Reference view selection in DIBR-based multiview coding

Thomas Maugey, *Member IEEE,* Giovanni Petrazzuoli, Pascal Frossard, *Senior Member IEEE* and Marco Cagnazzo, *Senior Member IEEE* and Béatrice Pesquet-Popescu, *Fellow Member IEEE*

***Abstract*—Augmented reality, interactive navigation in 3D scenes, multiview video and other emerging multimedia applications require large sets of images hence larger data volumes and increased resources compared to traditional video services. The significant increase of the number of images in multiview systems leads to new challenging problems in data representation and data transmission to provide high quality of experience on resource-constrained environments. In order to reduce the size of the data, different multi view video compression strategies have been proposed recently. Most of them use the concept of reference or key views that are used to estimate other images when there is high correlation in the dataset. In such coding schemes, the two following questions become fundamental: i) how many reference views have to be chosen for keeping a good reconstruction quality under coding cost constraints? ii) where to place these key views in the multiview dataset? As these questions are largely overlooked in the literature, we study the reference view selection problem and propose an algorithm for the optimal selection of reference views in multiview coding systems. Based on a novel metric that measures the similarity between the views, we formulate an optimization problem for the positioning of the reference views such that both the distortion of the view reconstruction and the coding rate cost are minimized. We solve this new problem with a shortest path algorithm that determines both the optimal number of reference views and their positions in the image set. We experimentally validate our solution in a practical multiview distributed coding system and in the standardized 3D-HEVC multi view coding scheme. We show that considering the 3D scene geometry in the reference view positioning problem brings significant rate-distortion improvements and outperforms traditional coding strategy that simply selects key frames based on the distance between cameras.**

***Index Terms*—Multiview distributed coding, key view positioning, inter-view correlation, view synthesis, multiview image coding**

I. Introduction

Several new applications based on multiview transmission systems have been recently developed, such as immersive communications, interactive systems, navigation in a 3D environment (see Fig. 1), etc [1], [2]. Such systems require large data volumes to describe the visual information in potentially complex 3D scenes. The most common approaches to represent such visual information rely on image-based models, which are built on sets of views that capture the 3D scene for several different viewpoints. The image-based representation is well aligned with the current capture and rendering hardware systems, which typically acquire 2D images and display images on 2D screens[1]. The main drawback of image-based models is however the large redundancy between different views that increases significantly the size of the data. A point in the 3D scene is generally visible from multiple viewpoints and thus represented in multiple views. The size of the data increases with the number of views of the 3D scene, while the new information (*i.e.,* the actual new pixels) in the representation increases much slower. Hence, there is a need for effective multiview image coding techniques that can reduce the inter-view redundancy hence the data size. This is especially the case for the development of systems that consider 100 [3] or 1600 [4] views, for example, for navigation. This can be achieved with inter-view prediction techniques that rely on disparity vector fields [5], [6] or depth maps [7], [8], [9].

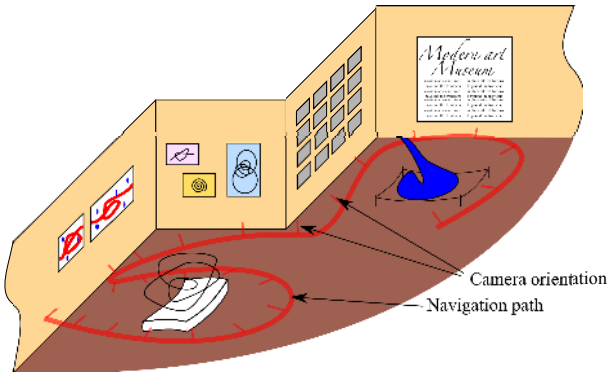


Fig. 1. Interactive multiview navigation application. Example of a navigation path, made of N viewpoints, in a static scene. In such scenario, N is large.

In order to enable view prediction, most coding schemes in the literature rely on the concept of *reference view*, which is by definition the view used for the prediction of the other ones. These views are sometimes called base views in the video coding standards [10], key pictures in distributed video coding schemes [11], or reference views in interactive schemes [12]. One can rapidly observe that the positioning of such reference views has a large impact on the quality of the inter-view prediction, hence in the distortion and the coding rate of the multiview representation. The problem of reference view positioning has been mostly overlooked in the literature. It bears some resemblances with the positioning of reference frames in video sequences. This has been studied for both standard [13], [14] and distributed video coding schemes [15], [16]. The reference frame positions are adjusted based on the content of video sequences. Typically, a high speed motion video will profit from a larger number of reference frames. Such works mostly rely on heuristics or simple motion modeling but demonstrate that coding gains are achieved with proper positioning of reference views. These techniques cannot however be applied to multi view coding, where the correlation information comes from geometric considerations, hence very different from motion in video sequences.

In this paper, we study the problem of selecting the reference views for prediction-based coding of multiview datasets. We first propose and validate a novel similarity model that captures the redundancy between different views of the 3D scene. We assume that the prediction across views is based on *Depth Image-Based Rendering (DIBR)* techniques. These are the most efficient approaches for

Thomas Maugey is with the team-project SIROCCO at INRIA/IRISA, Campus Beaulieu, 35042 Rennes, France (e-mail: thomas.maugey@inria.fr).

Pascal Frossard is with the Signal Processing Laboratory (LTS4), Institute of Electrical Engineering, École Polytechnique Fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland (e-mail: pascal.frossard@epfl.ch).

G. Petrazzuoli, M. Cagnazzo, and B. Pesquet-Popescu are with the Department of Image and Signal Processing, Institut Mines-Telecom, Paris F-75634, France, and with the Laboratory for Communication and Processing of Information (LTCI), Centre National de la Recherche Scientifique (CNRS), Paris, France (e-mail: giov.petr@gmail.com; cagnazzo@telecom-paristech. fr; pesquet@telecom-paristech.fr).

This work was partially funded by the Hasler Foundation under the project NORIA (Novel image representation for future interactive multiview systems)

[1]We note that the stereo scenario is included in the image-based model, since stereo images are two 2D views in a particular setting.

prediction, since each pixel of the reference view is projected on its position on the predicted view using depth information [17]. Our new model states that an inter-view prediction based on depth maps provides two different kinds of regions in the predicted view: a) the predictable pixels and b) the disoccluded areas. In the region of predictable pixels, the reconstruction by DIBR does not introduce any distortion as long as the geometry information is accurate. In the disoccluded regions however, inter-view prediction is not possible, and the prediction error can be very large. Additional information has therefore to be coded for these regions. We show here that the coding rate of the predicted views grows linearly with the size of the disoccluded areas for a constant distortion. We demonstrate the validity of this model on multiple multiview sequences and we build a rate-distortion model for multiview encoders, which is used in the reference view selection problem. We then formulate this problem for the general scenario where views are predicted by one or several reference viewpoints in order to be generic and cover various multiview coding schemes. The advantage of our solution is thus to take into account the geometry of the scene, when choosing the number of key views and their positions such that the representation of the predicted view is done with the optimally low coding cost. We propose an original problem formulation to select both the number of reference views and their position, such that a proper tradeoff between distortion and coding rate can be achieved. We eventually solve the problem with a new shortest path algorithm. The shortest path formulation permits to avoid a complex full search algorithm on both the number of reference views and their positions. Other works have proposed to optimally choose the set of views to be coded in a multi-view plus depth scenario [18], [19]. Among a set of views captured by the acquisition system, the sender is able to discard some views in the coding process and eventually synthesize them at the receiver. These works have also proposed a rate allocation algorithm between depth and texture signals for the selected views. The objective of these algorithms is however different from the one pursued in this work, namely the choice of the reference views in a prediction-based coding scheme where all views are transmitted and not only a part of them.

We then test our reference view selection algorithm in coding experiments with both the 3DVC framework [20] and a state-of-the-art multiview DSC algorithm proposed in [21], [22]. The latter consists in using depth images for correlation estimation at the encoder side, and side information generation at the decoder side. The reference views are transmitted alone, while for WZ views we send only the occluded regions by shape adaptive algorithms [23], [24]. We experimentally demonstrate in both schemes the great potential of the proposed reference view positioning algorithm. First, it determines the optimal number of reference views, and thus avoids a complex full search over all the possible numbers of reference views. Second, the optimal positioning of reference views leads to significantly reduced coding cost compared to a traditional equidistant key view distribution that is blind to correlation between views. The proposed work thus offers an efficient tool to optimize the coding structure in multiview settings and to reduce storage and bandwidth resources for emerging multiview applications.

The remaining of the paper is structured as follows. In Section II, we describe the framework of the reference view positioning problem and depict the main ideas of our solution. In Section III, we introduce our novel model for inter-view similarity. Then, in Section IV, we describe in detail the problem formulation and our shortest path optimization algorithm. Finally, in Section V, we evaluate the benefit of our optimal reference view positioning algorithm for two representative multiview coding methods, namely a recent distributed video coding scheme and a traditional multiview video coding approach.

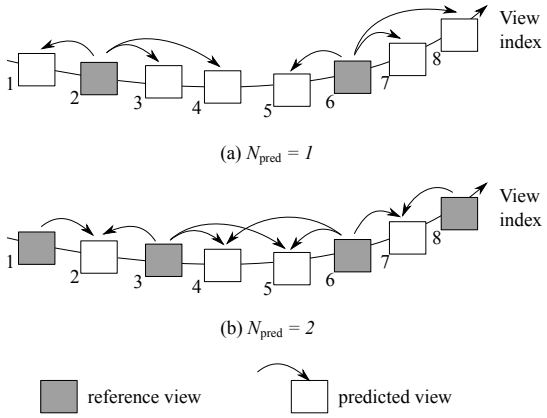


Fig. 2. Example of segment partitioning of a set of 8 camera views for (a) $N_{\text{pred}} = 1$ and (b) $N_{\text{pred}} = 2$. The key views are respectively the sets $\mathcal{K} = \{2, 6\}$ and $\mathcal{K} = \{\{1, 3\}, \{3, 6\}, \{6, 8\}\}$ and their attached segments are respectively the sets $S = \{\{1, 3, 4\}, \{5, 7, 8\}\}$ and $S = \{\{2\}, \{4, 5\}, \{7\}\}$.

II. Reference view positioning

A. Framework

We study here the scenario of multiple camera 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 views (*e.g.*, $N > 10$). The N views of the 3D scene have to be coded, and transmitted to users for decoding and reconstruction of the 3D scene. We assume that these N views are arranged on a 1D path within the 3D scene (*i.e.,* they can be indexed with an integer between 1 and N), and that they are not necessarily rectified. Additionally, we suppose that the texture image, the depth map and the camera parameters are available for each of the N views. The availability of depth maps is justified by the arrival of depth sensors in the market, which makes depth acquisition cheap and accurate [25]. The camera parameters are known or estimated from gyroscopic and GPS devices that equip every recent capture systems or even smartphones. They are sufficient to define the extrinsic parameters of a camera, namely the rotation and translation parameters [26].

We consider a generic framework that does not depend on the specific multiview coder as long as it involves view prediction (*e.g.*, predictive coder or distributed coder). There are N_{K} reference views (also called key views in the following) in the whole dataset, and $\mathcal{K}$ is the set of indices for these reference views. The reference views define segments that are sets of consecutive views predicted from the references via DIBR [10]. The number of reference views used for the generation of one predicted view is denoted by N_{pred} and is fixed for every segment (it depends for example on the configuration and the adopted coder). The set of segments is denoted by $\mathcal{S}$. The segments are analogous of the Group Of Pictures (GOP) in video coding in the sense that they are DIBR-predicted views from the same key views. An example of the view arrangement is given in Fig. 2 for $N_{\text{pred}} = 1$ and 2. We assume in the following that the depth maps are also coded, but at a sufficiently high quality to preserve the DIBR accuracy (*e.g.,* by using contour preserving techniques [27], [28], [29]). At the decoder side, depth maps are only used to perform prediction of camera views (*i.e.,* we do not consider virtual viewpoint in our framework). In the above framework, the couple of sets $(\mathcal{K}, \mathcal{S})$ defines the general prediction structure of the multiview encoder (see Fig. 2 for an example). Finding the optimal configuration of these sets is the main objective of our paper.

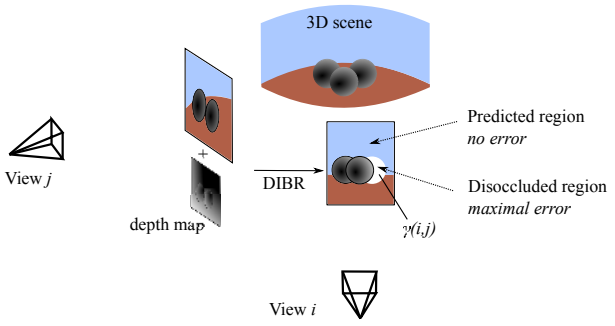


Fig. 3. Illustration of depth-image based rendering (DIBR) of camera n using reference view $n-1$.

B. Challenges

The purpose of the reference view selection problem is to determine the key views and segment sets $(\mathcal{K}, \mathcal{S})$ for a given quality such that the coding of the overall image dataset is effective from a rate-distortion perspective. The goal is thus to minimize the global coding rate of both key and predicted views, which depends on the number of key views and the innovation in the predicted views with respect to the reference ones. In general, the better the prediction by inter-view estimation, the smaller the information needed for completing the view reconstruction and the better the rate-distortion performance of the multiview encoder. When DIBR is used for inter-view prediction (see Fig. 3), the prediction effectiveness is high if the geometry information is accurate, since the projected pixels perfectly correspond to the ones in the target view. Therefore, in our hypothesis, the additional information that is needed for view reconstruction only corresponds to the *disoccluded region*. This is the part of the predicted view that is occluded in the reference view (*e.g.*, it is hidden by a foreground object or out of the image boundaries). The coding rate for this additional information grows with the size of the occlusion, which further depends on relative positions of both the reference and predicted views. More precisely, if these positions are not carefully controlled in the compression scheme, the disocclusion area can be large and the coding rate is high. In this case, the reference view is not very similar to the predicted one. It is important to note that the size of the disocclusion region does not only depend on the distance between the two viewpoints but also varies with the geometrical properties of the 3D scene. For example, a distance of 10 cm between two cameras leads to different disocclusion sizes depending if the objects in the scene are at 1 or 10 meters from the cameras.

Overall, the coding rate and therefore the optimal reference view selection are driven by the geometry of the scene and the dependencies between the positioning of the N_{K} reference cameras along the navigation path. The reference view selection problem is further driven by the coding cost of these reference views, which are generally more expensive than the predictive views. This implies the need of optimizing the number of key views N_{K}.

The optimization of the key frame positioning therefore consists in finding jointly the optimal number of key views (N_{K}) their positions on the navigation paths, $\mathcal{K}$ and the corresponding segments of predicted views, $\mathcal{S}$. This is achieved by minimizing the disocclusion sizes hence the coding rate, for each predicted view in a segment. In other words, we should choose the reference views in order to maximize the overall similarity between the reference views and the predicted views. The optimization further deals with the following tradeoff: choosing a higher number of optimally positioned reference views reduces the dissimilarity with the predicted views but at the same time, the total rate for reference views increases. We introduce below our rate-distortion model, which is used later for finding the optimal key frame positioning.

III. View similarity model

A. Dissimilarity metric

In this section, we study the relationship between disocclusion size and coding rate. We propose a new *dissimilarity* metric between two views which is related to the size of the disocclusion in the view prediction. We then derive a new rate-distortion model that links this dissimilarity metric with the coding performance. With accurate depth maps, we can compute the inter-view similarity exactly [21] as the correlation between views is driven by the geometry information. The similarity between views can be related to the image region that is common in two views. This concept introduced in [30] holds under two hypotheses that are commonly used in multiview video applications: i) the scene is lambertian (*i.e.,* a point in the 3D scene is observed under similar illumination conditions from the different viewpoints) and ii) the navigation path remains approximately at a uniform distance from the scene[2].

Equipped with the above notion, we introduce here the *dissimilarity* metric γ to measure the difference between views. This metric corresponds to the normalized size of the region that remains uncovered by the reference view. More precisely, if I_i and I_j are two different views, we compute the dissimilarity between them by measuring the size of the uncovered region after the DIBR-based projection of I_j onto I_i. For example, if I_j can estimate 80% of I_i, the dissimilarity $\gamma(i,j)$ is equal to 20% (see Fig. 3). This definition can be extended to the case where multiple reference views are used to predict a view. If I_i is the predicted view and $\mathcal{J} \subset \mathcal{K}$ the set of reference views, we denote by $\gamma(i,\mathcal{J})$ the size of the region in I_i, which cannot be predicted by the reference views in $\mathcal{J}$. The dissimilarity values can be computed for any multiview dataset. It depends on the depth maps and the intrinsic and extrinsic camera parameters. For the particular case of one reference view per segment, we can for example compute a matrix Γ, where the element of the i-th row and j-th row is equal to the dissimilarity $\gamma(i,j)$ between views I_i and I_j. Note that such a matrix Γ is not necessarily symmetric, since it may happen that $\gamma(i,j) \neq \gamma(j,i)$. For example, in the case of two views translated along the camera axis, the disocclusion is larger when the view synthesis is done in the backward direction since a portion of the scene may appear at the image border. However, under the earlier assumption that the navigation path remains at a uniform distance from the scene, the difference between $\gamma(i,j)$ and $\gamma(j,i)$ remains limited. We further note that, when the number of reference views is higher than one (*i.e.,* when the predicted views are interpolated rather than extrapolated), the disocclusion size (or dissimilarity) is often very small ($< 0.1\%$), except in some peculiar cases. Overall, the exact evaluation of the dissimilarity can be very complex, especially with multiple reference frames. In order to circumvent this issue, we redefine the dissimilarity with the following linear combination:

$$\gamma(i,\mathcal{J}) = \sum_{j \in \mathcal{J}} a_j \gamma(i,j), \tag{1}$$

where a_j is a coefficient modeling the influence of the reference view I_j in the interpolation of view I_i. In practice, we evaluate these coefficients based on the codec characteristics and on the importance given to the different reference views. The choice of these coefficients has not been optimized yet.

Finally we note that, for sake of simplicity, we assume that the cameras lie on a 1D path, i.e., the view sets follows a 1D curve

[2]This assumption is made so that the resolution of the color texture remains similar in the different views

N_{pred}	[22] 1	3D-HEVC 1	3D-HEVC 2
mansion	99.94	97.09	94.47
bikes	97.93	97.99	87.32
statue	99.97	97.01	93.58
church	99.12	98.43	96.23

TABLE I
ABSOLUTE VALUE OF THE PEARSON COEFFICIENT [IN %] FOR THE PAIR (DISSIMILARITY, BIT-RATE PER PIXEL), IN THE CASE OF ONE OR TWO REFERENCE VIEWS USED FOR PREDICTION (N_{pred}), USING THE DISTRIBUTED VIDEO CODER PROPOSED IN [22] AND 3D-HEVC.

trajectory. In contrary to most of the works in the literature where the cameras are constrained to be placed in a simple configuration, the 1D navigation path considered in our work can however contain complex camera transitions (with large rotations and translations).

B. Rate versus dissimilarity

We now relate the dissimilarity between views to the coding rate that is necessary for the non-key views. In particular, we formulate the hypothesis that the coding rate of a predicted view increases linearly with the dissimilarity with respect to the reference view and we validate this intuition. For that purpose we use two predictive coder frameworks: the distributed codec proposed in [22] and the 3D-HEVC codec [31]. For both coders, we consider the case where all views are coded with the same distortion. This is done to guarantee a consistent inter-view navigation, and to avoid flickering effect [32].

In the context of the video coder proposed in [22], the predicted views are generated using a DIBR algorithm based on one reference view coded in intra mode. The disoccluded regions that cannot be estimated by DIBR are coded with a shape adaptive compression algorithm [23], [24]. In Fig. 4, we show the evolution of the rate used to code the predicted views as a function of their dissimilarity with respect to the key view for *mansion*, *bikes*, *statue* and *church* sequences, for a constant value of the mean squared error (MSE) calculated w.r.t. the original view that is assumed to be available. We see that the relationship is nearly linear. We have also computed the absolute Pearson coefficient between the dissimilarity and the bit-rate necessary for having a certain distortion for each sequence (*i.e.,* for a given quantization step size). The coefficients are given in Tab. I, where we remark that, in the worst case, there is a linear relationship of more than 97.93% between the dissimilarity and the bit-rate.

We now present a similar experiment with a more conventional coder, namely 3D-HEVC. We have run the 3D-HEVC for P-frames ($N_{\text{pred}} = 1$) and B-frames ($N_{\text{pred}} = 2$), *i.e.,* with one and two reference views respectively, in different positions and distances from the reference frames under constant QP (30) that implies a likely variable distortion. We have obtained the curves in Fig. 5. In these figures, we show the coding rate for the P- and B-Frames as a function of the dissimilarity between the predicted and the reference view(s). For the example with B-frames, the GOP is made of one intra frame (I_1) and one predicted frame (I_3), and one bidirectionally predicted frame (I_2) that is interpolated between the two frames. We plot the rate of the latter frame as a function of the dissimilarity with the other two. The coefficients a_1 and a_3 in Eq.(1) are set to $3/4$ and $1/4$ since the intra view I_1 has more influence on the bitrate coding of I_2 than the predicted view I_3. We observe from both the curves in Fig. 5 and Tab. I that the coding rate of a predicted view grows almost linearly with the dissimilarity, which allows us to introduce our rate model in the next section.

Finally, we consider that the effect of depth compression on the rate model is negligible. Indeed, in our model, the depth maps are used to estimate the similarity between views, i.e., the size of the occlusions. While compressed depth maps indeed bring inaccuracies on the projected pixels positions, it does not significantly change the size of the occlusions, which is the information that drives the view similarity estimation.

C. Rate Model

We recall that we impose that the quality over the views is constant. In other words, the rate of each view is set in order to achieve a predefined distortion value, such that the user navigation experience is pleasant (*i.e.,* there is no variation of the view quality during the navigation). From the experiments obtained above, we can derive an affine model for the coding rate depending on the percentage of occluded zones in the predicted views. Let us consider a view I_i that is predicted from one or multiple reference views described by the view set $\mathcal{J}$. The proposed model reads

$$r_{\text{P}}(i, \mathcal{J}) = \rho(D)\gamma(i, \mathcal{J}) + r_0(D), \tag{2}$$

where $\rho(D)$ is the slope and depends on the targeted distortion D, and $r_0(D)$ is the rate of the information that is transmitted when the occlusion size is zero (*i.e.,* for the geometry error correction) and also depends on D. The parameters $\rho(D)$ and $r_0(D)$ are estimated by linear fitting for each sequence and coder. In the following, we will sometimes drop the dependency regarding D for sake of conciseness.

The proposed model Eq. (2) relies on the observation that the coding rate of a predicted view is roughly linearly dependent on the dissimilarity with respect to the reference views [30]. Hence, the knowledge of the geometry of the scene (*e.g.*, the dissimilarity between the views) permits to model the coding rates for a given target distortion.

D. Rate allocation

We now discuss the allocation of the coding rate between reference and predicted views for a constant distortion over all views. It relies on the fact that, as said before, we want the distortion to be constant in order to guarantee a constant quality during multiview navigation and a good user experience. Each view i in the navigation path is characterized by a rate-distortion function for high bit-rate ([33]):

$$D_i(R_i) = \mu_i \sigma_i^2 2^{-\beta_i R_i},$$

where the bitrate R_i is expressed in bit per pixel and μ_i, σ_i and β_i are three parameters that depend on the source distribution. In more details, σ is the variance of the source, μ and β are two parameters depending on the distribution (e.g., Laplacian or Gaussian) [34]. We assume that all the key views have the same source characteristics (variance, distribution, etc.). In other words,

$$\forall i \in \mathcal{K}, \quad D_i(R_i) = D_{\text{K}}(R_i) = \mu_{\text{K}} \sigma_{\text{K}}^2 2^{-\beta_{\text{K}} R_i} \tag{3}$$

where μ_{K}, σ_{K} and β_{K} are the parameters corresponding to the key views. Furthermore, the distortion of the $N - N_{\text{K}}$ predicted views is equal to D_{K} on the regions covered by a DIBR-based projection from one or several key view(s) (if the depth data is perfect). Similarly to the above model, the distortion on the rest of the image (*i.e.,* the occluded areas) is equal to

$$\forall i \in [1, N] \setminus \mathcal{K}, \quad D_i(R_i) = D_{\text{P}}(R_i) = \mu_{\text{P}} \sigma_{\text{P}}^2 2^{-\beta_{\text{P}} R_i}. \tag{4}$$

We fix the view quality by fixing the rate of the key views R_{K}. Using Eq. (3) and (4), we determine the bitrate of the predicted views, R_{P}, necessary to have a constant quality, *i.e.,* $D_{\text{K}} = D_{\text{P}}$,

$$R_{\text{P}} = \frac{1}{\beta_{\text{P}}} \left(\beta_{\text{K}} R_{\text{K}} - \log_2 \left(\frac{\mu_{\text{K}} \sigma_{\text{K}}^2}{\mu_{\text{P}} \sigma_{\text{P}}^2} \right) \right). \tag{5}$$

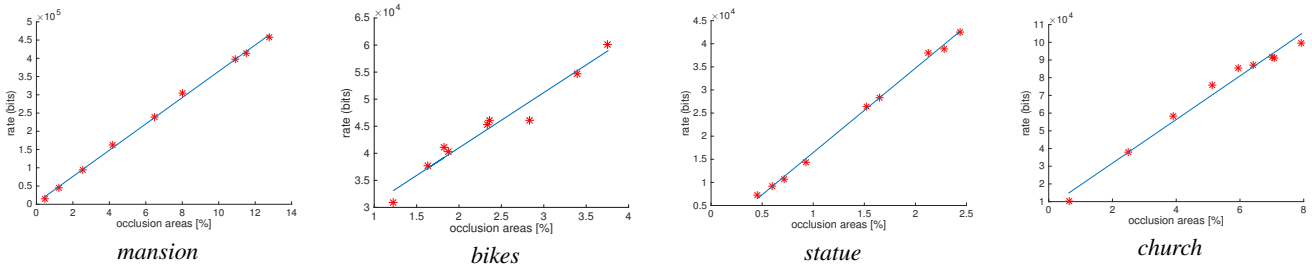
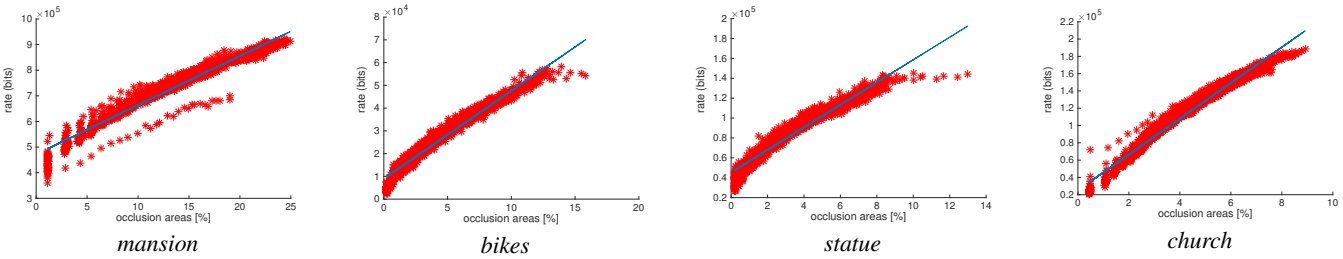
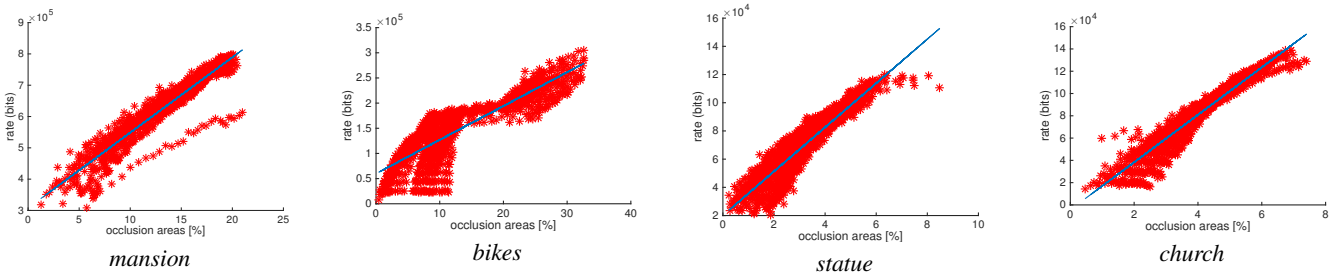


Fig. 4. Evolution of the predicted coding rate for different views as a function of its dissimilarity with the reference view, for a constant MSE distortion over views (namely 10). The predicted views are coded using the distributed coding scheme proposed in [22].



(a) one reference view (P-frames)



(b) two reference views (B-frames)

Fig. 5. Evolution of the predicted coding rate for different views as a function of its dissimilarity with the reference view(s). The predicted views are coded using the 3D-HEVC codec.

This formula allows us to relate the rate per pixel of a predicted view to the one of a reference view. This rate is used to define $\rho(D_{\mathrm{K}}) = MR_{\mathrm{P}}(D_{\mathrm{K}})$ in Eq. (2), where M is the number of pixels in the images. From different experiments, we have noticed that the parameters β_{K}, β_{P}, μ_{K} and μ_{P} vary quite a lot depending on the sequence.

IV. Optimal reference view positioning

A. Problem formulation

We formulate now the problem of optimal reference view positioning, where we search for the optimal number of key views, and their position, in order to minimize the rate under quality constraints. In other words, the solution of our problem is characterized by the number of reference views, N_{K}, the segments $\mathcal{S} = \{\mathcal{I}_1, \ldots, \mathcal{I}_{N_{\mathrm{S}}}\}$ (each $\mathcal{I}_l$ containing the indices of the non-reference views predicted from the same key views), and the positions $\mathcal{K} = \{\mathcal{J}_1, \ldots, \mathcal{J}_{N_{\mathrm{S}}}\}$ of the associated key views, where N_{S} is the number of segments[3].

First, we assume that a non-reference view can be predicted from one or multiple reference views. We recall that the number of reference views used for the prediction of non-reference view is N_{pred} and it depends on the system settings or the multiview encoder characteristics. In our framework, we assume that N_{pred} is the same for all the non-reference views. For example, if $N_{\mathrm{pred}} = 1$ the non-reference views are extrapolated from neighbor key views, and if $N_{\mathrm{pred}} = 2$, they are rather interpolated.

We are now looking for the optimal key view allocation, such that a constant quality is reached for all the views. This means that the rate of each view is adjusted in order to guarantee a constant distortion over the view set. Under these hypotheses, an optimal key view positioning corresponds to a solution where the coding cost is minimized. The coding cost is given by the sum of the segment sizes. Each segment size is the sum of the frame rates composing this segment, namely the rate of the key view ($r_{\mathrm{K}} = MR_{\mathrm{K}}$) and of the predicted views (r_{P}). The problem of key view positioning is thus defined by the following optimization problem:

$$(N_{\mathrm{K}}^*, \mathcal{K}^*, \mathcal{S}^*) = \underset{(N_{\mathrm{K}}, \mathcal{K}, \mathcal{S})}{\arg\min} \left(\sum_{k=1}^{N_{\mathrm{K}}} r_k + \sum_{l=1}^{N_{\mathrm{S}}} \sum_{i \in \mathcal{I}_l} r_{\mathrm{P}}(i, \mathcal{J}_l) \right).$$

Using Eq (2), the problem formulation becomes:

$$(N_{\mathrm{K}}^*, \mathcal{K}^*, \mathcal{S}^*) = \underset{(N_{\mathrm{K}}, \mathcal{K}, \mathcal{S})}{\arg\min} \left(\sum_{k=1}^{N_{\mathrm{K}}} r_k + \sum_{l=1}^{N_{\mathrm{S}}} \sum_{i \in \mathcal{I}_l} \Big(\rho(D)\gamma(i, \mathcal{J}_l) + r_0(D) \Big) \right). \quad (6)$$

The above problem does not have any straightforward solution for arbitrary multiview datasets and generic navigation paths apart from an exhaustive search approach. In the next section, we propose a new

[3] if $N_{\mathrm{pred}} = 1$, we have $N_{\mathrm{S}} = N_{\mathrm{K}}$, and if $N_{\mathrm{pred}} = 2$, we have $N_{\mathrm{S}} = N_{\mathrm{K}} - 1$

optimization method to find the solution $(N_{\mathrm{K}}^*, \mathcal{K}^*, \mathcal{S}^*)$ to the problem (6), by proper decomposition according to the key view positions.

B. Shortest path algorithm

In this section, we consider two instances of the optimization problem in Eq. (6), which correspond to two values of the number of key views used in prediction N_{pred}, respectively $N_{\mathrm{pred}} = 1$ and $N_{\mathrm{pred}} = 2$. These values correspond to the most common of the scenarios in view prediction, namely extrapolation and interpolation. For both cases, we cast the optimization problem as the search of the shortest path in a graph.

1) Coding with view extrapolation: Let us start with the case $N_{\mathrm{pred}} = 1$. In this situation, the non-reference views are predicted with only one reference view. We first build a graph as illustrated in Fig. 6, where the vertices (j, i) correspond to the case when view I_i is predicted with reference view I_j, for all $j \leq N$ and $i \leq N$. These vertices represent all the coding options for each of the views. A coding solution is therefore described by a path from node $(1, 1)$ to node (N, N) with a succession of vertical jumps and horizontal segments, as illustrated in Fig. 6. Coding solution represents the structure of the segment. In particular each segment corresponds to a sequence of nodes $(i, j)_{i \in \mathcal{I}}$, made of views $i \in \mathcal{I}$ predicted from view j. For example, the coding solution (a) in Fig. 6 is made of two key views I_2 and I_5 and two attached segments $\mathcal{I}_1 = \{1, 3\}$ and $\mathcal{I}_2 = \{4, 6, 7, 8\}$. The corresponding path in the graph is made of two horizontal segments on lines 1 and 3 (the indices of the key views) between respectively columns $1 - 3$ and $4 - 8$ (the indices $\mathcal{I}_1$ and $\mathcal{I}_2$ of the predicted views in the segments). Similarly, the coding solution (b) is made of three horizontal segments and three key views. The number of horizontal segments thus corresponds to the number of key views N_{K}.

In order to make the shortest path problem in graph Γ equivalent to the minimization problem in Eq. (6), we have to build the connections and weights such that a typical solution path (as described before) would have the cost as in Eq. (6). For this purpose, the edges between vertices are built under four rules detailed in Fig. 6. Rule 1 and 2 build the edges in the first and last columns of vertices and set the weights to 0. All views are thus possible candidates for being the first and the last reference. Rule 3 sets the vertical edges that correspond to the beginning of a new coding segment. An edge linking vertices (j, i) and $(j', i + 1)$ represents the end of a segment having I_j as reference view, and the beginning of a new segment having $I_{j'}$ as reference. The border between these two segments is between the two predicted views I_i and I_{i+1}. The cost of this kind of edge is naturally the rate of a key view, r_{K} (it corresponds to the cost of starting a new segment). We have the constraint that $j' > i$ since the reference view should be within a segment. In other words, all the vertical edges cross the diagonal (the line where $i = j$). Finally, Rule 4 sets the horizontal connections. An horizontal edge corresponds to including a new predicted view in the current segment. In this sense, the edge cost corresponds to the rate of a predicted view r_{P} (Eq. (2)).

We prove in the Appendix that if we run a shortest path algorithm (*e.g.*, Djikstra [35]) we obtain the minimal cost, hence the optimal values for N_{K}^*, $\mathcal{K}^*$ and $\mathcal{S}^*$, which solves the problem in Eq. (6).

2) Coding with view interpolation: Let us now study the case $N_{\mathrm{pred}} = 2$. In this case, a segment is made of two reference views at its extremes. The extremes are the rightmost and leftmost views of the segment, in order to enable interpolation of all the views in between. The graph construction is a bit different from the case of $N_{\mathrm{pred}} = 1$ and is summarized in Fig. 7. Since two views I_j and $I_{j'}$ now determine a segment, it can be indicated by an edge that links the two views. The associated cost is thus the addition of r_{K} (the cost of an additional reference view) and the sum of all the intermediate view rates $\rho\gamma\big(i, \{j, j'\}\big) + r_0$. The algorithm starts with a cost of r_{K} which corresponds to the transmission cost of I_1 and then counts the reference view cost when they are chosen to close a segment. The problem is now equivalent to the one posed in Eq. (6). As in the previous case, we run a Djikstra algorithm between vertices 1 and N.

V. Experiments

A. Analysis of the optimal positioning

Before testing the proposed algorithm on real datasets, we analyse here the behaviour of our solution for specific situations in order to illustrate its properties. In the case of $N_{\mathrm{pred}} = 1$, we study synthetic scene as shown in Fig. 9, and compute the corresponding dissimilarity matrices. A typical dissimilarity matrix is shown in Fig. 8, where a dissimilarity of 1 is shown in white and a dissimilarity of 0 is black. A position (i, j) in this matrix indicates the dissimilarity of view j when view i is used as reference. For example the view subset pointed by arrow 1 corresponds to a navigation segment where the content is varying quickly from one view to another one. It is due for example to close objects or to large distance between cameras. On the contrary the region pointed by arrow 2 contains views that are highly similar to each other. Two partitioning solutions are represented in this matrix by two paths going from the up left corner to the bottom right corner. Each horizontal segment on row i points the views (column indices) belonging to a segment attached to the reference view i. In the example in Fig. 8, the blue positioning solution is made of 8 segments, with reference views: $\mathcal{K} = \{6, 19, 32, 45, 57, 70, 82, 95\}$. We evaluate the cost of a partitioning as explained in Eq. (6). The cost is expressed as a normalized cost, where the normalization factor is the cost of a reference view. We see in the synthetic example in Fig. 8 that the red path follows the evolution of the inter-view dissimilarities (*e.g,* smaller segment in region 1) and would intuitively lead to a lower cost than the equidistant positioning (blue). In all the following experiments, we compare our optimal positioning algorithm with the solution classically adopted in the literature, *i.e.,* an equidistant positioning not aware of the scene. Since this baseline method does not have any tool for determining the optimal N_{K}, we run a full search algorithm on the N_{K} values. For each N_{K}, we position the reference views equidistantly (*i.e.,* with a constant view index interval between them), and we evaluate the rate cost of such a solution. We pick the N_{K} which has the minimum total rate cost and we compare it with the rate obtained with our optimal solution (we recall that it avoids any full search algorithm). Therefore, in the following, when we present a gain of our solution with respect to the equidistant solution, it corresponds to the comparison of the rate cost obtained with our algorithm, and the one obtained with the best equidistant solution.

We first consider a scene made of a flat background and some foreground objects. They are 2D squares, parallel to the background and placed at different distances. We assume that the camera set is also on a 1D horizontal line, parallel to the background. We further consider that all the cameras are parallel and rectified. A view from the top of such a scene is shown in Fig. 9. For this scene, the dissimilarity γ between two consecutive views can be formally estimated. Let us assume that the distance between two views is δ. We have the following cases which all rely on the relationship that the disparity is proportional to the inverse of the depth.

- If no foreground is visible in the reference view, all the image content of the reference view is shifted with the same disparity value equal to $f\delta/Z_b$, where Z_b is the distance between the background and the reference camera plane, and f is the focal

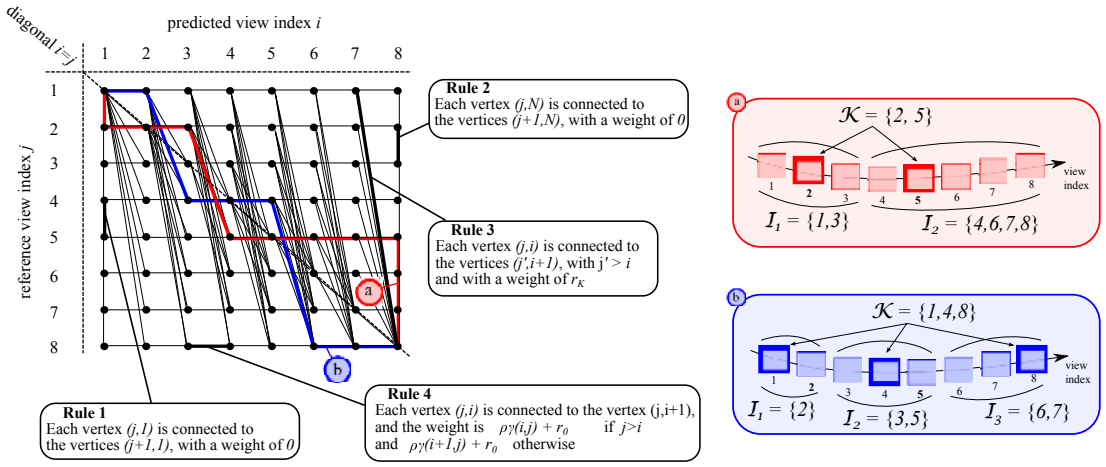


Fig. 6. Graph initialization rules for the shortest path algorithm. Two partitioning solution (a) and (b) are drawn as paths in the graph: horizontal segments correspond to segments in $\mathcal{S}$ and vertical jumps are in correspondence with the key views.

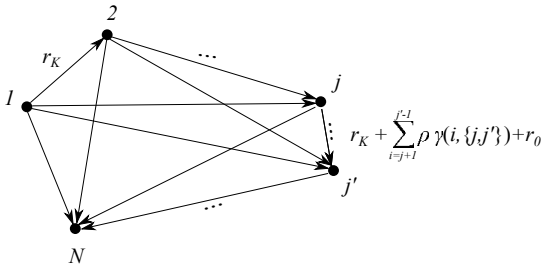


Fig. 7. Graph initialization for shortest path algorithm with $N_{\text{pred}} = 2$. Nodes are the reference views. Each one is linked to the next ones only with a cost of $r_{\text{K}} + \sum_{i=j+1}^{j'-1} \left(\rho\gamma(i,\{j,j'\}) + r_0\right)$. The target path reaches vertices 1 to N.

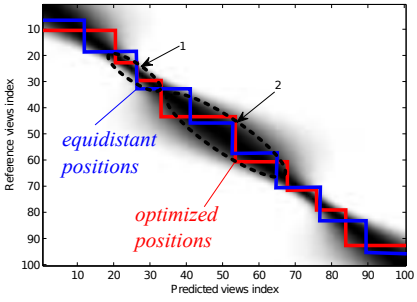


Fig. 8. Black and white respectively indicate a dissimilarity of 0 and 1. A point (j, i) in this matrix indicates the similarity of view i using j as reference. Blue and red paths respectively represent the equidistant and optimized reference view positions. Arrows 1 and 2 show two different correlation modes (resp. low and high) between neighboring views.

length. Therefore, the dissimilarity reads :

$$\gamma = \frac{f\delta}{Z_b}\frac{1}{W},$$

where W is the width of the image. This relationship is linear with the distance, and if there is no foreground object in the scene, the optimal distribution would lead to an equidistant reference view distribution (only depending on the distance). We show in Fig. 10 (a) and (b) the theoretical comparison with equidistant reference view positioning for a scene without background using the rate models of Sec. III. We see that there is no gain between our solution and the equidistant ones. Our solution keeps however one advantage, which is the fact that it finds the optimal N_{K} and thus avoids a full search algorithm.

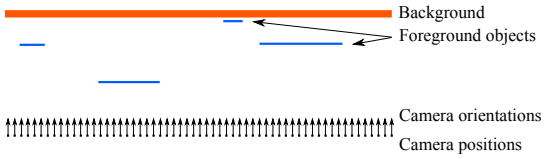


Fig. 9. Toy-example scene from top. Foreground objects are parallel to the background and the camera planes. All the cameras are parallel and positioned on a 1D line.

- If a foreground object at depth Z_f and of height H_f is visible on the reference camera, the dissimilarity increases with respect to the previous case, because of the occlusions. The normalized size of the occlusion is equal to $\frac{f\delta}{Z_b}\frac{1}{W}\frac{H_f}{H}$ (where H is the height of the predicted image). Then, the dissimilarity is equal to

$$\gamma = \frac{f\delta}{Z_b}\frac{1}{W} + \frac{f\delta}{Z_b}\frac{1}{W}\frac{H_f}{H}.$$

$\frac{H_f}{H}$ corresponds to the relative height of the foreground object in the image. However, this is true only if the distance δ is not too big. After a certain point, the foreground disappears from the scene and the second term decreases linearly to 0, and the similarity thus comes back to the first case where no foreground objects were visible. This equation can be generalized if more than one foreground object is visible from the reference view.

In Fig. 10 (c) and (d), we show the partitioning results for a synthetic scene made of 10 foreground objects. These curves have been calculated using rate models of Sec. III. We see that the optimal partitioning (red curve) has a gain compared to the equidistant positioning (blue curve). We can conclude that our solution brings improvements for scenes where the geometry is not similar over the view set.

In addition to the previous examples, we build artificial dissimilarity matrices Γ, with a dissimilarity that is varying sinusoidally. Although these are synthetic variations, they might be realistic for complex scene or view repartition. In Fig. 11, we show the obtained curves for two artificial scenes. We see that even with the best N_{K} view equidistant positioning, our optimized solution leads to a rate gain of 34% and 26%. The optimal positioning is shown in Fig. 11(b) and (d). We see that our reference view positioning solution better follows the variation of the correlation. More precisely, the size of

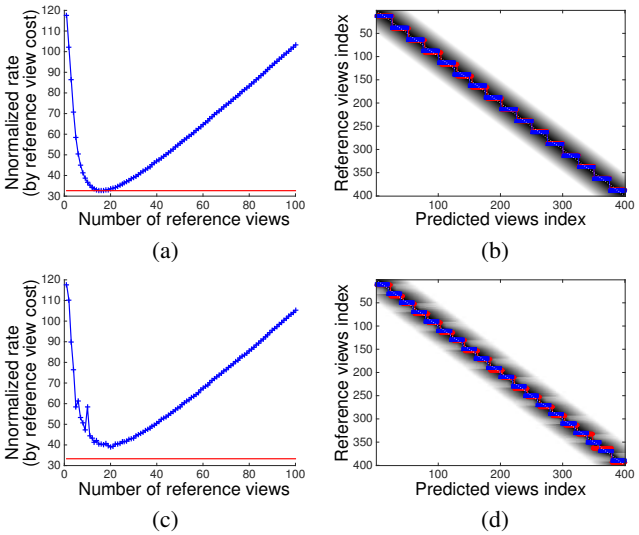


Fig. 10. Positioning results for scenes without foregrounds (a) and (b), and with 10 foregrounds (c) and (d). Blue and red curves respectively correspond to equidistant and optimized reference view positioning. For each synthetic scene (*i.e.*, for each row), left figure is the rate cost for different values of N_{K} and right one shows the positioning.

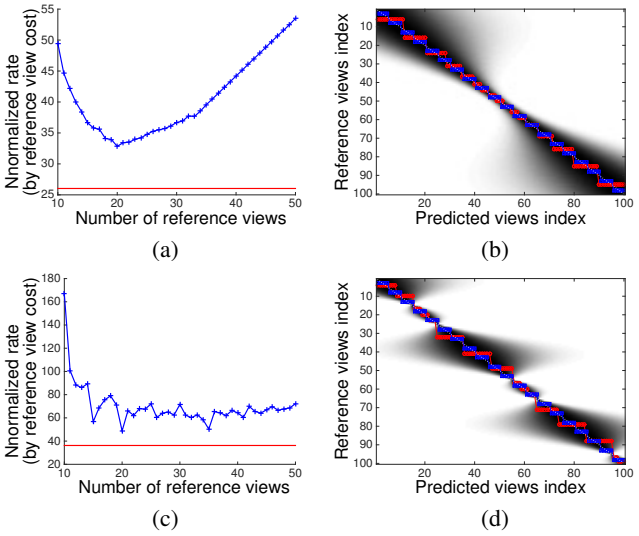


Fig. 11. Blue and red curves respectively correspond to equidistant and optimized reference view positioning. For each synthetic scene (*i.e.*, for each row), the left figure is the rate cost for different values of N_{K} and the right one shows the positioning.

the segments is larger for highly correlated views.

B. Applications to existing coders

Now we test our reference view positioning algorithm on real datasets. The proposed study is mostly developed for systems with a high number of views and for which depth maps are available. We therefore use the super multi-view dataset provided by Disney Research in [36], [3]. It is made of 100 aligned views of a static scene. Instead of using the original version of the dataset, we have created a more challenging scenario. The views have sub-sampled irregularly. In particular, we have selected 25 views, namely, 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, in the dataset of [36]. We have also tested our solution on a challenging test sequence, called *new Tsukuba* [39]. This synthetic dataset is made of 1600 views (color+depth), in which the camera transitions can be very complex (*e.g.,* large rotations).

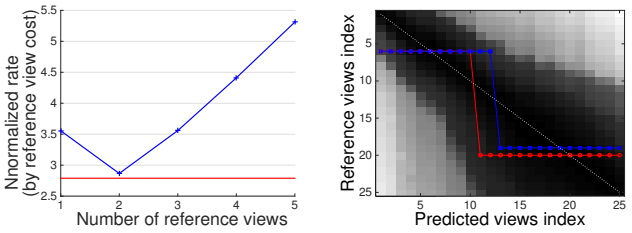


Fig. 12. Positioning results for "Bikes" datasets (51 views). Blue and red curves respectively correspond to equidistant and optimized reference view positioning. Left figure is the rate cost for different values of N_{K} and right one shows the positioning.

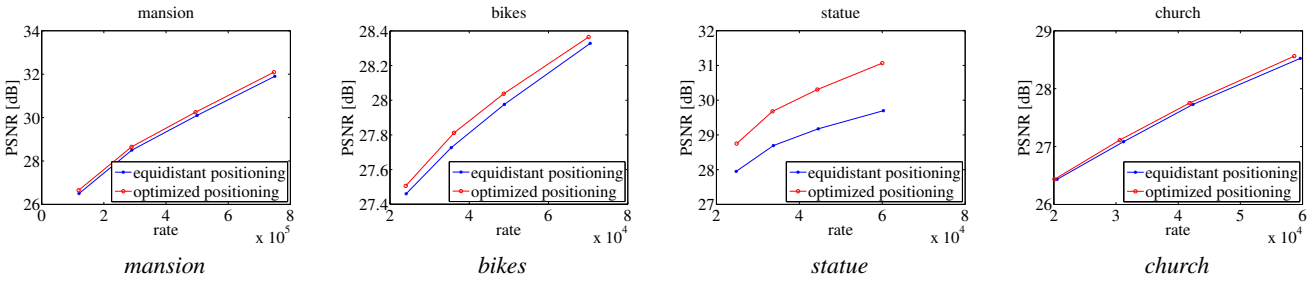
	[22]	3D-HEVC	
N_{pred}	1	1	2
mansion	-5.80	-1.97	-1.80
bikes	-7.76	-1.59	-0.96
statue	-47.75	-2.19	-5.48
church	-2.79	-1.49	-2.17

TABLE II
RATE REDUCTION $\Delta_{\mathrm{R}}[\%]$ OF OUR OPTIMIZED KEY VIEW POSITIONING W.R.T. AN EQUIDISTANT APPROACH

As a preliminary test for setting the parameters of the experiments we have computed the ratio $\frac{R_{\mathrm{P}}}{R_{\mathrm{K}}}$ for each sequence, where R_{P} and R_{K} are the bit-rates (in bits per pixels) necessary for obtaining the same distortion in the predicted views (on the occlusion zones) and in the texture key views, respectively, as defined in Eq. (4) and (3). This ratio is averaged for all the available views (for an equidistant path with two key views) and for different values of QP for the key views. This ratio strongly depends on the statistical properties of the occlusion regions in the predicted views. For example, smooth occluded region are coded more efficiently than highly textured ones, which impacts on this ratio.

Then, we have measured the impact of our key view positioning solution on the performance of the distributed source coder proposed in [22]. The texture key views (of reference views) are coded with H.264/AVC at four different QPs, namely 31, 34, 37 and 40. The corresponding QP for depth key views is chosen according the empirical rule proposed in [37]. Then, we have computed the Rate Distortion performance of our solution and the optimal equidistant scheme (*i.e.,* the optimal number of N_{K} reference view equidistantly positioned on the view set), as shown in Fig. 12. For each QP, we have computed the total bit-rate (the bit-rate for predicted views is chosen in order to have the same distortion on the occluded predicted view and the key view) and the average PSNR on all the views. The Rate Distortion curves are in Fig. 13. The bit-rate reduction expressed by the Bjontegaard metric [38] are shown in Tab. II.

In order to further validate our solution with another coder, we have implemented it within the 3D-HEVC coding standard. The depth maps are coded only for INTRA views. The QP used for INTRA depth maps is calculated from the QP used for INTRA, by the empirical rule proposed in [37]. The QPs used for texture are 31, 34, 37 and 40. As for the extrapolation techniques, we have sub-sampled irregularly the views in order to have a non trivial solution with two INTRA views at the two extremes and all B-Frames for the other views. As done before, we have compared our optimized positioning with the optimal equidistant scheme (*i.e.,* the optimal number of N_{K} reference view equidistantly positioned on the view set). The rate-distortion performance can be found in Tab. II, for two configurations: $N_{\mathrm{pred}} = 1$ and $N_{\mathrm{pred}} = 2$.



mansion *bikes* *statue* *church*

Fig. 13. Rate (bits) PSNR (dB) comparison between optimal and equidistant reference view positioning, using the DSC coder, [22] based on predictions with one reference view.

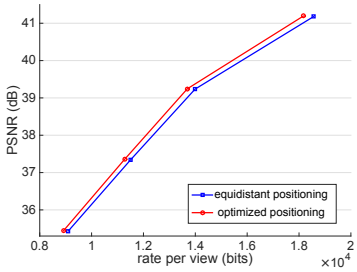


Fig. 14. Rate (bits) PSNR (dB) comparison between optimal and equidistant reference view positioning, using 3D-HEVC, based on predictions with two reference views, for *new Tsukuba*.

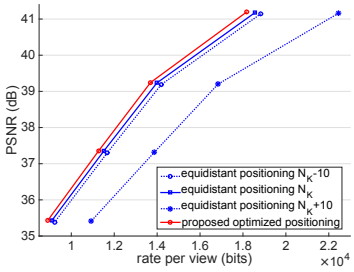


Fig. 15. Rate (bits) PSNR (dB) performance of our solution and three equidistant with different values of reference view (N_{K} being the optimal), using 3D-HEVC, based on predictions with two reference views, for *new Tsukuba*.

Moreover, we have compare our key view positioning solution with the optimal equidistant one for the "new Tsukuba" dataset. We have encoded the 1600 views with 3D-HEVC coder with $N_{\mathrm{pred}} = 2$, using both optimized and equidistant configurations. The results are shown in Fig. 14. The Bjontegaard gain measured for this scenario is 2.23% in rate reduction.

C. Discussion

In the previous section, we have shown that the proposed solution leads to some rate gains in DVC or 3D-HEVC coders. These gains demonstrate the benefit of relying on the geometry information when defining the reference views. More precisely they show that, once the optimal number of key views is known, the proposed rate model based on depth and similarity metric is reliable to efficiently position them on the 1D view set. In other words, the geometry content has a direct impact on the rate performance, and the rate model is an efficient way to take it into account.

The above results have to be interpreted as rate gains between our method and an equidistant approach that is already a sort of oracle method. More precisely, our approach is compared with the optimal equidistant view positioning method with an optimal number of reference viewpoints. But no method in the literature proposes a smart way to obtain such an optimal number. The equidistant algorithm has already high performance but it does not correspond to an actual practical solution. We therefore propose here another experiment to measure one of the main benefits of our proposed solution, namely a solution to find the optimal number of key views without a full search. In this experiment, the dataset is *new Tsukuba* and the codec is 3D-HEVC with $N_{\mathrm{pred}} = 2$. Let N_{K} be the optimal number of reference views found by our algorithm. In this example, it is 179 (for a set of 1600 views). It is easily understandable that a full search over the 1600 possible values of N_{K} is impossible. Let us therefore consider the case of an ad hoc method that makes an error of 6% in the determination of the optimal N_{K}. This error corresponds to more or less 10 additional or missing key views. In Fig. 15, we show the performance of such suboptimal equidistant methods. We see that the performance of our solution is much higher (especially when the number of reference view is higher). In that case, the RD gain measured with Bjontegaard is 21%.

In summary, the interest of the proposed method is mostly twofolds: i) to determine the optimal number of key views and ii) to position them optimally in the view set. Both contributions are supported by a new and accurate rate model, and a shortest path algorithm formulation. The proposed method thus avoids a full search that can be very complex in practical multiview settings.

VI. Conclusion

In this paper, we have proposed an approach for optimally choosing the number and the positions of reference views in a multi-view image coding scheme. For that purpose, we have proposed a new dissimilarity metric and linked it to the coding rate of the predicted views *via* a rate-distortion model, validated on multiple sequences. One of the main advantages of our solution is to perform the view selection based only on the knowledge of the dissimilarity metric, which is simply deduced from the geometry information. Maximizing the similarity is actually equivalent to minimizing the rate. Compared to a full search solution that would run the real prediction and coding of views for each candidate subset of reference views, our method builds its optimal solution based only on the original depth maps information. Future work may focus on the extension of our method for view sets of higher dimension (ex: 2D).

VII. Appendix: Proof of Optimality

In this section, we prove that *finding the shortest path in the graph of Fig. 6 is equivalent to finding the optimal solution of Eq (6).*

Proof: Let us consider a path following the rules detailed in Sec. IV.B.1 and Fig. 6, with a shape of stairs. We want to prove that the cost associated to this path is equivalent to the formula inside the minimization of Eq (7), since the shortest path found by the Djikstra algorithm [35] for example, is the path that has the lowest cost among

all possible paths. Since the path is made of different horizontal sub-path (see the rules detailed in Sec IV.B.1 and Fig. 6), we first calculate the cost of an horizontal sub-path (e.g., between vertices (j, i_1) and (j, i_2)): $\sum_{i=i_1}^{i_2}(\rho\gamma(i,j)+r_0)$, which is the sum of the edge weights composing this segment (*i.e.,* the sum of the coding rate of the non-reference views in the coding segment). If we add the cost of every sub-path composing a total path, we obtain the right term, namely $\sum_{l=1}^{N_\mathrm{S}}\sum_{i\in\mathcal{I}_l}\Big(\rho(D)\gamma(i,\mathcal{J}_l)+r_0(D)\Big)$. If we add the costs of all the vertical transitions, r_K (namely the coding rate of the key views), we obtain the term $\sum_{k=1}^{N_\mathrm{K}-1} r_k$. This almost corresponds to the left term in Eq. (7), since we miss one key view cost r_K (actually, the first one). This is why we consider each path with a cost augmented of r_K. The key frame selection problem is thus equivalent to finding the path with minimal cost between nodes $(1, 1)$ and (N, N).

This proof demonstrates the optimality of our solving method in the sense of Eq (6) based on the model of Eq. (2). However, we note that the optimality in terms of actual rate performance is not formally demonstrated, but it is strongly supported by the rate model validation introduced in Sec. III.

REFERENCES

[1] M. Tanimoto, "FTV: Free-viewpoint television," *IEEE Signal Processing Magazine*, vol. 27, no. 6, pp. 555–570, Jul. 2012.

[2] F. Dufaux, B. Pesquet-Popescu, and M. Cagnazzo, *Emerging Technologies for 3D Video: Creation, Coding, Transmission and Rendering*. Wiley, 2013.

[3] 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.

[4] S. Martull, M. Peris, and K. Fukui, "Realistic CG stereo image dataset with ground truth disparity maps," in *ICPR Workshop*, 2012.

[5] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, "Efficient prediction structures for multiview video coding," *IEEE Trans. on Circ. and Syst. for Video Technology*, vol. 17, no. 11, pp. 1461–1473, Nov. 2007.

[6] Y. Chen, Y. Wang, K. Ugur, M. Hannuksela, J. Lainema, and M. Gabbouj, "The emerging MVC standards for 3d video services," *EURASIP J. on Adv. in Sign. Proc.*, vol. 2009, pp. 1–13, 2009.

[7] K. Müller, P. Merkle, and T. Wiegand, "3D video representation using depth maps," *Proc. IEEE*, vol. 99, no. 4, pp. 643–656, Apr. 2011.

[8] S. Yea and A. Vetro, "View synthesis prediction for multiview video coding," *EURASIP J. on Sign. Proc.: Image Commun.*, vol. 24, pp. 89–100, 2009.

[9] S. Shimizu, M. Kitahara, H. Kimata, K. Kamikura, and Y. Yashima, "View scalable multiview video coding using 3-d warping with depth map," *IEEE Trans. on Circ. and Syst. for Video Technology*, vol. 17, no. 11, pp. 1485–1495, Nov. 2007.

[10] M. Hannuksela, D. Rusanovsky, W. Su, L. Chen, R. Li, P. Aflaki, D. Lan, H. Joachimiak, M. amd Li, and M. Gabbouj, "Multiview-video-plus-depth coding based on the advanced video coding standard," *IEEE Trans. on Image Proc.*, vol. 22, pp. 3449–3458, 2013.

[11] C. Guillemot, F. Pereira, L. Torres, T. Ebrahimi, R. Leonardi, and J. Ostermann, "Distributed monoview and multiview video coding: Basics, problems and recent advances," *IEEE Signal Processing Magazine*, vol. 24, no. 5, pp. 67–76, Sep. 2007, spec. Iss. on Sig. Process. for Multiterminal Commun. Syst.

[12] T. Maugey, I. Daribo, G. Cheung, and P. Frossard, "Navigation domain representation for interactive multiview imaging," *IEEE Trans. on Image Proc.*, vol. 22, no. 9, pp. 3459–3472, Sep. 2013.

[13] H. . F. W. Bin Li ; Jizheng Xu ; Li, "Optimized reference frame selection for video coding by cloud," in *IEEE Int. Workshop on Multimedia Sig. Proc.*, Hangzhou, China, Oct. 2011.

[14] D. Alfonso, B. Biffi, and L. Pezzoni, "Adaptive GOP size control in H.264/AVC encoding based on scene change detection," in *Signal Processing Symposium NORSIG*, Rejkjavik, Iceland, Jun 2006.

[15] C. Yaacoub, J. Farah, and B. Pesquet-Popescu, "Content adaptive gop size control with feedback channel suppression in distributed video coding," in *Proc. IEEE Int. Conf. on Image Processing*, Cairo, Egypt, Nov. 2009.

[16] 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.

[17] 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.

[18] G. Cheung, V. Velisavlevic, and A. Ortega, "On dependent bit allocation for multiview image coding with depth-image-based rendering," *IEEE Trans. on Image Proc.*, vol. 20, pp. 3179–3194, 2011.

[19] J. Chakareski, V. Velisavlevic, and V. Stankovic, "User-action-driven view and rate scalable multiview video coding," *IEEE Trans. on Image Proc.*, vol. 22, no. 9, pp. 3473–3484, Sep. 2013.

[20] J. Ohm, D. Rusanovsky, A. Vetro, and K. Muller, "JCT3V-B1006 work plan in 3D standards development," 2012.

[21] 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.

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

[23] M. Cagnazzo, G. Poggi, L. Verdoliva, and A. Zinicola, "Region-oriented compression of multispectral images by shape-adaptive wavelet transform and SPIHT," in *Proc. IEEE Int. Conf. on Image Processing*, Singapore, Oct. 2004.

[24] 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.

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

[26] 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.

[27] J. Gautier, O. Le Meur, and C. Guillemot, "Efficient depth map compression based on lossless edge coding and diffusion," in *Picture Coding Symposium*, Kraków, Poland, May 2012, pp. 81–84.

[28] I. Daribo, G. Cheung, and D. Florencio, "Arithmetic edge coding for arbitrarily shaped sub-block motion prediction in depth video coding," in *Proc. IEEE Int. Conf. on Image Processing*, Orlando, FL, USA, Sep. 2012.

[29] M. Calemme, M. Cagnazzo, and B. Pesquet-Popescu, "Lossless contour coding using elastic curves in multiview video plus depth," *APSIPA Transactions on Signal and Information Processing*, vol. 2014, p. 14, Dec. 2014.

[30] L. Toni, T. Maugey, and P. Frossard, "Correlation-aware packet scheduling in multi-camera networks," *IEEE Trans. on Multimedia*, vol. 16, no. 2, pp. 496–509, 2014.

[31] K. Müller, H. Schwarz, D. Marpe, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, P. Merkle, H. Rhee, G. Tech, M. Winken, and T. Wiegand, "3D high-efficiency video coding for multi-view video and depth data," *IEEE Trans. on Image Proc.*, vol. 22, no. 9, pp. 3366 – 3378, May 2013.

[32] J. Pandel, "Measuring of ickering artifacts in predictive coded video sequences," in *Internat. Work. on Image Analysis for Multim. Interactive Services*, Klagenfurt, Austria, May 2008.

[33] T. M. Cover and J. A. Thomas, *Elements of Information Theory, Second Edition*. Hardcover, 2006.

[34] A. Fraysse, B. Pesquet-Popescu, and J. Pesquet, "On the uniform quantization of a class of sparse source," *IEEE Trans. on Inform. Theory*, vol. 55, pp. 3243–3263, Jul. 2009.

[35] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein *et al.*, *Introduction to algorithms*. MIT press Cambridge, 2001, vol. 2.

[36] "http://www.disneyresearch.com/project/lightfields/."

[37] 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.

[38] G. Bjontegaard, "Calculation of average PSNR differences between RD curves," 13th VCEG-M33 Meeting, Austin, TX, USA, Tech. Rep., Apr. 2001.

[39] M. Peris, A. Maki, S. Martull, Y. Ohkawa, and K. Fukui, "Towards a simulation driven stereo vision system," in *IEEE International Conference on Pattern Recognition (ICPR)*, Tsukuba, Japan, Nov. 2012.