DISTRIBUTED MULTI-HYPOTHESIS CODING OF DEPTH MAPS USING TEXTURE MOTION INFORMATION AND OPTICAL FLOW

Matteo Salmistraro[◊] *Marco Zamarin*[◊] *Lars Lau Rakêt*[⋆] *Søren Forchhammer*[◊]

[◊]DTU Fotonik, Technical University of Denmark, Ørsteds Plads,
2800 Kgs. Lyngby, Denmark. Emails: {matsl, mzam, sofo}@fotonik.dtu.dk
[⋆]Department of Computer Science, University of Copenhagen, Universitetsparken 1,
2100 Copenhagen, Denmark. Email: larslau@diku.dk

ABSTRACT

Distributed Video Coding (DVC) is a video coding paradigm allowing a shift of complexity from the encoder to the decoder. Depth maps are images enabling the calculation of the distance of an object from the camera, which can be used in multiview coding in order to generate virtual views, but also in single view coding for motion detection or image segmentation. In this work, we address the problem of depth map video DVC encoding in a single-view scenario. We exploit the motion of the corresponding texture video which is highly correlated with the depth maps. In order to extract the motion information, a block-based and an optical flow-based methods are employed. Finally we fuse the proposed Side Informations using a multi-hypothesis DVC decoder, which allows us to exploit the strengths of all the proposed methods at the same time.

Index Terms— Distributed Source Coding, Depth Map Coding, Wyner-Ziv Coding, Optical Flow, Distributed Video Coding.

1. INTRODUCTION

In this work we address the coding of depth maps, using DVC [1, 2] as basis of our coding architecture.

Depth maps are particular images enabling the calculation of the distance of an object from the camera. A video representation format that is gaining popularity is the so-called "video-plus-depth", where in addition to texture data (the luminance and chrominance information of the scene), per-pixel depth information is also provided [3, 4]. Depth data allows fast generation of virtual views using the so-called Depth-Image-Based-Rendering (DIBR) algorithms [4], which makes the video-plus-depth format suitable for 3DTV and free viewpoint system implementations [5]. Moreover, it can be used for a number of purposes that can be of interest in modern video surveillance scenarios such as scene matting, activity detection and object tracking [3].

DVC is a video coding paradigm that allows shifting the complexity from the encoder side to the decoder side due to the fact that Motion Estimation (ME)—which heavily contributes to the computational complexity in state-of-the-art video codecs—can be performed at the decoder. Typical DVC scenarios feature strict power consumption constraints at the transmitter side, requiring low-complexity encoders, while the requirements are less stringent at the decoder. A multi-camera video surveillance scenario is a good example of a system with such requirements [2]. In a typical DVC architecture [1] inter-coded frames (i.e. frames coded by means of motion estimation and compensation) are substituted by the so-called Wyner-Ziv (WZ) frames. WZ frames are encoded in a different manner: parity check data are calculated and transmitted. An Intra-coded frame is referred to as Key Frame (KF) and is encoded and transmitted as in traditional video coding. At the decoder side KFs are used to estimate WZ frames by means of ME. The estimated frame, called Side Information (SI), can be corrected using parity bits from the encoder. The SI generation algorithm is therefore of crucial importance as the quality of the estimated frames directly affects the amount of additional parity bits required, and consequently the Rate-Distortion (RD) performance of the system. The core part of the proposed decoder is the Transform Domain WZ (TDWZ) codec [6]. At the encoder the WZ frame is DCT transformed and quantized. Each DCT coefficient is organized in bitplanes, and for each bitplane a LDPCA [7] encoder calculates the parity bits. At the decoder the SI is generated using an interpolation-based technique, for example Overlapped Block Motion Compensation (OBMC) [6]. A subset of the parity bits are sent to the decoder. The decoder tries to correct the errors present in the corresponding bitplane of the SI using the parity bits. If the decoding is not successful new bits are requested. Another key element of the decoder is the noise modelling, which is important in order to provide the LDPCA decoder with the likelihood of the value of each bit. The errors present in the SI are modelled as Laplacian distributed errors. In order to calculate the distribution, an estimation of the residual is needed. The residual is the difference between the SI and the original frame, which can not be directly calculated in practice. For more information on

DVC coding the reader is referred to [1, 2, 6].

Since texture and depth represent different aspects of the same 3D scene, the two components show a high correlation [8]. In a video-plus-depth DVC scenario such correlation can be exploited to improve the overall coding efficiency e.g. by refining the depth SI generation using texture motion information.

In this paper Transform Domain WZ coding of depth maps is addressed in a mono-view video-plus-depth scenario. This scenario is interesting when addressed with DVC because two dependent streams can be independently encoded but dependently decoded. This approach can be generalized to a multi-camera scenario, where a depth camera and a texture camera are used together, making inter-camera cooperation difficult or not feasible. Texture data are supposed to be available at the decoder and are used to improve the WZ decoding of depth data. Three different SIs are generated and fused using a multi-hypothesis approach [9]. The first SI is generated by applying block-based texture motion vectors to the depth component; the second one is obtained by applying the texture optical flow to the depth component; finally the third one is generated by means of motion estimation from depth data only. The three SIs present different characteristics and provide accurate estimation of the to-be-decoded depth frame in different regions.

1.1. Related Works

The use of texture motion information for depth compression purposes has been explored in conventional predictive coding in [10] and more recently in [11]. The same concepts can be exploited in a DVC decoder for accurate SI generation, as done in [12] in which multiple decoded texture frames are used. In our work we suppose that the decoder has access to the corresponding texture frame of the to-be-decoded depth frame, while in [12] only the texture frames corresponding to the depth KFs are used. Moreover, we investigate optical-flow-based methods, while [12] investigate only block-based methods. The multi-hypothesis decoder employed is the same as in [9] where OBMC was used with block-based extrapolation and optical flow-based interpolation in order to improve a texture-based DVC decoder. We use the same approach in order to effectively fuse three different SIs.

A preliminary study of the aforementioned problem has been performed in [13] but only the block-based method was presented and no fusion technique was proposed.

Optical flow-based SI generation has already been used for example in [9]. In this case the flows were used to interpolate an unknown texture frame given the previous and successive texture frames. In our framework we use the flow to extract the motion information from texture frames.

The remainder of this paper is organized as follows: Section 2 describes the proposed SI generation algorithms and the relative SI fusion method. In Section 3 experimental results are discussed. Finally, Section 4 summarizes the presented work.

2. SI GENERATION AND FUSION

In this section we describe the two proposed SI generation algorithms for depth maps, exploiting texture motion information. We also analyse the employed fusion procedure. In addition a third SI based on OBMC [6] on depth video is included in the fusion procedure. This decoder is used as basis for evaluating the performance of the two texture-based SIs and the performance of the fusion procedure. It has to be noted however, that OBMC has not been devised for depth maps and it has not been modified in this work.

2.1. Texture-based SI generation algorithms

The main idea behind the proposed methods is that the motion of the texture is highly correlated with the one encountered in the depth data. For the to-be-decoded depth frame X at instant t, assume that the depth maps at instants $t-1$, and $t+1$ (D_{t-1} and D_{t+1}, respectively) are known. We can use the motion information of the texture to warp D_{t-1} and D_{t+1} towards X obtaining the SI Y. In order to perform the aforementioned procedure the texture frames at instants $t-1$, t, and $t+1$ (C_{t-1}, C_t, C_{t+1}, respectively) are available at the decoder. The Motion Vectors (MVs) are calculated from C_t to C_{t-1} and from C_t to C_{t+1}. The MVs are used in turn to motion compensate D_{t-1} and D_{t+1}, obtaining two depth SIs Y_1 and Y_2, respectively. The final SI, Y, is calculated as the arithmetic average of Y_1 and Y_2. The residual, R_Y, is calculated as the absolute difference between Y_1 and Y_2. The argument behind this simple choice is that if a region in X presents simple motion, it will be well predicted. Hence Y_1 and Y_2 will agree in the particular area, leading to low residual estimation. If on the contrary the two estimated frames disagree, the residual will be higher.

The methods used to calculate the motion from the texture data is of central importance to the SI quality. We have selected two different ME approaches.

2.2. Block-Based Side Information Generation

We consider the so-called "Adaptive Rood Pattern Search" (ARPS) ME algorithm proposed in [14]. This approach may not provide the lowest MSE (Mean-Squared-Error) between the motion compensated texture frame and the original one, however, it is able to capture the motion between the frames in a robust way, leading to fewer artefacts in the warped (depth) frame. ARPS has been proposed as a way to reduce the complexity of the ME process in state-of-the-art predictive coding, but thanks to the adaptive nature of the pattern and the refinement step, it produces superior results compared with

full search in the given setup. This Block-Based SI generation is referred as BB.

2.3. Optical Flow Side Information Generation

As an alternative to BB, we consider an Optical Flow (OF) [15] SI generation. As opposed to BB, the OF based ME is global, in the sense that individual motion vectors are estimated for every pixel. Given a set of texture frames C_t and $C_{t'}$, $(t' = t+1, t-1)$, in pixel domain, we want to estimate the dense flow field v such that the optical flow constraint

$$D(\boldsymbol{x}, v) \triangleq C_{t'}(\boldsymbol{x} + v(\boldsymbol{x})) - C_t(\boldsymbol{x}), \tag{1}$$

where $\boldsymbol{x}$ denotes a point in the image, is close to zero.

The optical flow constraint (1) will not be sufficient for motion estimation, and in order to make the problem well posed, one has to penalize irregular behavior. Here we focus on the TV-L^1 energy, where data fidelity between two frames is measured by the L^1-norm of the optical flow constraint, and the regularization term penalizes the total variation of the estimated motion:

$$E(v) = \int \lambda \|D(\boldsymbol{x}, v)\| + \|\mathscr{D} v(\boldsymbol{x})\| \, \mathrm{d}\boldsymbol{x}. \tag{2}$$

The total variation of a vector valued function is not uniquely defined, and several definitions have been used for this problem [16, 17, 18]. Here we use the definition of [19], since this method does not suffer from the channel smearing (i.e. independent optimization of the two channels of the motion vectors, the x- and y-components) of other definitions.

The final estimate of the motion v is recovered from iteratively minimizing a linearized version of (2) using the duality based splitting of [16]. The minimization is performed in a coarse to fine pyramid. We use 65 pyramid levels wit a scaling factor of 1.05, and Gaussian blurring of C_t and $C_{t'}$ with standard deviation 0.5, and on each level we perform 90 warps, with 1 outer and 10 inner iterations [16]. Furthermore we remove outliers by performing a median filtering of the flow for each warp. The parameter λ was set to 480. Compared to optical flow based interpolation [9] this value may seem high, however for the given test setup with higher temporal and spatial resolution, as well as the direct knowledge of the texture state, this higher weight on data fidelity is adequate. For more details on the implementation we refer to [18]. OF may lead to the non pixel location problem, in which a target position in D_{t-1} and D_{t+1} does not have integer coordinates. In this case bicubic interpolation is used.

2.4. Side Information Fusion

In order to exploit all the presented SIs (BB, OF, OBMC) a robust fusion technique is needed. In [9] it has been demonstrated that a multi-hypothesis decoder can be used to effectively combine block-based and pixel-based motion estimation techniques. In our work, we use the three SI decoders approach (referred as 3SI) as a way to fuse the three SIs. The multi-hypothesis decoder allows implementing a rate-based optimization strategy by using a number of parallel LDPCA decoders. Each LDPCA decoder is fed with a different weighted combination of the conditional probabilities for a given bitplane, and the syndromes coming from the encoder. Each bitplane contains the co-located bits of a given DCT coefficient. The decoded sequence of the first converging decoder is chosen as solution, and the corresponding weights used to combine the SIs are also used in the reconstruction process to improve the PSNR of the decoded frame. This method, thanks to the multi-decoder structure, shows robust gain, good performance and is therefore employed in this work as the fusion technique. However, the 3SI approach will increase the complexity of LDPC decoding up to 6 times.

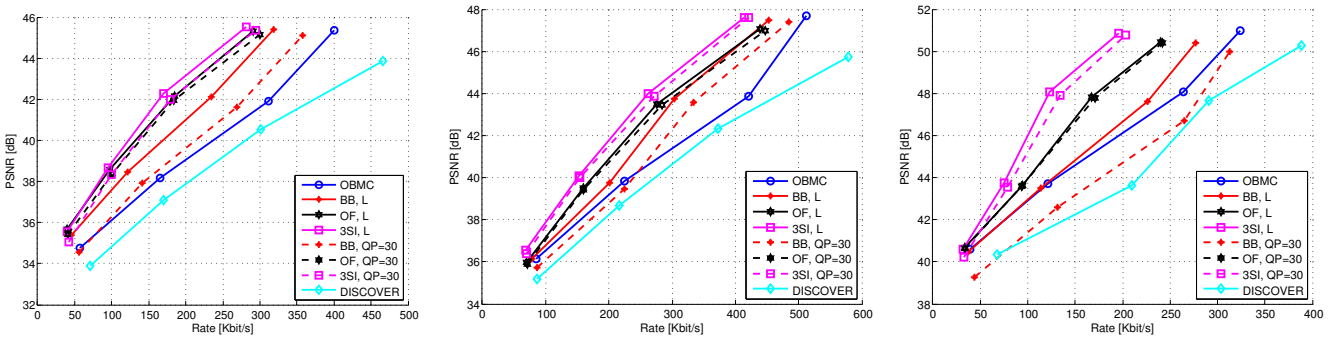
3. EXPERIMENTAL RESULTS

The system has been tested on the sequences "Breakdancers" and "Ballet" from Microsoft Research [20], and "Dancer" from Nokia Research [21]. We used the central view of the three sequences, at 15 fps downsampled to CIF resolution. The quantization matrices Qi = 1, 4, 7, 8 of the DISCOVER [22] project are employed. The KFs are H.264/AVC Intra encoded using QP = 40, 37, 31, 29 and are matched with the quantization matrices. We have tested the first 100 frames of each sequence and reported the results for Group-Of-Pictures (GOP) 2, 4, 8. The performance of the WZ frames has been evaluated and compared with the single SI OBMC decoder. In Tables 1-3 we list the Bjøntegaard differences [23] between the single SI OBMC decoder and the 3SI decoder. The results for lossless coded textures are listed as "QP = L", while the results using compressed textures, are listed with the QP used for compression. Texture compression has been performed with a standard H.264/AVC Intra coder[1]. In Figs. 1a-1c the RD curves for GOP2 are reported. We have also reported the performance of the single SI system and the performance of DISCOVER. Only the performance for WZ frames is reported. It has to be noted that the parameters of the 3SI decoder are the same for all the sequences and the quality level of the textures.

In Section 2, the SI generation for GOP2 has been outlined. In the cases of GOP4 and GOP8 a hierarchical coding structure [6] is used. First the SI for the central WZ frame is generated using C_{t-k}, C_{t+k}, D_{t-k}, and D_{t+k}, where k corresponds to half of the GOP size. The decoded WZ frame splits the GOP in two smaller GOPs in which the procedure can be iterated until all the WZ frames have been decoded.

From the results presented, it can be seen that the OF outperforms all the other single SI methods, showing also high robustness against texture quantization, while the BB

[1] JM 18.1 Reference Software, available at `iphome.hhi.de/suehring/tml`



(a) Ballet (b) Breakdancers (c) Dancer

Fig. 1: RD curves, WZ frames only, GOP2.

method suffers at lower qualities of the texture frames. The single SI OBMC-based decoder outperforms DISCOVER [24] codec in all the studied conditions and for all the investigated sequences. The 3SI is able to correctly fuse the three SIs, performing, on average, better or as well as the best available SI for the particular RD point. The improvements between the single SI OBMC decoder and the 3SI decoder ranges from 1.50 to 4.95 dB and from 21.24% to 49.06% bit-rate Bjøntegaard savings. Interestingly, the improvements for GOP8, are higher in the case of compressed textures for the Ballet and Breakdancers sequences (Table 3). A justification can be found in the non-linear low-pass filtering nature of the quantization, leading to more robust results, which in case of complex motion can be of benefit.

Sequence	**QP**	**ΔPSNR** [dB]	**ΔRate** [%]
Ballet	L	2.98	−46.46
	20	2.85	−44.99
	30	2.40	−39.79
Breakdancers	L	2.12	−34.02
	20	2.07	−33.28
	30	1.87	−31.16
Dancer	L	2.05	−42.90
	20	2.04	−40.41
	30	1.82	−36.24

Table 1: Bjøntegaard Distances between OBMC and the proposed methods, GOP2.

Sequence	**QP**	**ΔPSNR** [dB]	**ΔRate** [%]
Ballet	L	3.16	−44.71
	20	3.06	−43.32
	30	2.57	−38.11
Breakdancers	L	1.71	−23.87
	20	1.68	−23.52
	30	1.50	−21.24
Dancer	L	2.47	−42.54
	20	2.38	−42.03
	30	2.00	−38.81

Table 2: Bjøntegaard Distances between OBMC and the proposed methods, GOP4.

4. CONCLUSION

In this work we addressed the problem of DVC-based depth-map coding. We devised algorithms to produce higher quality SIs, employing the texture frames. We used two methods in order to extract the motion information from the texture frames: a block-based method and an optical flow-based one. The optical flow achieved better performance and superior robustness to quantization of the textures compared with the block-based system. The multi-hypothesis decoder proved to be an effective and robust way to fuse the three generated SIs outperforming the best single SI available. The improvements between the single SI OBMC decoder and the multi-hypothesis decoder ranges from 1.50 to 4.95 dB and from 21.24% to 49.06% Bjøntegaard bit-rate savings.

Sequence	**QP**	**ΔPSNR** [dB]	**ΔRate** [%]
Ballet	L	3.03	−42.80
	20	3.46	−46.62
	30	2.98	−41.53
Breakdancers	L	1.80	−23.95
	20	1.95	−25.55
	30	1.76	−23.37
Dancer	L	4.95	−49.06
	20	4.74	−47.61
	30	4.43	−45.04

Table 3: Bjøntegaard Distances between OBMC and the proposed methods, GOP8.

5. REFERENCES

[1] B. Girod, A.M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 71 –83, January 2005.

[2] F. Pereira, "Distributed video coding: basics, main solutions and trends," in *Proc. of IEEE ICME*, Piscataway, NJ, USA, 2009, ICME'09, pp. 1592–1595, IEEE Press.

[3] K. Müller, P. Merkle, and T. Wiegand, "3-D video representation using depth maps," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 643 –656, April 2011.

[4] P. Kauff, N. Atzpadin, C. Fehn, M. Müler, 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, 2007.

[5] M. Tanimoto, M.P. Tehrani, T. Fujii, and T. Yendo, "Free-Viewpoint TV," *Signal Processing Magazine, IEEE*, vol. 28, no. 1, pp. 67 –76, January 2011.

[6] X. Huang and S. Forchhammer, "Cross-band noise model refinement for transform domain Wyner-Ziv video coding," *Signal Processing: Image Communication*, vol. 27, no. 1, pp. 16 – 30, 2012.

[7] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive codes for distributed source coding," *EURASIP Signal Processing Journal*, vol. 86, no. 11, pp. 3123–3130, November 2006.

[8] I. Daribo, C. Tillier, and B. Pesquet-Popescu, "Motion vector sharing and bitrate allocation for 3D video-plus-depth coding," *EURASIP J. Appl. Signal Process.*, vol. 2009, pp. 1–13, January 2008.

[9] X. Huang, L.L. Rakêt, H.V. Luong, M. Nielsen, F. Lauze, and S. Forchhammer, "Multi-hypothesis transform domain Wyner-Ziv video coding including optical flow," in *Proc. of IEEE MMSP*, October 2011, pp. 1 –6.

[10] H. Oh and Y.-S. Ho, "H.264-based depth map sequence coding using motion information of corresponding texture video," in *Proc. of PSIVT*, Berlin, Heidelberg, 2006, pp. 898–907, Springer-Verlag.

[11] M. Winken, H. Schwarz, and T. Wiegand, "Motion vector inheritance for high efficiency 3D video plus depth coding," in *Proc. of IEEE PCS*, May 2012, pp. 53 –56.

[12] G. Petrazzuoli, M. Cagnazzo, F. Dufaux, and B. Pesquet-Popescu, "Wyner-Ziv coding for depth maps in multiview video-plus-depth," in *Proc. of IEEE ICIP*, September 2011, pp. 1817 –1820.

[13] M. Salmistraro, M. Zamarin, and S. Forchhammer, "Wyner-Ziv Coding of Depth Maps Exploiting Color Motion Information," *Proceedings of SPIE, the International Society for Optical Engineering*, vol. 8666, pp. 8666–14, 2013.

[14] Y. Nie and K.-K. Ma, "Adaptive rood pattern search for fast block-matching motion estimation," *Image Processing, IEEE Transactions on*, vol. 11, no. 12, pp. 1442 – 1449, December 2002.

[15] R. Szeliski, *Computer Vision: Algorithms and Applications*, Springer-Verlag New York, Inc., New York, NY, USA, 1st edition, 2010.

[16] C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime TV-L^1 optical flow," in *In Ann. Symp. German Association Patt. Recogn*, 2007, pp. 214–223.

[17] L.L. Rakêt, L. Roholm, M. Nielsen, and F. Lauze, "TV-L^1 optical flow for vector valued images," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*, Yuri Boykov *et al.*, Ed., vol. 6819 of *Lecture Notes in Computer Science*, pp. 329–343. Springer, 2011.

[18] L.L. Rakêt, L. Roholm, A. Bruhn, and J. Weickert, "Motion compensated frame interpolation with a symmetric optical flow constraint," in *Advances in Visual Computing*, George Bebis *et al.*, Ed., vol. 7431 of *Lecture Notes in Computer Science*, pp. 447–457. Springer Berlin Heidelberg, 2012.

[19] B. Goldluecke, E. Strekalovskiy, and D. Cremers, "The natural vectorial total variation which arises from geometric measure theory," *SIAM Journal on Imaging Sciences*, vol. 5, no. 2, pp. 537–563, 2012.

[20] L.C. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 600–608, 2004.

[21] "Extension of existing 3DV test set toward synthetic 3D video content," ISO/IEC JTC1/SC29/WG11, Doc. M19221, Daegu, Korea, January 2011.

[22] "DISCOVER project test conditions," December 2007, http://www.img.lx.it.pt/ discover/test_conditions.html.

[23] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," in *VCEG Meeting*, Austin, USA, April 2001.

[24] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: Architecture, techniques and evaluation," *Proc. of IEEE PCS*, November 2007.