Key view selection in distributed multiview coding

Thomas Maugey #, Giovanni Petrazzuoli *, Pascal Frossard #, Marco Cagnazzo *, Béatrice Pesquet-Popescu *

# *Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland*
{*thomas.maugey, pascal.frossard*}*@epfl.ch*

* *Télécom ParisTech, Paris, France*
{*petrazzu, cagnazzo, pesquet*}*@telecom-paristech.fr*

***Abstract*—Multiview image and video systems with large number of views lead to new problems in data representation, transmission and user interaction. In order to reduce the data volumes, most distributed multiview coding schemes exploit the inter-view redundancies at the decoder side, using view synthesis from key views. In the situation where many views are considered, the two following questions become fundamental: i) how many key views have to be chosen for keeping a good reconstruction quality with reasonable coding cost? ii) where to place them optimally in the multiview sequences? We propose in this paper an algorithm for selecting the key views in a distributed multiview coding scheme. Based on a novel metric for the correlation between the views, we formulate an optimization problem for the positioning of the key views such that both the distortion of the reconstruction and the coding rate cost are effectively minimized. We then propose a new optimization strategy based on shortest path algorithm that permits to determine both the optimal number of key views and their positions in the image set. We experimentally validate our solution in a practical distributed multiview coding system and we show that considering the 3D scene geometry in the key view positioning brings significant rate-distortion improvements compared to distance-based key view selection as it is commonly done in the literature.**

***Index Terms*—distributed multiview coding, key view positioning, inter-view correlation, depth-image based rendering**

I. Introduction

With the advent of immersive applications, users have the possibility to navigate among a large quantity of viewpoints. However, moving from traditional multiview sequences with 8 views [1] to sequences with 100 views for example [2] raises many questions, in particular about the effective encoding of such large amounts of correlated data. Effective solutions for multiview data could be constructed on distributed source coding principles. However, most distributed multiview encoders rely on a side information construction from reference views (called key views) and it is important to study the positioning of the key views in the large image set. Intuitively, the key views should be chosen such that the other images can be efficiently estimated. This problem is similar to the positioning of the key frames in a video sequence [3], which however follow a different correlation model than multiview image sets.

We first propose a novel correlation model that captures the redundancy between different views of the 3D scene. We consider multiview images only and we assume that the scene is either static or that the multiple views are captured at a unique instant. The proposed model states that an inter-view prediction based on depth maps (as it is done in efficient distributed multiview coding schemes [4], [5]) provides two different kinds of regions in the predicted image: a) the predicted pixels and b) the disoccluded areas[1]. This is true if the viewpoints remain at a constant distance from the scene, which is our assumption in this paper. In the region of estimated pixels, we assume that the reconstruction is error-free as long as the geometry information is accurate. In the disoccluded regions however, view prediction is not possible, so that no information is available. Our model thus assumes that the coding rate of the predicted views grows linearly with the size of these disocclusions and that the slope of this relationship is given by the average coding rate per pixel. Based on this model, we propose an original problem formulation to select both the number of key views and their position in the multiview image set. We solve the problem with a novel and computationally efficient shortest path algorithm. Our solution takes into account the geometry of the scene, when choosing the number of key views and their positions such that the reconstruction of the estimated view is done with an optimally low rate cost. We show in experiments with a state-of-the-art multiview DSC architecture [4] that the proposed key view positioning algorithm obtains significantly reduced coding cost compared to a traditional equidistant key view distribution that is blind to correlation between views. Hence, the multiview coding systems with high number of viewpoints that are considered in more and more application nowadays, may benefits from our correlation model and our efficient key view positioning solution.

II. Key view positioning

A. Scenario

We study here the scenario of multiple views capturing a static scene. In order to match the challenges posed by the new multiview applications, we assume that the image set is made of a high number N of images (*e.g.*, $N > 20$). We

[1] In the following, we also use the terms: disoccluded zones, regions or dissocclusion holes.

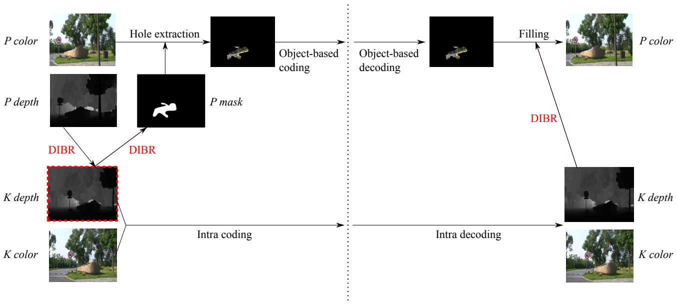

Fig. 1. Distributed multiview coding scheme proposed in [4] and adopted here.

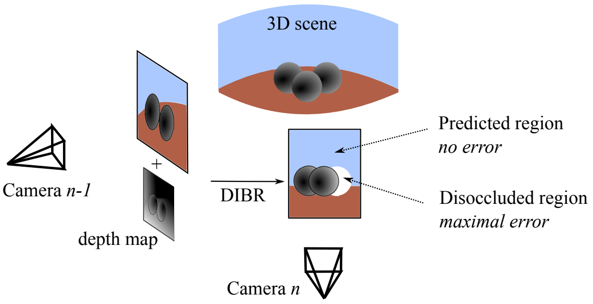

Fig. 2. Illustration of depth-image based rendering (DIBR) of camera n using key view $n-1$.

assume that these N images lie on a 1D view set within the 3D scene, and that they are not necessarily rectified. Additionally, we work under the hypothesis that the texture images, the depth map and the camera parameters are available for each of the N images. The availability of depth maps is supported by the arrival of depth sensors in the market, which makes its acquisition cheap and accurate [6]. The camera parameters are given by gyroscopic and GPS devices that equip every recent capture systems or even smartphones. They are sufficient to define the extrinsic parameters of a camera, including rotation and translation parameters [7].

We formulate our key view selection problem in the context of the distributed coding scheme proposed in [4], [5] that significantly outperforms other distributed multiview techniques. The N views of the 3D scene are split into key (K) and predicted (P) views, coded, and transmitted to users for decoding and reconstruction of the 3D scene. We assume to have N_K K views in the whole dataset, which are denoted by I_{i_k}, with $k \in \mathcal{K}$ and $\mathcal{K}$ is the set of indices for the K views. Each K view is associated to a segment (the set of segments is denoted by $\mathcal{S}$), which is the set of views that are estimated from the K view. These segments are a Group Of Pictures (GOP) in the view direction [4].

The key views are simply Intra encoded (for example with H.264/INTRA), while only the unpredictable regions of the P views are encoded, *i.e.,* the portions of the image that cannot be estimated from neighboring views at the decoder. The pixels belonging to these areas are detected distributively, at the encoders, using a double depth-image based rendering (DIBR) [8]: the depth map of the predicted view is firstly projected onto the key view projection, and, after a Bertalmio inpainting of the disoccluded zones, the resulting depth map is projected back onto the predicted view position (see Fig. 1). The resulting disocclusions holes are an estimation of the positions of the disoccluded areas after the DIBR prediction performed at the decoder. These areas are coded using object-based coding technique [9]. We assume, as in [4], that depth maps are also coded, but at a sufficiently high quality to preserve the DIBR accuracy.

At the decoder, each P view is estimated by warping the K view and filling the holes with the coded disocclusions. The depth maps are only used to perform prediction (no virtual viewpoint is synthesized in this framework). In our system, the parameters $(\mathcal{K}, \mathcal{S})$ defines the general prediction structure of the multiview encoder. Finding the optimal values of these parameters is the objective of our paper.

B. Coding rate model

Inter-view estimations aim at building the best possible estimation of a view from the information available in the reference viewpoint. The purpose is to minimize the coding rate for predicted views, *i.e.*,, the number of bits required to code the predicted views, which depends on the level of similarity between the predicted view and the reference one. There exists a simple relationship in coding: the better the estimation, the smaller the information needed for complementing the view reconstruction. Hence, we expect a gain in the rate-distortion performance for coders that perform good predictions.

When DIBR is used for inter-view estimation (see Fig. 2), we cannot expect huge improvement of prediction effectiveness if the geometry information is already accurate, since projected pixels perfectly match with the corresponding ones in the target view. Therefore, the innovation that is needed for view reconstruction only corresponds to the *disoccluded regions*. Intuitively, the coding rate for predicted frames grows with the size of these disocclusions. More precisely, if I_1 and I_2 are two images, the disocclusion size can be measured from the size of the projection of I_1 onto I_2 (*e.g.*, I_1 can estimate 80% of I_2, the disocclusion size is thus 20%, see Fig. 2). We introduce the metric $\gamma(I_2, I_1)$ that gives the size (*i.e.* the relative number of pixels) of the region in I_2 that cannot be predicted from the key view I_1. It is called *dissimilarity* in the following. In this work, we assume that for a constant distortion, the coding rate of a predicted frame evolves linearly with its dissimilarity with the K view used as reference. If we call R_P the per pixel bit-rate used for coding the occluded zones in the predicted frames, the total number of bits used for coding the P frames, let it be r_P, varies as follows:

$$r_P(I_P, I_K) = M\gamma(I_P, I_K)R_P, \qquad (1)$$

where M is the number of pixels in the image (we write $\rho = M \cdot R_P$ in the following).

We naturally that the relative positions of both the reference and predicted views impacts on the coding rate, and that the disocclusion size does not only depend on the distance between the two viewpoints but also varies with the properties of the 3D scene. The key view selection problem is thus driven by the geometry of the scene and leads to the dependent positioning of N_K reference cameras along the navigation path (in addition of the coding cost of these key views). The model proposed in Eq. (1) is the basis of our formulation proposed in next section.

III. Key View Selection Algorithm

From now on, we target a constant quality over the view set. Therefore, the rate of the key views, which are assumed to be coded similarly, is set at r_k (which is an input variable of the following problem). The coding rate of the non key views depends on the position of the key views used for prediction. A solution of our problem is thus characterized by the number of key views, N_K, the segments $\mathcal{S} = \{S_1, \ldots, S_{N_K}\}$ that contains the indices of the P views, and their associated key views indices $\mathcal{K} = \{i_1, \ldots, i_k, \ldots, i_{N_K}\}$.

The problem of key view positioning is defined by the following minimization:

$$\begin{aligned}(N_K^*, \mathcal{K}^*, \mathcal{S}^*) &= \underset{(N_K,\mathcal{K},\mathcal{S})}{\arg\min} \sum_{k=1}^{N_K} \left(r_k + \sum_{j \in S_k} r_P(I_j, I_{i_k}) \right) \\ &= \underset{(N_K,\mathcal{K},\mathcal{S})}{\arg\min} \sum_{k=1}^{N_K} \left(r_k + \sum_{j \in S_k} \rho\gamma(I_j, I_{i_k}) \right) \quad (2)\end{aligned}$$

where, $i_k \in \mathcal{K}$ corresponds to the index of the key views. The first term in the parenthesis is the rate of the K view, while the second term is the rate cost of the P views.

In order to find the optimal solution $(N_K^*, \mathcal{K}^*, \mathcal{S}^*)$, we build the graph illustrated in Fig. 3. Horizontal connections in this graph are set to $\rho\gamma(I_j, I_i)$, *i.e.*, the value of the rate of P view I_j using I_i as key view (Eq. (1)). Each node (i, j) of the upper triangular graph (*i.e.*, $j > i$) is vertically connected to all the nodes (i', j) of the lower triangular graph (*i.e.*, $i' > i$). Vertical connections symbolize a new segment. Therefore their attached weight is r_K. The objective of the optimization problem (2) is equivalent to finding the best path between the point $(1, 1)$ and (N, N) in this graph. The path has to satisfy the two following constraints: i) as soon as it goes horizontally, it must continue at least to the diagonal (included), ii) as soon at it goes vertically, it also has to cross the diagonal. The obtained candidate paths are thus made of a certain number of horizontal segments. This number corresponds to the number of key views N_K. As an example, let us take a segment at a line i going from $i - l$ until $i + l'$. The interpretation of this segment is the following: the key view is i and all views from $i - l$ until $i + l'$ are estimated with view i. The views from $i - l$ until $i - 1$ are predicted in the backward direction. If we calculate the cost of this part of the path, it is equal to: $\sum_{j=i-l}^{i+l'} \rho\gamma(I_j, I_i)$. If we add now the cost of a vertical transition, r_K, we obtain an overall cost equivalent to the cost in Eq. (2). Therefore, if we run a shortest path algorithm (*e.g.*, Djikstra [10]) we obtain the minimal cost, hence the optimal values for N_K^*, $\mathcal{K}^*$ and $\mathcal{S}^*$.

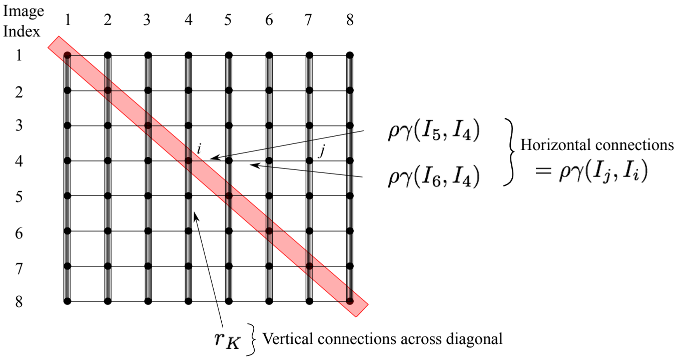

Fig. 3. Graph initialization for shortest path algorithm: horizontal connections are weighted by $\gamma(I_i, I_j)$ and all vertical ones that cross the diagonal are set to r_K.

IV. Experiments

In these experiments, we use the *bikes* and *statue* multiview datasets provided by Disney Research in [2]. In order to avoid to obtain, as optimal solution, a trivial key view positioning, we have sub-sampled irregularly the 50 views provided by [2]. For example, for *bikes*, we have selected 25 views: 1, 5, 9, 13, 17, 18, 21, 24, 27, 30, 33, 36, 37, 39, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51 and we have found the optimal path and the optimal number of K views using the proposed algorithm. We preliminarly compute the ratio $\frac{R_P}{R_K}$ for each sequence, where R_P and R_K are the bit rates (in bits per pixels) necessary for obtaining the same distortion on the P views (on the occlusion zones) and on the texture K views, respectively. This ratio is averaged for all the available views and for different values of QP for the K views. We have obtained a value of 0.98 and 3.86 for respectively *bikes* and *statue* datasets. This ratio strongly depends on the statistical properties of the occlusion regions in the P-frames.

We have computed the matrix Γ (made of all the $\gamma(I_i, I_j)$) for both *bikes* and *statue* multiview datasets and we have calculated the optimal positioning using our original formulation. Results are shown in Fig. 4 for *bikes* datasets (similar results are obtained for *statue*). Fig. 4(b) represents the dissimilarity matrix (white is 1 and black is 0). A point (i, j) in this matrix indicates the similarity of view j using view i as reference. Two partitioning solutions are represented in this matrix by two paths going from the left border to the right border. Each horizontal segment on row i points the views (column indices) belonging to a segment attached to the key view i. The blue positioning solution is the equidistant[2] while the red one is the results of our proposed partitioning. We see that our key view positioning solution follows better the variations of the correlations. More precisely, the size of the segments are larger for navigation subsets where the views are highly correlated. Reversely, the navigation segments become smaller for low

[2] In the sense of view index.

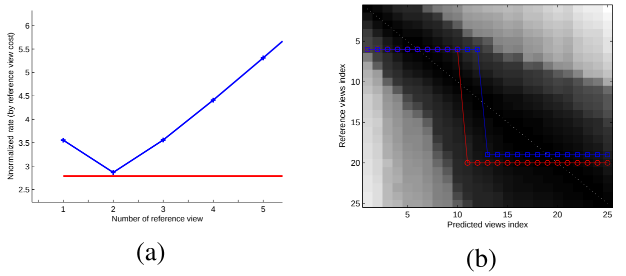

Fig. 4. Positioning results for *bikes* datasets (51 views). Blue and red curves respectively correspond to equidistant and optimized reference frame positioning. Left figure is the rate cost for different values of N_K and right image shows the corresponding positioning.

correlation regions. We additionally evaluate the cost of these partitionings with Eq. (2) and show it in Fig. 4(a). The cost is expressed as a normalized cost, where the normalization factor is the cost of a key view. We see that a first interest of our technique resides in the fact that it directly finds the optimal number of key views, contrary to the blind equidistant one, which needs to perform a full search over all the key view numbers. Moreover, we see that our positioning, in red, leads to a rate reduction compared to the blue equidistant positioning for different N_K values.

Then, we have run the distributed source coder proposed in [4] with the segments shown in Fig. 4. The texture K views are coded with H.264/AVC at four different QPs, namely 31, 34, 37 and 40. The corresponding QP for depth reference view is chosen according the empirical rule proposed in [11]. We have computed the Rate Distortion performance by comparing with an equidistant scheme with the same number of K views N_K, as shown in Fig. 4.

The rate distortion curves for *bikes* and *statue* multiview datasets are in Fig. 5. According to Fig. 4, we can remark that a bit rate reduction is achieved, and in addition we have found that we the optimal numbers of K views without performing a full search. The bit rate reduction measured by Bjontegaard metric are -7.73% and -47.75% for *bikes* and *statue* multiview datasets.

Unfortunately, the depth maps provided by [2] are not perfect: they are computed by using a dense disparity estimation algorithm. The consequence is that we cannot assure that the PSNR on the K views is the same as the PSNR computed on the synthesized areas for the P views. Nevertheless, the RD performance of the optimized scheme outperforms the one of the equidistant key-view positioning.

V. Conclusion

In this paper, we have proposed a solution to determine the optimal number of key views along with their positions, in a distributed multiview coding system. Based on an original correlation model, we have formulated the optimization problem as a derivation of a shortest path algorithm. The obtained results demonstrate the potential of our approach and show the benefits of exploiting inter-view correlation when positioning the key views in the multiview dataset. Our algorithm can typically be implemented between the capture and the encoding process, in the context of stored or on-demand content for example (depth-image based rendering operations are easily parallelizable). Our method could further be simplified if needed, by using coarser versions of the images (e.g., downsampled images) or of the algorithm (e.g., block-based algorithm). Moreover the geometry could be estimated only periodically, as it generally evolves more slowly than the frame rate. As a consequence, our algorithm can be implemented in practice and may be interesting to reduce significantly the coding costs. Future work will focus on extending this study to any other scenario where views are predicted as, for example, in most of the standard multiview coding schemes.

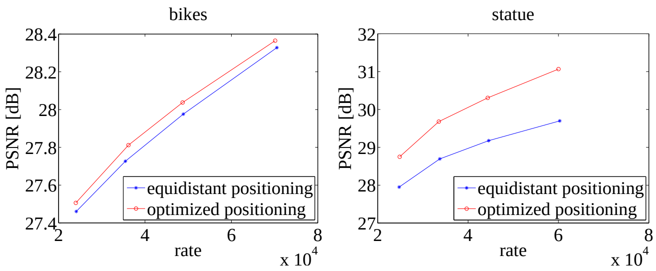

Fig. 5. Rate distortion performance for *bikes* and *statue* multiview datasets.

References

[1] [Online]. Available: http://research.microsoft.com/en-us/um/people/sbkang/3dvideodownload/

[2] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. Gross, “Scene reconstruction from high spatio-angular resolution light fields,” *ACM Transactions on Graphics*, vol. 32, no. 4, 2013.

[3] E. Masala, Y. Yu, and X. He, “Content-based group-of-picture size control in distributed video coding,” *Signal Processing: Image Communication*, vol. 2014, pp. 332–344, Feb. 2014.

[4] G. Petrazzuoli, T. Maugey, M. Cagnazzo, and B. Pesquet-Popescu, “A distributed video coding system for multi-view video plus depth,” in *Asilomar Conference on Signals, Systems and Computers*, vol. 1, Pacific Groove, CA, 2013, pp. 699–703.

[5] ——, “Depth-based multiview distributed video coding,” *IEEE Trans. on Multimedia*, vol. 16, no. 7, nov. 2014.

[6] G. Alenya and C. Torras, “Lock-in time-of-flight (TOF) cameras: A survey,” *IEEE Sensors Journal*, vol. 11, pp. 1917–1926, 2011.

[7] D. Tian, P. Lai, P. Lopez, and C. Gomila, “View synthesis techniques for 3D videos,” *Proc. of SPIE, the Int. Soc. for Optical Engineering*, vol. 7443, 2009.

[8] C. Fehn, “Depth-image-based rendering (DIBR), compression and transmission for a new approach on 3D-TV,” *Proc. SPIE, Stereoscopic Image Process. Render.*, vol. 5291, pp. 93–104, 2004.

[9] M. Cagnazzo, G. Poggi, and L. Verdoliva, “Region-based transform coding of multispectral images,” *IEEE Trans. on Image Proc.*, vol. 16, no. 12, pp. 2916–2926, Dec. 2007.

[10] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, 1st ed. Addison-Wesley Professional, January 1989.

[11] D. Rusanovsky, K. Muller, and A. Vetro, “Common test conditions of 3DV core experiments,” January 2013, ITU-T SG 16 WP 3 & ISO/IEC JTC1/SC29/WG11 JCT3V-C1100.