Multiview Video Representations for Quality-Scalable Navigation

A. De Abreu[#⋆], L. Toni[#], T. Maugey[#] , N. Thomos[†] , P. Frossard[#] , F. Pereira[⋆]

[#] *Ecole Polytechnique Fédérale de Lausanne (EPFL),* [†] *University of Essex ,* [⋆] *Instituto Superior Técnico (IST)*
{ana.deabreu, laura.toni, thomas.maugey, pascal.frossard}@epfl.ch
nthomos@essex.ac.uk; fp@lx.it.pt

***Abstract*—Interactive multiview video (IMV) applications offer to users the freedom of selecting their preferred viewpoint. Usually, in these systems texture and depth maps of captured views are available at the user side, as they permit the rendering of intermediate virtual views. However, the virtual views' quality depends on the distance to the available views used as references and on their quality, which is generally constrained by the heterogeneous capabilities of the users. In this context, this work proposes an IMV scalable system, where views are optimally organized in layers, each one offering an incremental improvement in the interactive navigation quality. We propose a distortion model for the rendered virtual views and an algorithm that selects the optimal views' subset per layer. Simulation results show the efficiency of the proposed distortion model, and that the careful choice of reference cameras permits to have a graceful quality degradation for clients with limited capabilities.**

***Index Terms*—Interactive multiview video, multiview video plus depth, navigation range, scalable representations, view synthesis.**

I. Introduction

In an interactive multiview video (IMV) application, different viewpoints are offered to video users who can interactively select the view of their preference within a certain navigation range. A common data format used in these systems is *multiview video plus depth* (MVD), where for each texture frame of a captured view there is an associated depth map, which is required for intermediate view rendering purposes using a depth-image-based rendering (DIBR) technique [1]. However, high quality view synthesis requires the presence of many camera views separated by small distances. This requires important transmission resources, in terms of number of reference views to be sent to clients, which might not always be feasible in realistic scenarios where different users typically have very different bandwidth capabilities. In an IMV system, the adaptation to the different capabilities of the clients could be done by varying the navigation range offered to the users, by reducing the number of reference views or the quality of the views used as references for the view synthesis for a given navigation range. Most previous works in multiview video systems, addressing the bandwidth limitation problem, have focused on optimizing the coding structure [2], [3] or proposing a novel multiview data representation [4], overlooking the transmission aspects. In [5], the authors focus on the delivery strategies of interactive multiview video applications and propose a *layered QoE* concept. However, the considered approach is limited to equally distant cameras, with the virtual views' distortion model being a function of only the distance from the reference views.

In this work, we focus on a scalable multiview video representation in a network characterized by a large diversity of client bandwidth capabilities and channel conditions. In particular, let $\mathcal{V} = \{1, 2, \ldots, V\}$ be the set of views encoded at the sender side. These views are encoded at the same quality, ensuring a consistent distortion level, at least at the available viewpoints. For each coded view $v \in \mathcal{V}$, texture and depth maps are available, allowing the generation of intermediate virtual viewpoints. In particular, at the decoder side each user can reconstruct any view of the discrete set $\mathcal{U} = \{1, 1 + (1/K_1), 1 + (2/K_1), \ldots, V - (1/K_{V-1}), V\}$; being $(K_v - 1)$ the number of views synthesized between the two adjacent coded views v and $v + 1$. Note that, $\mathcal{U}$ defines the set of the total number of views available for user request through decoding or synthesis. Then, the coded views' streams $\mathcal{V}$ are organized into layered subsets $\mathcal{L} = \{\mathcal{L}_1, \cdots, \mathcal{L}_C\}$, in a coarse-to-fine way, offering a progressive increase of the navigation quality. In particular, the finite set of cameras $\mathcal{V}$ is divided in C subsets such that $\mathcal{L}_1 \cup \mathcal{L}_2 \cup \ldots \cup \mathcal{L}_C \subseteq \mathcal{V}$, with $\mathcal{L}_i \cap \mathcal{L}_j = \emptyset, i \neq j$, where $\mathcal{L}_1$ and $\mathcal{L}_C$, are the most and the least important subsets, respectively. When the frames from the c most important subsets of camera are received and decoded, the quality of the interactive navigation is:

$$D_c = D\left(\bigcup_{i=1}^{c} \mathcal{L}_i\right) = \sum_{\substack{u \in \mathcal{U}, \\ u: v_r, v_l \in \bigcup_{i=1}^{c} \mathcal{L}_i}} q_u \; D_u \qquad (1)$$

where D_u is the distortion of view u, v_r and v_l are the closest right and left reference views to view u, and q_u is the view popularity factor, considered to express the probability that a user selects view $u \in \mathcal{U}$ at a switching time instant. Note that, $D_c \geq D_{c+1}$, since each camera views subset is a refinement of the navigation quality experienced by the user.

An analytical model of the distortion of a rendered virtual view is also proposed in this work, where texture and depth maps quality information of the reference views are considered. Finally, a novel dynamic programming-based algorithm is proposed in order to find the optimal subset of the coded views streams $\mathcal{V}$ per layer. Experimental results show the distortion improvement obtained by optimizing the views streams per layer and the efficiency of the proposed virtual views' distortion model.

II. Virtual View Rendering and Distortion Model

At the decoder side, a user is able to reconstruct any view $u \in \mathcal{U}$. If the requested view is not available at the decoder, then it needs to be synthesized. We consider the *depth-image-based rendering* (DIBR) technique to render a view $u \in \mathcal{U}$, using the closest available right and left reference texture and associated depth maps, $v_r = \{v_r^t, v_r^d\}$ and $v_l = \{v_l^t, v_l^d\}$, for $(v_r, v_l) \in \mathcal{V}$. First, for each reference view, each pixel (x, y) is projected into the virtual view position (x', y'). These projected pixels, from the right and left reference views, form the textures $\hat{v}^t_{r,u}$ and $\hat{v}^t_{l,u}$, respectively. We follow a similar approach to the one in [6], where one of the reference views is considered as the dominant view. In particular, we first consider the pixels projected from the closest reference view to the virtual viewpoint. This view is denoted as v_1^t, for $v_1^t \in \{v_r^t, v_l^t\}$, and its projection as $\hat{v}^t_{1,u}$. Then, the missing pixels in $\hat{v}^t_{1,u}$ are filled from the projection of the second reference view, $\hat{v}^t_{2,u}$.

Note also that some pixels may not be available from any of the reference views, due to rounding error and/or disocclusions, these pixels are filled with inpainting methods [7]. Here, a simple inpainting approach based on the interpolation of the neighboring available pixel values is assumed.

Overall, for each pixel (x, y) of the virtual view u, we have:

$$u(x,y) = \begin{cases} \hat{v}^t_{1,u}(x,y) & \text{if} \quad (x,y) \in (1-\alpha)u \\ \hat{v}^t_{2,u}(x,y) & \text{if} \quad (x,y) \in (1-\gamma)\alpha u \\ \text{Inpainting} & \text{if} \quad (x,y) \in \gamma\alpha u \end{cases} \tag{2}$$

where α denotes the proportion of pixels disoccluded in the closest reference view projection, and γ the proportion of pixels from αu that are not available in neither the right nor the left reference view projection.

This leads to the following virtual view' distortion model:

$$\begin{aligned} D_u = (1-\alpha)\left(D_{\hat{v}^t_{1,u}} + D_{\hat{v}^d_{1,u}}\right) + \\ (1-\gamma)\alpha\left(D_{\hat{v}^t_{2,u}} + D_{\hat{v}^d_{2,u}}\right) + \gamma\alpha D_{inp} \end{aligned} \tag{3}$$

where, $D_{\hat{v}^t_{1,u}}$ and $D_{\hat{v}^t_{2,u}}$ denote the average distortion per pixel due to texture errors, and $D_{\hat{v}^d_{1,u}}$ and $D_{\hat{v}^d_{2,u}}$ stand for the average distortion per pixel due to depth map errors, both calculated on the projected pixels from the corresponding reference view. The average distortion per pixel in the inpainted areas is denoted by D_{inp}, which is assumed to take a constant value that only depends on the scene content. The proportion of disoccluded pixels, α and γ, are obtained from the depth maps of the reference views, which are available at the sender side. As pixel intensity values are copied from the reference views to their projections, the distortion of the projected views, $D_{\hat{v}^t_{1,u}}$ and $D_{\hat{v}^t_{2,u}}$, corresponds to the distortion of the reference views $D_{v_1^t}$ and $D_{v_2^t}$, which can be modeled in terms of the rate as $\sigma^2 2^{-2R}$ [8].

Depth maps errors accounts for position errors, and it has been shown that the distortion value of the projected image linearly increases with the distance to the virtual view u [9]. Therefore, in this work, $D_{\hat{v}^d_{1,u}}$ and $D_{\hat{v}^d_{2,u}}$, are linearly modeled as a function of the distance to the reference view, i.e., $D_{\hat{v}^d_{1,u}} = m_D b_{1,u}$, where, $b_{1,u}$ stands for the baseline distance between virtual view u and reference view v_1, while m_D is the growing rate of the distortion of the projected view. The distortion $D_{\hat{v}^d_{2,u}}$ is similarly defined. In this work, we opt for depth maps encoded at low compression ratio (high quality), since they contribute with a small proportion of the overall rate, compared to texture data. Thus, we only consider the distortion due to errors originally present in the depth maps, due to capturing or estimation error.

III. Subset view selection for scalable navigation

After describing the main characteristics of our IMV scalable system, we first formulate the optimization problem addressed in this paper. Then, we propose a dynamic programming based algorithm to solve the optimization problem.

A. Problem formulation

The problem addressed here is to find the optimal subset of coded views' streams $\mathcal{V}$ per layer $\mathcal{L}^* = \{\mathcal{L}_1^*, \cdots, \mathcal{L}_C^*\}$, such that the expected distortion is minimized, while the rate constraints per layer, $R_{max} = \{R_1, \cdots, R_C\}$, are satisfied. This rate set R_{max} is defined given the number of layers and user's bandwidth. Here, R_{max} is an input of our problem. In particular, we have:

$$\mathcal{L}^* = \arg\min_{\mathcal{L}} \sum_{c=1}^{C} p_c D_c \tag{4}$$

$$\sum_{v=1}^{V}\sum_{i=1}^{c} x_{v,i} r_v \le R_c, \qquad \forall c \in \{1, \cdots, C\}$$

where p_c stands for the proportion of users able to receive up to layer $\mathcal{L}_c$, r_v is the rate of the encoded view v and $x_{v,i}$ is a binary decision variable, set to one if the view v is included in the subset of layer $\mathcal{L}_i$, and zero otherwise.

B. Optimal subset view selection algorithm

To solve the problem posed in Eq. (4), we propose an algorithm based on dynamic programming, where a graph is considered to represent all possible feasible solutions. In particular, the novel proposed algorithm proceeds as follows:
1) Graph creation: We create a graph to represent all the possible solutions. Each graph stage corresponds to a layer of the scalable representation model, starting from $\mathcal{L}_1$, and each node corresponds to a possible views subset solution for a given layer. In particular, the *graph creation* process is summarized as follows:

- *Nodes definition* – Given the number of layers and rate constraint R_{max}, the nodes of each layer are created representing all the possible feasible combinations of two, three, and up to V views. Due to the fact that views are encoded at the same quality, if a node in $\mathcal{L}_c$ is fully contained in another node of the same layer, then only the node representing the larger set is kept, as with the same rate constraint R_c more views can be transmitted, meaning that the overall distortion of the views D_c (Eq. 1) is minimized.

- *Links definition* – A link is defined between two nodes in layers $\mathcal{L}_c$ and $\mathcal{L}_{c+1}$, if the node in $\mathcal{L}_c$ is fully contained in the node in $\mathcal{L}_{c+1}$. A node, in $\mathcal{L}_c$ with $c > 1$, that does not have any incoming link is pruned from the graph.

In Fig. 1, a graph is illustrated for the case of V captured views and three layers. Given a rate constraint, R_1, R_2 and R_3, only two views are considered in $\mathcal{L}_1$, while for layers $\mathcal{L}_2$ and $\mathcal{L}_3$ one and two additional views are included in the node subset, respectively. Each graph link has the expected layer distortion cost, given a particular views subset in a layer.

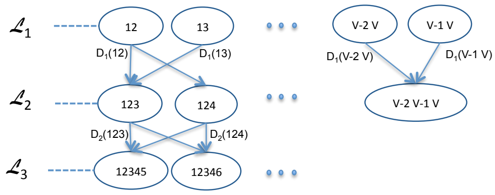


Fig. 1. Graph of proposed algorithm for V coded views and three layers.

2) Graph pruning: We now consider the case when two links converge into a single node, meaning two alternative set of views in $\mathcal{L}_c$ with different RD solutions, provide the same sets of views in a layer $\mathcal{L}_{c+1}$, with the same rate and quality. Applying the Bellman's Optimality Principle [10], if two paths converge into the same node, providing the same views combination (independently of the order), the solution with higher distortion up to that point should be pruned, as from that layer on both solutions have the same remaining subsets of views to be considered, meaning that those paths that are not locally optimal will not be optimal overall. For instance, from Fig. 1 consider the nodes representing views $\{1, 2\}$ and $\{1, 3\}$ from $\mathcal{L}_1$, converging into the node $\{1, 2, 3\}$ in $\mathcal{L}_2$, if $D_1(12) < D_1(13)$, then the link between nodes $\{1, 3\}$ and $\{1, 2, 3\}$ is pruned.

3) Optimal selection: By traversing the graph from $\mathcal{L}_1$ to $\mathcal{L}_C$ we are able to compute the accumulated distortion $\sum_{c=1}^{C} p_c D_c$ for each possible solution and find the optimal subset of the coded views' streams $\mathcal{V}$ per layer, $\mathcal{L}^*$.

IV. Performance Assessment

This section presents the test conditions and performance results obtained in different scenarios when the search of the optimal subset of coded views' streams $\mathcal{V}$ per layer is performed with our proposed algorithm.

A. Test Conditions

We consider three different data sets *Ballet* sequence (1024×768, 15Hz) [11] and *Statue* (2622×1718) and *Bikes* images (2676×1752) [12]. Though the main focus of this work is on video, we have considered the image data sets *Statue* and *Bikes* due to the relatively high quality of their depth maps, compared with the available multiview video sequences. We consider, $V = 6$, for Ballet, and $V = 5$, for Statue and Bikes data sets. In particular, for Ballet video sequence we consider the encoded views $\mathcal{V} = \{0, 2, 3, 4, 5, 7\}$ and the total provided view points $\mathcal{U} = \{0, 1, 2, 3, 4, 5, 6, 7\}$. For the Statute image data set we consider $\mathcal{V} = \{50, 75, 80, 85, 95\}$ and $\mathcal{U} = \{50, 55, 60, 65, 70, 75, 80, 85, 90, 95\}$, while for Bikes $\mathcal{V} = \{10, 20, 30, 40, 50\}$ and $\mathcal{U} = \{10, 15, 20, 25, 30, 35, 40, 45, 50\}$ were considered.

The MVC reference software JMVC v8.2 [13] has been used to encode both texture and depth maps of the considered data sets. The views can be independently or jointly encoded to reduce redundant information between the views. If inter-view prediction is allowed, MVC is applied ensuring that each view is only referenced by views from the same or upper layers, meaning that all the view coding dependencies are available at the user decoder, independently of the layer the user is able to subscribe to. We adopt here the IP PS, where hierarchical B-frames [14] are used in the temporal domain while I- and P-frames are used at the anchor position (frames that do not use temporal prediction for encoding, although they do allow inter-view prediction in the same time instant [15]).

Given the users' bandwidth capacity constraint, the number of layers and the rate allocation per layer is established. In these simulations we consider, a maximum of three, for Ballet, and four layers for Statue and Bikes data sets. Regarding the rate allocation per layer, for Ballet we assume a maximum bit rate per layer of $R_{max} = \{3, 3, 3\}$ Mbps, and for Statue and Bikes a maximum bit budget per layer of $R_{max} = \{3.5, 2, 2, 2\}$Mb and $R_{max} = \{5, 2.5, 2.5, 2.5\}$Mb, respectively.

B. Results and Analysis

Given the test conditions, we first evaluate the performance of the virtual view distortion model and algorithm proposed. Then, given the virtual view distortion model and selection algorithm, inter-view coding prediction with MVC is evaluated.

1) Virtual view distortion model and selection algorithm performance: Assuming that available views are independently encoded, the optimal views' subset $\mathcal{L}^*$ of the coded views' streams is estimated. The optimal solution obtained with the proposed virtual view distortion model (Section II), $\mathcal{L}^*_M$, is compared with the solution where empirical distortion calculation, MSE, is considered, $\mathcal{L}^*_E$. In both cases, $\mathcal{L}^*_M$ and $\mathcal{L}^*_E$ were found with the proposed algorithm (Section III-B). As an additional benchmark, we use a state of the art solution, where the layer subset organization is done based on the minimum distance between encoded and synthesized views [5]. This solution is denoted here by $\mathcal{L}_d$ and it is evaluated in terms of MSE performance.

The results are shown in Table I for the three data sets considered. As it can be seen the subset organization based on the distance between encoded and virtual views, $\mathcal{L}_d$, was never the optimal solution. In general, the distortion model showed a good performance by providing the same optimal solution $\mathcal{L}^*_M$ than the empirical optimal solution $\mathcal{L}^*_E$ for Ballet

TABLE I
COMPARISON OF OPTIMAL EMPIRICAL ($\mathcal{L}^*_E$) AND MODELED ($\mathcal{L}^*_M$) SUBSET SETTING AND A DISTANCE BASED LAYER SETUP ($\mathcal{L}_d$).

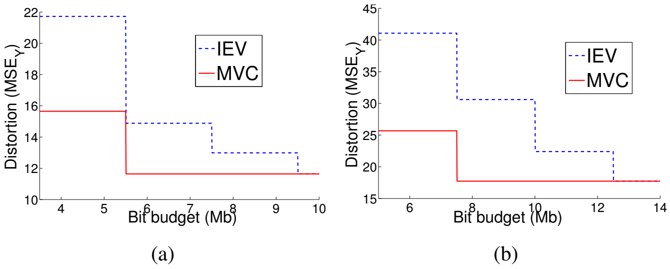
Data Set		$\mathcal{L}_d$	$\mathcal{L}^*_M$	$\mathcal{L}^*_E$
Ballet	$\mathcal{L}_1$	$\{0\ \ 7\}$	$\{2\ \ 5\}$	$\{2\ \ 5\}$
	$\mathcal{L}_2$	$\{2\ \ 5\}$	$\{0\ \ 7\}$	$\{0\ \ 7\}$
	$\mathcal{L}_3$	$\{3\ \ 4\}$	$\{3\ \ 4\}$	$\{3\ \ 4\}$
	D	50.72	38.78	38.78
Statue	$\mathcal{L}_1$	$\{50\ \ 95\}$	$\{50\ \ 95\}$	$\{50\ \ 85\}$
	$\mathcal{L}_2$	$\{80\}$	$\{75\}$	$\{75\}$
	$\mathcal{L}_3$	$\{75\}$	$\{85\}$	$\{95\}$
	$\mathcal{L}_4$	$\{85\}$	$\{80\}$	$\{80\}$
	D	15.69	15.31	15.25
Bikes	$\mathcal{L}_1$	$\{10\ \ 50\}$	$\{20\ \ 50\}$	$\{20\ \ 50\}$
	$\mathcal{L}_2$	$\{30\}$	$\{10\}$	$\{10\}$
	$\mathcal{L}_3$	$\{20\}$	$\{30\}$	$\{30\}$
	$\mathcal{L}_4$	$\{40\}$	$\{40\}$	$\{40\}$
	D	27.78	26.64	26.64

and Bikes data sets, or by providing a close to optimal solution for Statue image data set.

2) Inter-view coding prediction performance: In order to evaluate the performance of the inter-view dependencies with MVC, we use independently encoded views for comparison. We consider the virtual view distortion model and the selection algorithm proposed. Both, the Statue and Bikes image data sets are considered with the same test conditions stated before. For Statue data set, when inter-view dependencies was allowed, only two layers were obtained as the optimal solution, $\mathcal{L}^* = \{\mathcal{L}^*_1 = [50\ 75\ 85]\,;\, \mathcal{L}^*_2 = [80\ 95]\}$, while for the independently encoded case four layers were required in the optimal solution $\mathcal{L}^* = \{\mathcal{L}^*_1 = [50\ 95]\,;\, \mathcal{L}^*_2 = [75]\,;\, \mathcal{L}^*_3 = [85]\,;\, \mathcal{L}^*_4 = [80]\}$. A similar solution was obtained for Bikes data set where, $\mathcal{L}^* = \{\mathcal{L}^*_1 = [10\ 30\ 50]\,;\, \mathcal{L}^*_2 = [20\ 40]$ when inter-view dependencies were enabled in MVC, and $\mathcal{L}^* = \{\mathcal{L}^*_1 = [20\ 50]\,;\, \mathcal{L}^*_2 = [10]\,;\, \mathcal{L}^*_3 = [30]\,;\, \mathcal{L}^*_4 = [40]\}$ when views were independently encoded. This means that by allowing inter-view dependencies, we are able to send more views per layer, which is translated in higher expected quality for the same rate, since we consider that views are encoded at the same quality. This is better illustrated in Fig. 2 where both approaches are compared in terms of bit budget constraint per user and expected distortion D_c, which is determined by the layer the user is able to receive given the bit budget constraint.

V. CONCLUSION

We have proposed an algorithm that efficiently selects the optimal subsets of views streams for a scalable layered transmission in IMV applications. We consider a system where the network is characterized by users with heterogeneous bandwidth capabilities, and we aim to minimize their navigation distortion. A distortion model for the rendered virtual views has been also proposed, which has shown to reproduce closely empirical results when used together with the proposed selection algorithm. It has been shown through simulation, that by adopting the proposed algorithm we are able to reduce the navigation distortion in a scalable IMV application. Future work may focus on the extension of the current optimization algorithm to systems where the reference views do not have fixed quality, but their choice can be RD optimized to further reduce the expected distortion.



(a) (b)

Fig. 2. Independently (IEV) and jointly (MVC) encoded views compared in terms of bit budget constraint per user and expected distortion, for (a) Statue and (b) Bikes image data sets.

REFERENCES

[1] M. Schmeing and X. Jiang, *Depth Image Based Rendering*, P. S. Wang, Ed. Springer Berlin Heidelberg, 2011.

[2] A. De Abreu, P. Frossard, and F. Pereira, “Fast MVC prediction structure selection for interactive multiview video streaming,” in *Proc. of Picture Coding Symposium*, San Jose, CA, USA, December 2013.

[3] Z. Liu, G. Cheung, and Y. Ji, “Optimizing distributed source coding for interactive multiview video streaming over lossy networks,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 23, no. 10, pp. 1781–1794, Oct 2013.

[4] T. Maugey, I. Daribo, G. Cheung, and P. Frossard, “Navigation domain representation for interactive multiview imaging,” *IEEE Trans. on Image Processing*, vol. 22, no. 9, pp. 3459–3472, Sept 2013.

[5] L. Toni, N. Thomos, and P. Frossard, “Interactive free viewpoint video streaming using prioritized network coding,” in *IEEE Int. Workshop on Multimedia Signal Processing*, Pula, Italy, Sept 2013.

[6] Y. Zhao, C. Zhu, Z. Chen, D. Tian, and L. Yu, “Boundary artifact reduction in view synthesis of 3D video: From perspective of texture-depth alignment,” *Broadcasting, IEEE Transactions on*, vol. 57, no. 2, pp. 510–522, June 2011.

[7] Z. Tauber, Z.-N. Li, and M. S. Drew, “Review and preview: Disocclusion by inpainting for image-based rendering,” *Trans. Sys. Man Cyber Part C*, vol. 37, no. 4, pp. 527–540, Jul. 2007.

[8] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY, USA: Wiley-Interscience, 1991.

[9] K. Muller, P. Merkle, and T. Wiegand, “3-D video representation using depth maps,” *Proc. of IEEE*, vol. 99, no. 4, pp. 643–656, April 2011.

[10] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 1999.

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

[12] 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, pp. 73:1–73:12, Jul. 2013.

[13] JMVC 8.2 software. [Online]. Available: garcon.ient.rwth-aachen.de

[14] H. Schwarz, D. Marpe, and T. Wiegand, “Analysis of hierarchical B pictures and MCTF,” in *Proc. of IEEE Int. Conf. on Multimedia and Expo*, Toronto, Ontario, Canada, July 2006.

[15] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, “Efficient prediction structures for multiview video coding,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 17, no. 11, pp. 1461–1473, November 2007.