding modes for QP_d and $QP_d \pm 1$ are with high probability the same. Inspired by this observation, the number of tested QP and coding mode pairs will be reduced. To do this, firstly, we down-sample the possible QP range, which means that the new QP candidates for depth maps become $\mathcal{Z}_{QP} = \{QP_i, i = 1, 2, 3...\}$. The down-sampling process could be uniform or nonuniform, in this article, for simplicity, uniform down-sampling is used so as to have five QPs to test, which means $\mathcal{Z}_{QP} = \{QP_t - T, QP_t - T/2, QP_t, QP_t + T/2, QP_t + T\}$. After down-sampling, for each QP_i the R-D optimization is carried out. In order to demonstrate the simplified algorithm, let us assume that the optimal coding mode is M_i for QP_i. Secondly, each M_i will be compared with the following one, *i.e.*, M_{i+1}, for $i = 1, 2, 3, 4$. For example, if $M_i = M_{i+1}$, for all the $\{QP_d | QP_i < QP_d < QP_{i+1}\}$, the only available coding mode is M_i; whereas if $M_i \neq M_{i+1}$, the possible coding modes could be either M_i or M_{i+1}. Finally, for the refined set of QP and coding mode pairs, the one that leads to the minimal J' will be used to encode the current MB's depth map. The detailed procedure of this fast bits allocation algorithm is depicted in Algorithm 1. It is also found that, the significant computational complexity reduction is not obtained by sacrificing the overall bits allocation performance, and this will be demonstrated by the experimental results in Section 3.

3 Experimental Results

In the experiments, we use the video sequences: BookArrival, Kendo, Pantomine, Newspaper, Poznan Street and Ballet, and the detailed test setting is listed in Table 1. The proposed algorithm is implemented based on H.264/AVC reference software JM 14.0 [17], and View Synthesis Reference Software [18] is used for view synthesis. The performance of the proposed approach is obtained by using its fast algorithm described in Section 2.3 if not otherwise noted.

In the first set of experiments, we compared the R-D performance of the proposed bits allocation algorithm with fixed ratio 5 : 1 bits allocation method and full search algorithm [11]. The results of bits allocation schemes in

Algorithm 1 Fast bit rate allocation algorithm

for $QP_i \in Z_{QP}$ **do**

encode current MB with QP_i and R-D function (13);
record the minimal cost $RD_{cost}[QP_i] \Leftarrow J'$;
record the optimal coding mode $mode[QP_i] \Leftarrow O^*$;

end for

/* j is the index of sub searching range after down-sampling */

for $j = 1$ to 4**do**

for $qp\_i = QP_j$ to QP_{j+1}**do**

encode macroblock with qp_i, $mode[QP_j]$, $mode[QP_{j+1}]$;
$RD_{cost}\_1 \Leftarrow$ R-D cost using coding mode $mode[QP_j]$;
$RD_{cost}\_2 \Leftarrow$ R-D cost using coding mode $mode[QP_{j+1}]$;
$RD_{cost}[qp\_i] \Leftarrow min(RD_{cost}\_1, RD_{cost}\_2)$;
if $RD_{cost}\_1 < RD_{cost}\_2$ **then**

$mode[qp\_i] \Leftarrow mode[j]$;

else

$mode[qp\_i] \Leftarrow mode[j+1]$;

end if

end for

end for

$cost \Leftarrow \infty$; /* select the pair leads to minimal cost */

for $qp\_i = QP_t - T$ to $QP_t + T$**do**

if $RD_{cost}[qp\_i] \leq cost$ **then**

$cost \Leftarrow RD_{cost}[qp\_i]$;
$QP^* \Leftarrow qp\_i$;
$mode^* \Leftarrow mode[qp\_i]$;

end if

end for

encode current MB with QP^* and $mode^*$;

[12–14] are not reported, because as reported in [13, 14], these methods are less performing than [11] in terms of the R-D performance. The R-D curves of different bits allocation algorithms are reported in Fig. 2. Comparing the proposed RT-MBA approach with Morvan's full search algorithm, it is noted that to get the same synthesized view quality, 10 % to 20 % overall bit rate is saved for the video sequences BookArrival, Kendo, Pantomine and Newspaper and PoznanStreet, whereas more than 40 % overall bit rate can be saved for the Ballet sequence. This is because for the Ballet sequence, the quality of the depth map has a big impact on the synthesized view, and it requires more bit rate than the texture videos in order to get the best R-D performance. This is the reason for which Ballet has more gain than the other sequences, for which the depth maps only account for about 20 % of the texture bit rate.

In Table 2, we compare the R-D performance of the proposed RT-MBA approach with Byung's algorithm [9]. Three separable methods have been proposed in Byung's article [9], it is important to note that the second and third methods can also be jointly applied with the proposed RT-MBA approach, so we compare the RT-MBA approach with the first method of [9], which is the main contribution of Byung's article, without using the second and third methods. In Byung's Method, the coding mode of each MB's depth map is optimally selected based on the new synthesized view distortion model, while the QP value for the whole depth map is fixed, which is pre-assigned. Thus, Byung's algorithm does not have the functionality of optimal bits allocation for the depth maps.

To have a fair comparison between the RT-MBA approach and Byung's algorithm, the same QP is used for the texture videos, while for the depth maps, firstly we encode the depth maps with the proposed RT-MBA approach, then for depth coding of Byung's algorithm, we use the QP parameter, which generates more bits than the proposed RT-MBA approach. This procedure ensures fair comparison by favoring Byung's algorithm. Nevertheless, as reported in Table 2, in spite of the fact that RT-MBA approach has 6.88 % less bits for the depth maps, the average synthesized view PSNR is 0.40 dB higher than Byung's

Table 1 The experimental environments for the simulations.

Sequences	BookArrival	Newspaper	Kendo	Pantomine	PoznanStreet	Ballet
Resolution	1024 × 768	1024 × 768	1024 × 768	1280 × 960	1920 × 1088	1024 × 768
GOP size	8	15	15	15	12	8
Frame	1 – 60	1 – 60	1 – 60	1 – 60	151 – 210	1 – 60
Intra period	8	15	15	15	12	8
View No. (I-P)	10 – 8	2 – 4	1 – 3	39 – 41	3 – 5	0 – 2
Frame rate	16.7	30	30	30	25	15

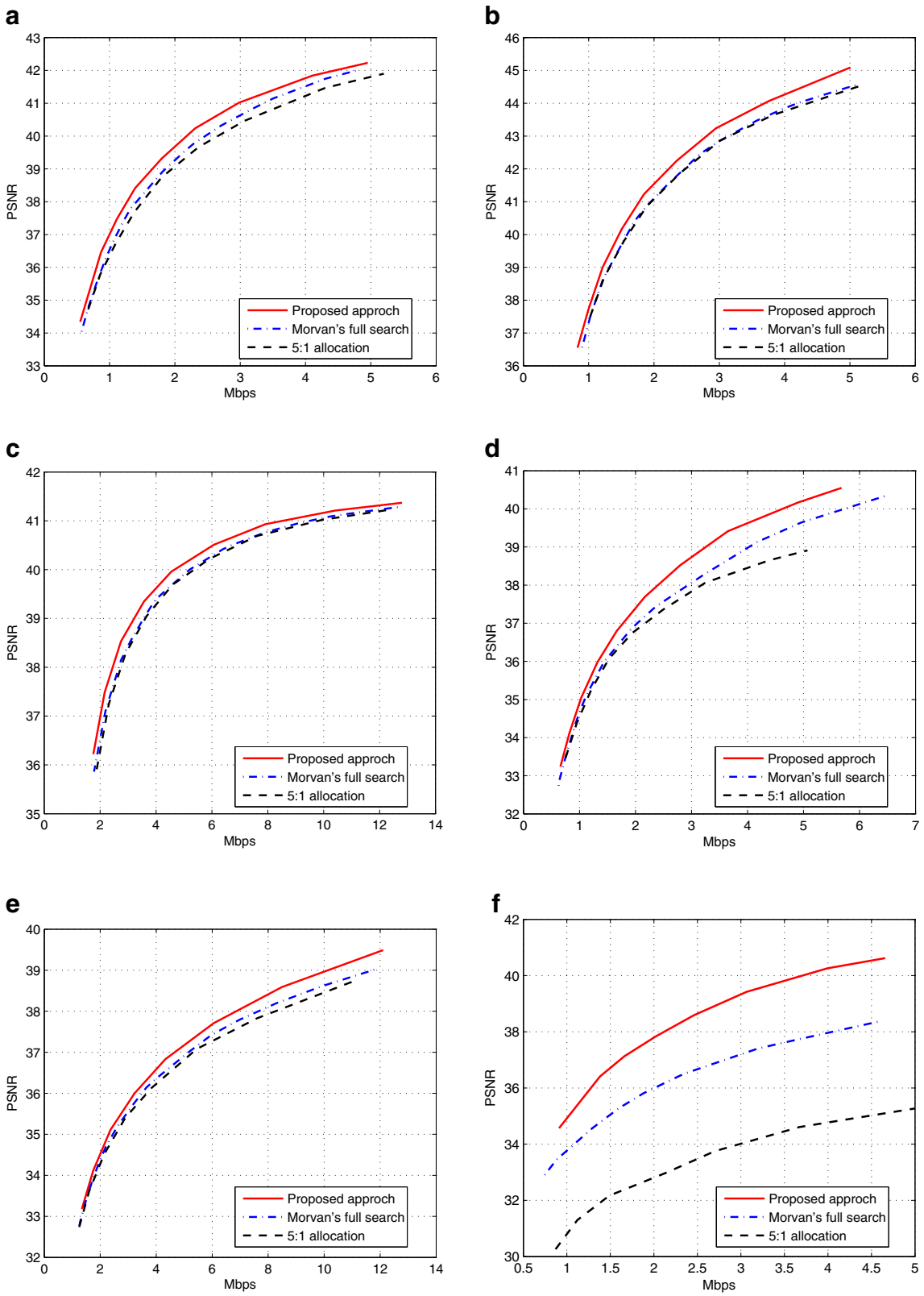


Figure 2 R-D performance of different bits allocation algorithms; (**a**) BookArrival, (**b**) Kendo, (**c**) Pantomine, (**d**) Newspaper, (**e**) PoznanStreet, (**f**) Ballet.

Table 2 Performance comparison between the proposed RT-MBA approach and Byung's approach (Method-1) [9].

Video sequence	Texture QP	Bit rate of RT-MBA (Kbps)	Bit rate of Byung's (Kbps)	RT-MBA PSNR (dB)	Byung PSNR (dB)
BookArrival	24	560.96(−8.78 %)	614.96	41.84(0.37)	41.47
	28	367.71(−1.50 %)	373.33	40.24(0.57)	39.81
	32	242.15(−6.67 %)	259.48	38.41(0.30)	38.11
Kendo	24	764.18(−5.65 %)	809.99	45.09(0.40)	44.69
	28	458.10(−1.05 %)	463.00	43.24(0.13)	43.11
	32	283.03(−3.48 %)	293.25	41.23(0.06)	41.17
Pantomine	26	2121.50(−8.33 %)	2314.36	41.21(0.02)	41.19
	30	1057.25(−10.14 %)	1176.60	40.51(0.03)	40.48
	34	498.87(−12.96 %)	573.18	39.35(0.06)	39.29
Newspaper	24	1257.58(−3.47 %)	1302.81	40.17(0.25)	39.82
	28	709.16(−7.27 %)	764.81	38.52(0.29)	38.23
	32	400.06(−6.45 %)	427.65	36.79(0.13)	36.66
PoznanStreet	26	3002.80(−8.26 %)	3273.48	39.49(0.36)	39.13
	30	1560.74(−13.61 %)	1806.64	37.71(0.18)	37.53
	34	806.50(−0.64 %)	811.72	36.00(0.15)	35.85
Ballet	24	2233.47(−3.72 %)	2319.85	40.26(1.77)	38.49
	28	1542.16(−7.06 %)	1659.33	38.59(1.14)	37.45
	32	1102.65(−2.04 %)	1125.72	37.13(1.09)	36.04
Average	–	1053.8(−6.88 %)	1131.7	39.76(0.40)	39.36

method. This results serve to demonstrate the importance of using different QPs for the different regions of the depth maps.

The fast algorithm described in Section 2.3 could decrease the computational complexity significantly by reducing the number of coding mode options. Nevertheless, the complexity reduction is not obtained by sacrificing the overall bits allocation performance too much. Figure 3 reports the synthesized view PSNR versus the overall bit rate, including the texture videos and the depth maps, for the full version RT-MBA approach algorithm and its fast algorithm. The two curves are nearly overlapped for both the BookArrival and Kendo sequences. Whereas the for the Ballet sequence, the full version curve is about 0.2 higher than that of the fast algorithm. This is because for the Ballet sequence, the depth map accounts for more than 50 % of the total bit rate. For the video sequences Pantomine, Newspaper and PoznanStreet, the two curves are almost overlapped, the same as the BookArrival and Kendo sequences, so they are not reported here.

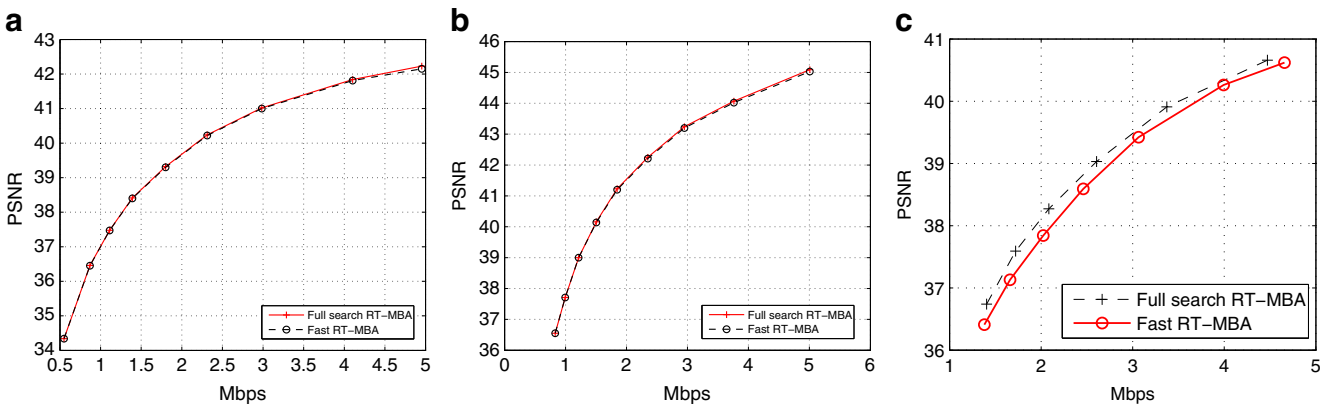


Figure 3 Synthesized view PSNR versus overall bit rate for the RT-MBA approach and its fast algorithm; (**a**) BookArrival, (**b**) Kendo, (**c**) Ballet.

4 Conclusions

In this paper, a bits allocation scheme for the depth maps in the texture-plus-depth format has been proposed. The proposed scheme allocates the bit rate resource in real-time fashion. Moreover, the resource allocation granularity is at MB level, which makes the proposed allocation scheme quite accurate. To reduce the computational complexity of the proposed full search algorithm, a fast implementation algorithm has been introduced, which can reduce the complexity significantly while maintaining the R-D performance. Experimental results have demonstrated that R-D performance of the proposed scheme is higher than other 3-D bits allocation algorithm.

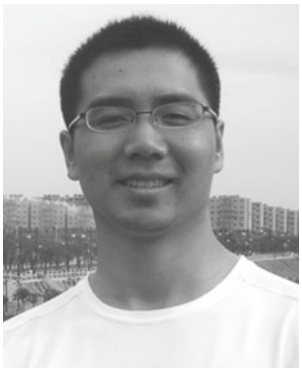
Acknowledgments This work was supported by the National Natural Science Foundation of China (NO. 60972085, NO. 61210006, NO. 61201211), Xi'an Jiaotong-Liverpool University Research Development Fund (RDF-11-01-11), Ph.D. Programs Foundation of Ministry of Education of China (No. 20120131120032), and the Excellent Youth Scientist Award Foundation of Shandong Province (No. BS2012DX021).

References

1. Merkle, P., Smolic, A., Muller, K., Wiegand, T. (2007). Efficient prediction structures for multiview video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, *17*(11), 1461–1473.
2. Vetro, A., Wiegand, T., Sullivan, G. (2011). Overview of the stereo and multiview video coding extensions of the h.264/mpeg-4 avc standard. *Proceedings of the IEEE*, *99*(4), 626–642.
3. Müller, K., Merkle, P., Wiegand, T. (2011). 3-d video representation using depth maps. *Proceedings of the IEEE*, *99*(4), 643–656.
4. Merkle, P., Smolic, A., Muller, K., Wiegand, T. (2007). Multi-view video plus depth representation and coding. In *IEEE international conference on image processing, 2007. ICIP 2007* (Vol. 1, pp. I–201–I–204).
5. Oh, K.-J., Vetro, A., Ho, Y.-S. (2011). Depth coding using a boundary reconstruction filter for 3-d video systems. *IEEE Transactions on Circuits and Systems for Video Technology*, *21*(3), 350–359.
6. Zhao, Y., Zhu, C., Chen, Z., Yu, L. (2011). Depth no-synthesis-error model for view synthesis in 3-d video. *IEEE Transactions on Image Processing*, *20*(8), 2221–2228.
7. Morvan, Y., Farin, D., de With, P. (2007). Depth-image compression based on an r-d optimized quadtree decomposition for the transmission of multiview images. In *IEEE international conference on image processing, 2007. ICIP 2007* (Vol. 5, pp. V–105–V–108).
8. Kim, W.S., Ortega, A., Lai, P.L., Tian, D., Gomila, C. (2010). Depth map coding with distortion estimation of rendered view. In *Proceedings of SPIE visual information processing and communication*.
9. Oh, B.T., Lee, J., sik Park, D. (2011). Depth map coding based on synthesized view distortion function. *IEEE Journal of Selected Topics in Signal Processing*, *5*(7), 1344–1352.
10. Fehn, C. (2004). Depth-image-based rendering (DIBR), compression and transmissio for a new approach on 3-D-TV. In *Proc. SPIE, stereoscopic image process. Render* (Vol. 5291, pp. 93–104).
11. Morvan, Y., Farin, D., de With, P.H.N. (2007). Joint depth/texture bit-allocation for multi-view video compression. In *Picture coding symposium (PCS)* (pp. 265–268).
12. Liu, Y., Huang, Q., Ma, S., Zhao, D., Gao, W. (2009). Joint video/depth rate allocation for 3-D video coding based on view synthesis distortion model. *Signal Processing: Image Communication*, *24*(8), 666–681.
13. Yuan, H., Chang, Y., Huo, J., Yang, F., Lu, Z. (2011). Model-based joint bit allocation between texture videos and depth maps for 3-d video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, *21*(4), 485–497.
14. Wang, Q., Ji, X., Dai, Q., Zhang, N. (2011). Free viewpoint video coding with rate-distortion analysis. *IEEE Transactions on Circuits and Systems for Video Technology*, *22*(6), 875–889.
15. Xiao, J., Tillo, T., Yuan, H. (2012). Real-time macroblock level bits allocation for depth maps in 3-d video coding. In W. Lin, D. Xu, A. Ho, J. Wu, Y. He, J. Cai, M. Kankanhalli, M.-T. Sun (Eds.), *Advances in multimedia information processing C PCM 2012, ser. Lecture notes in computer science* (Vol. 7674, pp. 232–240). Berlin: Springer. doi:10.1007/978-3-642-34778-8_21.
16. Wiegand, T., Sullivan, G., Bjontegaard, G., Luthra, A. (2003). Overview of the H.264/AVC video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, *13*(7), 560–576.
17. HHI Fraunhofer Institute. *H.264/AVC reference software*. Available online: http://iphome.hhi.de/suehring/tml/download/.
18. MPEG-3-DV view synthesis reference software. Available online: http://wg11.sc29.org/svn/repos/MPEG-4/test/trunk/3D/view_synthesis.

Jimin Xiao was born in Suzhou, China. He received the BS and MEng degrees in telecommunication engineering from Nanjing University of Posts and Telecommunications, Nanjing, China, in 2004 and 2007, respectively. Since 2009, he has been pursuing the Ph.D degree at the University of Liverpool, Liverpool, U.K. From 2007 to 2009, he was software engineer at Motorola (China) Electronics, Ltd., and later as a system engineer at Realsil (Realtek) Semiconductor Corp. Since Mar. 2013, he has been a visiting scholar in CeMNet lab of Nanyang Technological University, Singapore. His research interests are in the areas of video streaming, image and video compression, and 3-D video coding.

Tammam Tillo (M05-SM12) was born in Damascus, Syria. He received the Engineer Diploma in Electronic Engineering from Damascus University, Damascus, Syria, in 1994, and the Ph.D. in Electronics and Communication Engineering from Politecnico di Torino, Torino, Italy, in 2005. From 1999 to 2002 he was with Souccar for Electronic Industries, Damascus, Syria. In 2004 he was visiting researcher at the EPFL (Lausanne, Switzerland), and from 2005 to 2008, he worked as a Post-Doctoral researcher at the Image Processing Lab of Politecnico di Torino, and for few months he was Invited Research Professor at the Digital Media Lab, SungKyunKwan University, Suwon, S. Korea. In 2008 he joined Xian Jiaotong-Liverpool University (XJTLU), Suzhou, China. Currently, he is the Head of Electrical and Electronic Engineering Department and Acting Head of Department, Department of Computer Science and Software Engineering at XJTLU university. His research interests are in the areas of robust image and video transmission, image and video compression, and hyperspectral image compression.



Hui Yuan received the B.E. and Ph.D. degree in telecommunication engineering from Xidian University, Xian, China, in 2006 and 2011, respectively. He is currently a Lecturer with the School of Information Science and Engineering, Shandong University (SDU), Jinan, China. Now, he is also in City University of Hong Kong for post-doctor research. His current research interests include video coding, multimedia communication, etc.

Yao Zhao (M'06-SM'12) received the B.S degree from Fuzhou University in 1989 and the M.E degree from the Southeast University in 1992, both from the Radio Engineering Department, and the PhD degree from the Institute of Information Science, Beijing Jiaotong University (BJTU) in 1996. He became an associate professor at BJTU in 1998 and became a professor in 2001. From 2001 to 2002, he worked as a senior research fellow in the Information and Communication Theory Group, Faculty of Information Technology and Systems, Delft University of Technology, Netherlands. He is now the director of the Institute of Information Science, Beijing Jiaotong University. His research interests include image ideo coding, fractals, digital watermarking, and content based image retrieval. Now he is leading several national research projects from 973 Program, 863 Program, the National Science Foundation of China. He was the recipient of the National Outstanding Young Investigator Award of China in 2010.